UNIVERSIDADE DA BEIRA INTERIOR
Engenharia

Performance Assessment of Routing Protocols for IoT/6LoWPAN Networks

José Victor Vasconcelos Sobral

Tese para obtenção do Grau de Doutor em
Engenharia Informática
(3º ciclo de estudos)

Orientador: Prof. Doutor Joel José Puga Coelho Rodrigues

Covilhã, Julho de 2020

Dedication

To God and my parents, Conceição and Antônio Morais

Acknowledgments

Firstly, I would like to thank God for giving me health and strength to face this long journey.

I would like to thank and express my gratitude to my Professor and Supervisor, Joel José Puga Coelho Rodrigues, that carefully advised me in the course of these last four years. His patience, generosity, and knowledge were crucial to the development of this work.

I would also like to thank my parents, Conceição Ribeiro and Antônio Moraes Sobral Neto, to my brother, João Gabriel Sobral, and my girlfriend, Nayra Roberta Alves, for always being by my side, in the good and bad moments, and always give me good energies, even being so far away.

To the Brazilian National Council for Scientific and Technological Development (CNPq) and the Science without Borders program, through the grant number 201155/2015-0, for all the support and trust that was given to me and my doctorate project.

To all my colleagues of the Next Generation Networks and Applications (NetGNA) research group, for the shared moments of happiness and knowledge.

To all the research colleagues of the OASIS Lab, from the Federal University of Piauí (UFPI), Brazil, and of the IoT Reseach Group, from the National Institute of Telecommunication (Inatel), Brazil, for the shared knowledge to the conduction of this thesis work.

To all the Professors and employees from the University of Beira Interior (UBI), and the Instituto de Telecomunicações (IT), Covilhã delegation. This work is partially funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the Project UIDB/EEA/50008/2020.

Finally, to all persons that, in some way, have led me to reach this important achievement in my life.

Foreword

This thesis describes the research work performed in the scope of the doctoral research programme and presents its main contributions and achievements. This doctoral programme and inherent research activities were carried out at the Next Generation Networks and Applications Group (NetGNA) research group of the Departamento de Informática, Universidade da Beira Interior, Covilhã, Portugal and Instituto de Telecomunicações, Delegação da Covilhã, Portugal. The research work was supervised by Prof. Dr. Joel José Puga Coelho Rodrigues and financially supported by the National Council for Scientific and Technological Development (CNPq) through the grant contract 201155/2015-0. This work is also partially funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the Project UIDB/EEA/50008/2020.

List of Publications

Papers included in the thesis resulting from this 4-year doctoral research programme

1. **Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications**

   José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Jalal Al-Muhtadi, and Valery Korotaev

   Sensors, MDPI, ISSN: 1424-8220, Vol. 19, N. 9, Paper Id: 2144, May 2019.

   DOI: doi.org/10.3390/s19092144

2. **A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs**

   José V.V. Sobral, Joel J. P. C. Rodrigues, Ricardo A.L. Rabêlo, José C. Lima Filho, Natanael Sousa, Harilton S. Araujo, and Raimir Holanda Filho

   Journal of Network and Computer Applications, Elsevier, ISSN: 1084-8045, Vol. 107, April 2018, pp. 56-68.

   DOI: doi.org/10.1016/j.jnca.2018.01.015

3. **LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks**

   José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Kashif Saleem, and Vasco Furtado

   Sensors, MDPI, ISSN: 1424-8220, Vol. 19, N. 1, Paper Id: 150, January 2019.

   DOI: doi.org/10.3390/s19010150

4. **Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios**

   José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Kashif Saleem, and Sergei Kozlov

   IEEE Access, IEEE, ISSN: 2169-3536, Vol. 7, August 2019, pp. 107032-107046.

   DOI: doi.org/10.1109/ACCESS.2019.2932718

5. **Multicast Improvement for LOADng in Internet of Things Networks**

   José V.V. Sobral, Joel J. P. C. Rodrigues, Ricardo L. Rabêlo, and Jalal Al-Muhtadi

   Measurement, Elsevier, ISSN: 0263-2241, Vol. 148, December 2019, Article 106931.

   DOI: doi.org/10.1016/j.measurement.2019.106931

Publications resulting from this doctoral research programme included in the thesis as appendixes

6. **Performance Evaluation of Routing Metrics in the LOADng Routing Protocol**

   José V. V. Sobral, Joel J. P. C. Rodrigues, Neeraj Kumar, Chunsheng Zhu, and Raja W. Ahmad

   Journal of Communications Software and Systems, ISSN: 1845-6421, Vol. 13, No. 2, June 2017, pp. 87-95.

   DOI: doi.org/10.24138/jcomss.v13i2.376

7. **ERAOF: A new RPL protocol objective function for Internet of Things applications**

   Natanael Sousa, José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, and Petar Solic

   2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, Croatia, July 12–14, 2017, pp. 1–5.

8. **A Proposal for IoT Dynamic Routes Selection Based on Contextual Information**

   Harilton Da Silva Araújo, Raimir Holanda Filho, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Natanael Sousa, José C. C. L. S. Filho, and José V. V. Sobral

   Sensons, MDPI, ISSN: 1424-8220, Vol. 18, No. 2, Article 353, January 2018, pp. 1-16.

   DOI: doi.org/10.3390/s18020353

Other publications resulting from this doctoral research programme not included in the thesis

9. **A composite routing metric for wireless sensor networks in AAL-IoT**

   José V. V. Sobral, Joel J. P. C. Rodrigues, Kashif Saleem, Juan F. de Paz, and Juan M. Corchado

   9th IFIP Wireless and Mobile Networking Conference (WMNC 2016), Colmar, France, July 11–13, 2016, pp. 168–173.

   DOI: doi.org/10.1109/WMNC.2016.7543985

10. **Performance evaluation of LOADng routing protocol in IoT P2P and MP2P applications**

    José V. V. Sobral, Joel J. P. C. Rodrigues, Kashif Saleem, and Jalal Al-Muhtadi

    1st International Multidisciplinary Conference on Computer and Energy Science (SpliTech 2016), Split, Croatia, July 13–15, 2016, pp. 1–6.

    DOI: doi.org/10.1109/SpliTech.2016.7555943

11. **Performance Assessment of the LOADng Routing Protocol in Smart City Scenarios**

    José V. V. Sobral, Joel J. P. C. Rodrigues, and Augusto Neto

    IEEE First Summer School on Smart Cities (S3C 2017), Natal, Brazil, August 6–11, 2017, pp. 49–54.

    DOI: doi.org/10.1109/S3C.2017.8501394

Resumo

A Internet das Coisas, do inglês *Internet of Things* (IoT), propõe um paradigma de comunicação disruptivo para possibilitar que dispositivos, que podem ser dotados de comportamentos autónomos ou inteligentes, troquem dados entre eles buscando alcançar um objetivo comum. Os cenários de aplicação do IoT são muito variados e podem abranger desde um simples sistema de iluminação para casa até o controle total de uma linha de produção industrial. Na maioria das instalações IoT, as "coisas" são equipadas com um pequeno dispositivo, responsável por realizar as tarefas de comunicação e processamento de dados, que pode sofrer com severas restrições de hardware e energia. Assim, devido às suas características, a rede de comunicação criada por esses dispositivos é geralmente categorizada como uma *Low Power and Lossy Network* (LLN).

A grande variedade de cenários IoT representam uma questão crucial para as LLNs, que devem oferecer suporte aos diferentes requisitos das aplicações, além de manter níveis de qualidade de serviço, do inglês *Quality of Service* (QoS), adequados. Baseado neste desafio, os protocolos de encaminhamento constituem um aspecto chave na implementação de cenários IoT. Os protocolos de encaminhamento são responsáveis por criar os caminhos entre os dispositivos e permitir suas interações. Assim, o desempenho e as características da rede são altamente dependentes do comportamento destes protocolos. Adicionalmente, com base no protocolo adotado, o suporte a alguns requisitos específicos das aplicações de IoT podem ou não ser fornecidos. Portanto, estes protocolos devem ser projetados para atender as necessidades das aplicações assim como considerando as limitações do hardware no qual serão executados.

Procurando atender às necessidades dos protocolos de encaminhamento em LLNs e, consequentemente, das redes IoT, a *Internet Engineering Task Force* (IETF) desenvolveu e padronizou o *IPv6 Routing Protocol for Low Power and Lossy Networks* (RPL). O protocolo, embora seja robusto e ofereça recursos para atender às necessidades de diferentes aplicações, apresenta algumas falhas e fraquezas (principalmente relacionadas com a sua alta complexidade e necessidade de memória) que limitam sua adoção em cenários IoT. Em alternativa ao RPL, o *Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation* (LOADng) emergiu como uma solução de encaminhamento menos complexa para as LLNs. Contudo, o preço da simplicidade é pago com a falta de suporte adequado para um conjunto de recursos essenciais necessários em muitos ambientes IoT. Assim, inspirado pelas desafiadoras questões ainda em aberto relacionadas com o encaminhamento em redes IoT, esta tese tem como objetivo estudar e propor contribuições para melhor atender os requisitos de rede em cenários IoT. Uma profunda e abrangente revisão do estado da arte sobre os protocolos de encaminhamento adotados em IoT identificou os pontos fortes e limitações das soluções atuais. Com base nas debilidades encontradas, um conjunto de soluções de melhoria é proposto para superar carências existentes e melhorar o desempenho das redes IoT. As novas soluções são propostas para incluir um suporte confiável e eficiente capaz atender às necessidades das aplicações IoT relacionadas com suporte à mobilidade, heterogeneidade dos dispositivos e diferentes padrões de tráfego. Além disso, são introduzidos mecanismos para melhorar o desempenho da rede em cenários IoT que integram dispositivos com diferentes tecnologias de comunicação.

Os vários estudos realizados para mensurar o desempenho das soluções propostas mostraram o grande potencial do conjunto de melhorias introduzidas. Quando comparadas com outras abordagens existentes na literatura, as soluções propostas nesta tese demonstraram um

aumento do desempenho consistente para métricas relacionadas a qualidade de serviço, uso de memória, eficiência energética e de rede, além de adicionar novas funcionalidades aos protocolos base. Portanto, acredita-se que as melhorias propostas contribuiem para o avanço do estado da arte em soluções de encaminhamento para redes IoT e aumentar a adoção e utilização dos protocolos estudados.

Palavras-chave

Internet das Coisas, Protocolos de Encaminhamento, *Low power and Lossy Networks*, Protocolo de Encaminhamento RPL, Protocolo de Encaminhamento LOADng, Redes de Sensores sem Fio, Redes Ad Hoc Móveis, Identificação por Radiofrequência, Protocolos Anti-colisão RFID, *Directed Diffusion*, Sistemas *Fuzzy*, Otimização por Colônia de Formigas, *Ant System*, Desempenho de Redes, Qualidade de Serviço.

Extended Abstract in Portuguese

Introdução

Esta seção resume os 4 anos de trabalho de investigação no âmbito desta tese de doutoramento entitulada "*Performance Assessment of Routing Protocols for IoT/6LoWPAN Networks*". No decorrer deste resumo alargado, são descritos o enquadramento do tema, a delimitação do problema e os objetivos de investigação. Adicionalmente, são apresentadas as principais contribuições e conclusões do trabalho, além de perspectivas de trabalhos futuros.

Enquadramento do Tema

O paradigma da Internet das Coisas, do inglês *Internet of Things* (IoT), emergiu recentemente atraindo a atenção da indústria, das pessoas e da comunidade científica. O conceito de IoT propõe a possibilidade de objetos (as "coisas") se comunicarem entre si de forma pervasiva, buscando fornecer facilidades e possibilidade de automação em diferentes campos de aplicação [1]. De acordo com a Bain & Company [2], espera-se que o mercado de IoT cresça dos $ 235 bilhões gastos em 2017 para cerca de $ 520 bilhões em 2021. O relatório ainda revela que os investimentos nas áreas de "Integração de Sistemas" e "Redes" devem crescer entre de 15 % e 40 %, respectivamente, até 2021, em comparação com 2017.

Os cenários de IoT são compostos por dispositivos (que podem tomar decisões de forma autónoma e inteligentes) que podem ser utilizados na realização de atividades diárias para tornar mais fáceis diferentes aspectos da vida das pessoas e das organizações [3]. Atualmente, o conjunto mais significativo de aplicações IoT são focados em monitoramento ambiental, cidades inteligentes, saúde, automação residencial, e na indústria [4]. Devido às particularidades dos vários cenários de aplicação, as redes IoT podem formar uma estrutura complexa composta por dispositivos com diferentes capacidades. Os nós da rede podem divergir em características de hardware, fabricantes e também tecnologias de comunicação.

Em geral, os dispositivos IoT (que representam as "coisas") adotam o uso de microcontroladores (MCU) ou soluções *System-on-Chip* (SoC) para realizar as tarefas de processamento de dados e comunicação sem fios. Assim, esses nós tendem a possuir muitas limitações em termos de hardware relacionadas com a baixa capacidade de processamento e armazenamento. Além disso, dependendo do cenário de aplicação, os dispositivos de rede podem ter características de mobilidade e sofrer com restrições de energia (sendo equipados com baterias). Desta forma, devido às características dos nós que as compõem, as redes IoT são comummente classificadas como *Low-power and Lossy Networks* (LLN) [5]. Adicionalmente, os cenários IoT podem integrar dispositivos que usam diferentes tecnologias de comunicação em uma rede comum para permitir o surgimento de aplicações inovadoras [6]. Neste contexto, uma importante tecnologia frequentemente atribuída para fornecer características de ubiquidade e pervasividade para os cenários IoT é a Identificação por Radiofrequência, do inglês *Radio Frequency Identification* (RFID) [7]. Os sistemas RFID oferecem capacidade de identificação única aos cenários IoT de forma confiável, rápida e com baixo custo, possibilitando que os aplicativos possam rastrear e monitorizar a presença de dispositivos no ambiente em que estão inseridos [8].

Independente do campo de utilização e das tecnologias de comunicação existentes na rede, as aplicações IoT, visando fornecer serviços apropriados às necessidades dos utilizadores, podem requerer o cumprimento de várias métricas de qualidade de serviço, do inglês *Quality of Service* (QoS), relacionadas com a eficiência energética, latência, confiabilidade, disponibilidade e segurança [9]. Assim, baseada nessas necessidades, a escolha do protocolo de encaminhamento torna-se um fator chave para permitir que a redes possam atender às necessidades de QoS das aplicações IoT [10].

Os protocolos de encaminhamento são responsáveis por criar e manter os caminhos entre os nós da rede, além de realizar o envio dos pacotes de dados. Desta forma, o desempenho da rede torna-se altamente depende do comportamento do protocolo [11]. Devido à importância desse componente, a *Internet Engineering Task Force* (IETF) criou o grupo de trabalho *Routing over Low Power and Lossy Networks* (RoLL) para estudar os requisitos das aplicações de LLNs e desenvolver uma nova solução de encaminhamento para os atender. Assim, num esforço inicial, o RoLL apresentou os requisitos de encaminhamento para um conjunto de aplicações de LLNs tais como para a indústria, automação residencial, automação predial e cenários urbanos. Em seguida, com base nas necessidades identificadas, o grupo de trabalho desenvolveu o *IPv6 Routing Protocol for Low-Power and Lossy Networks* (RPL) [12], que foi definido como o protocolo de encaminhamento padrão para as LLNs. O RPL propõe uma solução de encaminhamento proativa onde um dispositivo central (também chamado de roteador de borda, do inglês *border router*) é responsável por construir uma árvore de encaminhamento otimizada para a recolha de dados enviada dos nós para o dispositivo central.

Embora o RPL tenha sido definido como um padrão para as LLNs e, consequentemente, o protocolo de encaminhamento "padrão" para redes IoT [13], vários estudos têm contestado a eficiência da solução proposta e a sua capacidade de atender os requisitos das aplicações IoT. Iova *et al.* [13] realizaram um estudo crítico para avaliar o comportamento do RPL em ambientes IoT. Os autores identificaram várias questões, principalmente relacionadas com o limitado suporte a tráfego ponto-a-ponto (P2P) e ponto-multiponto (P2MP), suporte ineficiente à mobilidade e o elevado consumo de memória que limita a heterogeneidade de dispositivos e a escalabilidade da rede. Adicionalmente, os autores destacaram que o RPL foi projetado tendo em conta, principalmente, o padrão de tráfego multiponto-a-ponto (MP2P), em que os pacotes de dados fluem dos nós para um dispositivo central. Contudo, as aplicações IoT mais inovadoras também podem requerer suporte eficiente e confiável aos tráfegos P2P e P2MP. Para melhor atender esses tipos de tráfego, foram propostas as melhorias P2P-RPL [14] e *Multicast Protocol for LLN* (MPL) [15], mas ambas as abordagens requerem um aumento no uso de memória que pode provocar o surgimento de outros efeitos negativos no desempenho da rede. Por fim, Iova *et al.* endossam que o surgimento de novas soluções de encaminhamento para redes IoT poderiam superar os pontos fracos do RPL.

Baseado nas referidas limitações do RPL e em outras fraquezas apresentadas pelo protocolo que são discutidas na literatura [16] [17], uma solução de encaminhamento menos complexa tem surgido como alternativa ao protocolo padrão. Esta nova solução, designada por *Lightweight On-demand Ad hoc Distance-vector routing protocol – Next Generation* (LOADng) [18], apresenta uma proposta de encaminhamento reativa onde os caminhos entre os nós são construídos sob demanda. Assim, os limitados recursos dos dispositivos da rede somente são

consumidos quando os nós precisam trocar pacotes de dados. Diferente do RPL, o LOADng ainda não é definido como um padrão para LLNs e encontra-se "*under-draft*" num grupo da IETF. Contudo, devido a sua importância, o LOADng foi incluído no padrão G3-PLC ITU-T para comunicações em cenários de *smart grid* [19].

Quando comparado com o RPL, a principal vantagem do LOADng é o seu núcleo mais leve e o comportamento reativo que permite ao protocolo adaptar-se melhor às limitações de hardware dos dispositivos da rede e a suportar tráfego de dados P2P. Em contraste, a característica de trabalho sob demanda pode provocar o aumento da latência extremo-a-extremo da ligação, afetando a QoS da rede. Adicionalmente, devido a sua implementação simplificada, o LOADng não apresenta nenhum suporte a tráfego P2MP, e o suporte à mobilidade é implicitamente atendido somente pelas características reativas do protocolo. Deste modo, nota-se que as principais soluções de encaminhamento atuais para LLNs podem não atender completamente às demandas das aplicações IoT. Assim, esta tese pretende estudar o desempenho dos protocolos de encaminhamento e propor melhorias para mitigar os problemas existentes no Estado da Arte de encaminhamento para IoT. A subseção a seguir é dedica a apresentar a delimitação do problema.

Delimitação do Problema

Nos cenários IoT, um protocolo de encaminhamento deve ser capaz de atender os requisitos da rede de acordo com a aplicação. Embora possam variar, as necessidades das aplicações IoT estão relacionados com suporte para dispositivos móveis e heterogêneos, níveis aceitáveis de QoS, suporte para diferentes padrões de tráfego e eficiência energética. Assim, considerando as demandas das redes IoT e os limitados recursos de hardware e em termos energéticos dos dispositivos, a tarefa do protocolo de encaminhamento torna-se mais difícil uma vez que ele deve ser leve, energeticamente eficiente, altamente confiável, além de requerer baixo poder de processamento.

A IETF, por meio do grupo de trabalho RoLL, propôs o RPL para atender as necessidades das aplicações das LLNs, nas quais a maioria dos cenários IoT estão inseridos. Contudo, desde a primeira proposta até hoje, vários estudos têm indicado limitações e falhas no protocolo RPL [13] [16] [20] [21] [22] [23]. Como alternativa às principais limitações do RPL, que incluem o alto uso de memória e elevada complexidade, o protocolo LOADng foi desenvolvido. A solução proposta, embora apresente um núcleo leve capaz de ser executado por dispositivos com extremas limitações de hardware, ainda sofre com a ausência de suporte a importantes recursos requeridos pela maioria das aplicações de IoT.

A implementação padrão do LOADng pode suportar bem o envio de dados P2P, mas não oferece um suporte eficiente aos padrões de tráfego P2MP e MP2P. Além disso, devido a sua simplicidade, o LOADng não apresenta nenhum mecanismo para suportar dispositivos móveis de forma confiável. Assim, a adoção do LOADng em cenários de IoT móvel pode requerer uma grande sobrecarga de controle que implica na redução da eficiência energética e QoS da rede. Além disso, considerando que o LOADng funciona de forma reativa, o elevado número de mensagens de controle necessárias durante as etapas de construção de rotas é uma questão que deve ser considerada para a adoção do protocolo [11]. Por fim, em uma visão geral dos cenários IoT, a integração de dispositivos com diferentes tecnologias de comunicação numa única rede

também representa um aspecto essencial para os protocolos de encaminhamento atuais e não apenas para o LOADng.

Objetivos de Investigação

O principal objetivo desta tese inclui o estudo do comportamento dos protocolos de encaminhamento disponíveis para redes IoT e, considerando suas atuais limitações, propor novas abordagens visando melhorar o desempenho da rede e atender os requisitos de encaminhamento das aplicações IoT. Para alcançar este objetivo principal, foram definidos os seguintes objetivos parciais:

- Realizar uma revisão sistemática do Estado da Arte para investigar os protocolos de encaminhamento mais relevantes para redes IoT existentes na literatura;
- Identificar os principais pontos fortes e limitações das abordagens de encaminhamento mais relevantes para LLNs;
- Desenvolver uma nova solução de encaminhamento para prover um suporte confiável para redes IoT heterogêneas que integram dispositivos com diferentes tecnologias de comunicação;
- Propor um novo mecanismo capaz de aumentar a autonomia e capacidade de auto adaptação do LOADng em cenários IoT heterogêneos;
- Propor uma solução para melhorar o suporte à mobilidade do LOADng em redes IoT móveis;
- Criar um novo mecanismo para permitir ao LOADng suportar os padrões de tráfego P2MP and MP2P de forma eficiente e com baixa sobrecarga;
- Avaliar o desempenho das soluções de encaminhamento propostas em comparação com as abordagens mais relevantes existenes na literatura.

Principais Contribuições

A primeira contribuição desta tese é uma revisão sistemática do estado da arte com o objetivo de identificar as principais soluções de encaminhamento para redes IoT. Inicialmente, o trabalho introduz uma visão geral dos documentos de recomendação apresentados pela IETF indicando os requisitos de encaminhamento de vários grupos de aplicações das LNNs. Em seguida, o estudo descreve os dois principais protocolos de encaminhamento para redes IoT (RPL e LOADng) e as melhorias mais relevantes apresentadas para eles na literatura relacionada. O trabalho realizado identificou que as soluções atuais são projetadas para cobrir as limitações e melhorar estes dois protocolos principais. Contudo, estas novas soluções apresentam continuamente necessidades e restrições que tornam suas adoções limitadas em cenários IoT. Esta contribuição é apresentada no segundo capítulo da tese e foi publicada na revista *Sensors Journal* (MDPI, Suíça) [24].

A segunda contribuição é a proposta de um *framework* para melhorar o desempenho dos protocolos de encaminhamento para redes IoT heterogêneas, nas quais dispositivos com diferentes tecnologias de comunicação, especificamente, o IEEE 802.15.4 e RFID, podem fazer parte do cenário da aplicação. A contribuição propõe o uso de sistemas *fuzzy* para melhorar o

mecanismo de classificação de rotas do protocolo de encaminhamento e melhorar o processo de leitura de etiquetas RFID. Um algoritmo de otimização por colônia de formigas é adotado para otimizar a base de conhecimento do sistema *fuzzy*. Concluiu-se que, para os cenários estudados, o *framework* proposto foi capaz de aumentar a QoS da rede e melhorar a eficiência e velocidade do processo de leitura das etiquetas RFID. Esta contribuição é apresentada no Capítulo 3 e foi publicada na revista *Journal of Network and Computer Applications* (JNCA), da Elsevier [25].

A terceira contribuição é apresentada no Capítulo 4 e foi publicada na edição especial intitulada "*Sensor Systems for Internet of Things*" da revista *Sensors Journal* (MDPI, Suíça) [26]. O trabalho introduz um novo mecanismo de descoberta de rotas para o LOADng, permitindo que os nós da rede com recursos limitados possam encontrar *gateways* para enviar pacotes de dados para a Internet. A solução proposta é complementada por um mecanismo de *cache* para as entradas da tabela de encaminhamento e um novo código para as mensagens de controle de erro. Concluiu-se que a adoção da solução proposta pode contribuir para reduzir a sobrecarga de controle da rede, aumentar a confiabilidade da entrega de pacotes e melhorar a eficiência energética.

A quarta contribuição apresenta um mecanismo para melhorar o desempenho do protocolo LOADng em cenários IoT móveis. A proposta aprimora os processos de descoberta de rotas do LOADng para permitir que os nós possam armazenar dados importantes sobre a disponibilidade dos seus vizinhos. Os nós utilizam os dados armazenados para mapear as mudanças na topologia da rede provocada pela mobilidade dos dispositivos e, assim, reduzir o encaminhamento de pacotes de dados por meio de caminhos interrompidos. Além disso, o mecanismo proposto pode aproveitar o movimento dos nós para encurtar caminhos previamente conhecidos e reduzir o número de transmissões necessárias para entregar os pacotes. Por fim, também é introduzida uma nova métrica de encaminhamento para permitir que o LOADng possa criar rotas baseado na confiabilidade das ligações e na distância entre os nós. Quando comparada a outras soluções, a proposta apresentada mostrou-se capaz de oferecer maior eficiência para várias métricas relacionadas a QoS e consumo de energia em diferentes cenários. Esta contribuição é exposta, em detalhe, no quinto capítulo da tese e foi foi publicada na revista *IEEE Access* [27].

Por fim, a última contribuição desta tese introduz recursos de suporte a transmissão *multicast* (P2MP) no protocolo LOADng. A proposta permite que o LOADng crie uma árvore de encaminhamento *multicast* para enviar pacotes de dados de um dispositivo central para um grupo específico de nós que anteriormente entraram num grupo *multicast*. Durante o processo de construção da árvore, a solução proposta usa mecanismos para reduzir a sobrecarga de mensagens de controlo, garantir a entrega de mensagens de respostas essenciais e diminuir o número de entradas nas tabelas de encaminhamento. Adicionalmente, a árvore de encaminhamento construída pode também ser usada para enviar pacotes de dados dos nós para o dispositivo central, reduzindo a necessidade de novos processos de descoberta de rotas e possibilitando o suporte aos tráfegos P2MP e MP2P. Por meio de experimentos realizados em um protótipo real, concluiu-se que a solução proposta pode adicionar novas funcionalidades ao LOADng de forma eficiente e torná-lo mais apropriado para cenários IoT. Esta contribuição é apresentada no Capítulo 6 e foi publicada na revista *Measurement*, da Elsevier [28].

Principais Conclusões

Esta tese abordou os desafios dos protocolos de encaminhamento em redes IoT. No decorrer do trabalho, foram identificadas várias limitações das atuais soluções de encaminhamento para LLNs e apresentou-se melhorias para superar ou reduzir o impacto desses problemas no desempenho da rede em cenários IoT.

No segundo capítulo deste trabalho, um estudo revisando o Estado da Arte de soluções de encaminhamento para LLNs adotados em cenários IoT foi apresentado. O trabalho conduzido estudou, principalmente, as melhorias introduzidas para os protocolos RPL e LOADng. Ao todo, mais de 30 propostas foram amplamente estudadas e suas qualidades e limitações foram destacadas individualmente. Concluiu-se que a maioria das propostas atuais buscam resolver problemas relacionados com as limitações dos protocolos base, nomeadamente suporte eficiente a mobilidade e padrões de tráfego P2P e P2MP. Adicionalmente, algumas propostas também buscam melhorar o desempenho da rede por meio de aprimoramentos para aumentar a QoS e eficiência energética. Por fim, o trabalho conduzido identificou importantes questões em aberto e destacou diretrizes para o desenvolvimento de novas propostas de encaminhamento para LLNs, concluindo que as atuais propostas podem não atender completamente muitos requisitos das aplicações IoT.

No Capítulo 3 introduziu-se um *framework* especialmente desenvolvido para redes IoT heterogêneas compostas por dispositivos com interface de comunicação IEEE 802.15.4 e elementos RFID. A proposta, que visa integrar os diferentes dispositivos em uma única rede, foi composta por dois elementos. O primeiro introduziu um novo algoritmo de classificação de rotas baseado em sistemas *fuzzy*, enquanto o segundo apresentou um protocolo anti-colisão para leitura de etiquetas RFID. Por meio das experimentações realizados, concluiu-se que nos cenários em que o *framework* proposto foi utilizado, a rede apresentou um desempenho melhor quando comparado com os cenários que não faziam uso da proposta. O conjunto dos mecanismos introduzidos aumentaram a taxa de sucesso da leitura de etiquetas RFID em 25%, sendo este processo realizado 115% mais rápido quando comparado com o protocolo anti-colisão RFID padrão. Além disso, os resultados obtidos expuseram que o uso do *framework* proposto pôde reduzir a taxa de perda de pacotes em até 75% e aumentou o balanceamento de carga da rede em até 55%.

O quarto capítulo desta tese apresentou o LOADng-IoT, uma versão melhorada do LOADng para permitir que os nós da rede possam buscar rotas para dispositivos ligados à Internet de forma autónoma, sem a necessidade de predefinição de um *gateway*. Tal funcionalidade é fornecida por um novo mecanismo de descoberta de caminhos que, amparado por um sistema de *cache* de rotas, pode reduzir a sobrecarga de mensagens de controlo para a criação dos caminhos. A proposta também introduz um novo código de erro para evitar que pacotes de dados sejam encaminhadas para nós ligados à Internet temporariamente impossibilitados de efetuar o correto tratamento dos pacotes. O LOADng-IoT foi comparado com o LOADng padrão e o LOADng-SmartRREQ em topologias em grelha, aleatórias e com nós móveis. Pelos resultados obtidos para os cenários estudados concluiu-se que o LOADng-IoT foi capaz de reduzir a sobrecarga de mensagens de controlo, reduzir o atraso da entrega de pacotes de dados, aumentar a taxa de entrega de pacotes e melhorar a eficiência energética da rede quando comparado com as demais soluções estudadas.

Os estudos conduzidos no decorrer do quarto capítulo da tese, além de apresentar o LOADng-IoT, expuseram as limitações do LOADng em redes IoT com suporte à mobilidade. Assim, para reduzir o impacto da movimentação dos dispositivos no desempenho da rede, o Capítulo 5 apresentou o LOADng-IoT-Mob. A nova proposta adapta os processos de descoberta de rotas herdados do LOADng-IoT, *SmartRREQ* e *Expanding Ring* para melhor gerir a tabela de encaminhamento dos nós e evitar o uso de rotas que tenham sido interrompidas pelo movimento dos dispositivos. A proposta também permite que os caminhos já existentes na tabela de encaminhamento possam ser encurtados caso os nós da rota se aproximem. Por fim, o LOADng-IoT-Mob também introduz uma nova métrica de encaminhamento buscando melhor definir a qualidade das ligações entre os nós da rede e criar caminhos mais confiáveis. O desempenho da solução proposta foi estudada em cenários IoT nos quais se variou a quantidade de nós na rede e a velocidade máxima de movimentação dos dispositivos móveis. Os resultados obtidos mostraram que, para os cenários estudados, o LOADng-IoT-Mob foi capaz de alcançar melhor desempenho para métricas relacionadas com a latência, taxa de entrega de pacotes, consumo de energia e sobrecarga de mensagens de controle. Adicionalmente, concluiu-se que a variação da velocidade máxima dos dispositivos, quando variada entre 1m/s e 9m/s, não foi capaz de impactar significativamente no desempenho dos protocolos estudados. Por fim, com relação ao uso de memória, o LOADng-IoT-Mob proposto requereu um incremento de 10 % no uso de memória flash e 1,55 % no uso de memória RAM, quando comparado com a implementação padrão do LOADng. Contudo, os benefícios obtidos pela proposta, que pode chegar a dobrar o desempenho da rede, podem compensar a memória extra requerida.

O Capítulo 6 introduziu uma melhoria que permite o protocolo LOADng suportar o envio de pacotes de dados *multicast* (P2MP). A proposta, denominada *Multicast* LOADng (M-LOADng), introduz um novo processo de descoberta de rotas dedicado à criação de uma árvore de encaminhamento *multicast*. Durante este processo, que é iniciado por um dispositivo central que deseja enviar dados em *multicast*, os nós da rede podem decidir juntar-se a um grupo *multicast* para formarem uma árvore que, posteriormente, é usada para o envio de pacotes de dados *multicast*. Adicionalmente, a árvore de encaminhamento gerada pode ser utilizada para o envio de dados dos nós para o dispositivo central que iniciou o processo de construção. Assim, a proposta possibilita não somente o tráfego de dados *multicast* (P2MP) mas também facilita o envio de pacotes MP2P. A proposta também introduz mecanismos que asseguram a entrega de mensagens de controle de resposta e, quando possível, evitam o caminho destas buscando reduzir a sobrecarga da rede. Para melhor atender os requisitos de diferentes aplicações IoT, o M-LOADng possibilita a transmissão de pacotes *multicast* em três modos: *unicast*, *broadcast* e *mixed*. Através das experimentações realizadas num protótipo real, e com base nos resultados obtidos, concluiu-se que o M-LOADng é capaz de aumentar a confiabilidade e estabilidade do desempenho da rede para métricas relacionadas com a eficiência energética, QoS e uso de memória. Adicionalmente, constatou-se que o M-LOADng usado em modo de transmissão *unicast* obteve melhores resultados para a métrica de taxas de perda de pacotes, enquanto a adoção do modo de transmissão *broadcast* levou a uma maior redução da latência. O uso do modo *mixed* apresentou desempenho balanceado entre os dois outros modos.

A proposta principal desta tese e os objetivos definidos foram atendidos com sucesso. Com base nas diferentes limitações e fraquezas das atuais soluções de encaminhamento mais relevantes para redes IoT, novas contribuições foram propostas. Especificamente, as novas soluções foram introduzidas para promover e melhorar o suporte à mobilidade, heterogenei-

dade de dispositivos e envio de pacotes de dados *multicast* (P2MP). As abordagens propostas permitiram aumentos de desempenho significativos para diferentes métricas, tais como a taxa de entrega de pacotes, latência ponta-a-ponta, eficiência energética, sobrecarga de mensagens de controlo e uso de memória. Assim, é possível afirmar que as contribuições apresentadas nesta tese podem levar os protocolos de encaminhamento para LLNs a um nível de confiabilidade e eficiência mais elevado, tornando-os mais apropriados e capazes de satisfazer os requisitos das aplicações IoT.

Perspectivas de Trabalhos Futuros

Para concluir esta tese, são sugeridas as seguintes direções de investigações futuras que resultaram do trabalho desenvolvido:

- Realizar a otimização das soluções propostas para as ajustar melhor aos limitados recursos de memória dos dispositivos presentes em cenários IoT;
- Considerar os requisitos das aplicações IoT relacionados com segurança buscando prover soluções eficientes e de baixa compexidade ao nível de encaminhamento;
- Desenvolver soluções de encaminhamento inteligentes e sensíveis ao contexto para atender as necessidades de aplicações IoT em tempo real;
- Propor o uso de técnicas confiáveis de predição de movimento para ajudar os protocolos de encaminhamento a suportarem melhor as mudanças de topologia provocadas pela mobilidade dos dipositivos IoT.

Referências

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] A. Bosche, D. Crawford, D. Jackson, M. Schallehn, and C. Schorling, "Unlocking opportunities in the internet of things," Bain & Company, Tech. Rep., 2018.

[3] B. Nour, K. Sharif, F. Li, S. Biswas, H. Moungla, M. Guizani, and Y. Wang, "A survey of internet of things communication using icn: A use case perspective," *Computer Communications*, vol. 142-143, pp. 95–123, 2019.

[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[5] J. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis, "Connecting low-power and lossy networks to the internet," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 96–101, April 2011.

[6] K. Gama, L. Touseau, and D. Donsez, "Combining heterogeneous service technologies for building an internet of things middleware," *Computer Communications*, vol. 35, no. 4, pp. 405–417, 2012.

[7] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. Mccann, and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," *IEEE Wireless Communications*, vol. 20, no. 6, pp. 91–98, December 2013.

[8] I. E. d. B. Filho, I. Silva, and C. M. D. Viegas, "An effective extension of anti-collision protocol for rfid in the industrial internet of things (iiot)," *Sensors*, vol. 18, no. 12, 2018.

[9] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of things applications: A systematic review," *Computer Networks*, vol. 148, pp. 241–261, 2019.

[10] B. P. Santos, O. Goussevskaia, L. F. Vieira, M. A. Vieira, and A. A. Loureiro, "Mobile matrix: Routing under mobility in iot, iomt, and social iot," *Ad Hoc Networks*, vol. 78, pp. 84–98, 2018.

[11] J. Tripathi, J. C. de Oliveira, and J. Vasseur, "Proactive versus reactive routing in low power and lossy networks: Performance analysis and scalability improvements," *Ad Hoc Networks*, vol. 23, pp. 121–144, 2014.

[12] R. Alexander, A. Brandt, J. Vasseur, J. Hui, K. Pister, P. Thubert, P. Levis, R. Struik, R. Kelsey, and T. Winter, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, Mar. 2012.

[13] O. Iova, P. Picco, T. Istomin, and C. Kiraly, "Rpl: The routing standard for the internet of things... or is it?" *IEEE Communications Magazine*, vol. 54, no. 12, pp. 16–22, December 2016.

[14] M. Goyal, E. Baccelli, M. Philipp, A. Brandt, and J. Martocci, "Reactive Discovery of Point-to-Point Routes in Low-Power and Lossy Networks," RFC 6997, Aug. 2013.

[15] J. Hui and R. Kelsey, "Multicast Protocol for Low-Power and Lossy Networks (MPL)," RFC 7731, Feb. 2016.

[16] T. Clausen, U. Herberg, and M. Philipp, "A critical evaluation of the ipv6 routing protocol for low power and lossy networks (rpl)," in *2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2011, pp. 365–372.

[17] H. S. Kim, J. Ko, D. E. Culler, and J. Paek, "Challenging the ipv6 routing protocol for low-power and lossy networks (rpl): A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2502–2525, Fourthquarter 2017.

[18] T. Clausen, J. Yi, A. Niktash, Y. Igarashi, H. Satoh, U. Herberg, C. Lavenu, T. Lys, and J. Dean, "The lightweight on-demand ad hoc distance-vector routing protocol-next generation (loadng)," Working Draft, IETF Secretariat, Internet-Draft draft-clausen-lln-loadng-15.txt, 2016.

[19] T. Clausen, J. Yi, and U. Herberg, "Lightweight on-demand ad hoc distance-vector routing - next generation (loadng): Protocol, extension, and applicability," *Computer Networks*, vol. 126, pp. 125–140, 2017.

[20] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek, "Dualmop-rpl: Supporting multiple modes of downward routing in a single rpl network," *ACM Transactions on Sensor-Networks (TOSN)*, vol. 11, no. 2, pp. 39:1–39:20, Mar. 2015.

[21] H. Fotouhi, D. Moreira, and M. Alves, "mrpl: Boosting mobility in the internet of things," *Ad Hoc Networks*, vol. 26, pp. 17–35, 2015.

[22] E. Borgia, "The internet of things vision: Key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.

[23] D. Airehrour, J. Gutierrez, and S. K. Ray, "Secure routing for internet of things: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 198–213, 2016.

[24] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, J. Al-Muhtadi, and V. Korotaev, "Routing protocols for low power and lossy networks in internet of things applications," *Sensors*, vol. 19, no. 9, 2019.

[25] J. V. Sobral, J. J. Rodrigues, R. A. Rabelo, J. C. L. Filho, N. Sousa, H. S. Araujo, and R. H. Filho, "A framework for enhancing the performance of internet of things applications based on rfid and wsns," *Journal of Network and Computer Applications*, vol. 107, pp. 56–68, 2018.

[26] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, K. Saleem, and V. Furtado, "Loadng-iot: An enhanced routing protocol for internet of things applications over low power networks," *Sensors*, vol. 19, no. 1, 2019.

[27] J. V. Sobral, J. J. P. Rodrigues, R. A. Rabêlo, K. Saleem, and S. A. Kozlov, "Improving the performance of loadng routing protocol in mobile iot scenarios," *IEEE Access*, vol. 7, pp. 107 032–107 046, 2019.

[28] J. V. Sobral, J. J. Rodrigues, R. A. Rabêlo, and J. Al-Muhtadi, "Multicast improvement for loadng in internet of things networks," *Measurement*, vol. 148, p. 106931, 2019.

[29] J. V. V. Sobral, J. J. P. C. Rodrigues, N. Kumar, C. Zhu, and R. W. Ahmad, "Performance evaluation of routing metrics in the loadng routing protocol," *Journal of Communications Software and Systems*, vol. 13, no. 2, pp. 87–95, 2017.

[30] N. Sousa, J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, and P. Solic, "Eraof: A new rpl protocol objective function for internet of things applications," in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, July 2017, pp. 1–5.

[31] H. d. S. Araújo, R. H. Filho, J. J. P. C. Rodrigues, R. d. A. L. Rabelo, N. d. C. Sousa, J. C. C. L. S. Filho, and J. V. V. Sobral, "A proposal for iot dynamic routes selection based on contextual information," *Sensors*, vol. 18, no. 2, 2018.

Abstract

The Internet of Things (IoT) proposes a disruptive communication paradigm that allows smart objects to exchange data among themselves to reach a common goal. IoT application scenarios are multiple and can range from a simple smart home lighting system to fully controlled automated manufacturing chains. In the majority of IoT deployments, things are equipped with small devices that can suffer from severe hardware and energy restrictions that are responsible for performing data processing and wireless communication tasks. Thus, due to their features, communication networks that are used by these devices are generally categorized as Low Power and Lossy Networks (LLNs).

The considerable variation in IoT applications represents a critical issue to LLN networks, which should offer support to different requirements as well as keeping reasonable quality-of-service (QoS) levels. Based on this challenge, routing protocols represent a key issue in IoT scenarios deployment. Routing protocols are responsible for creating paths among devices and their interactions. Hence, network performance and features are highly dependent on protocol behavior. Also, based on the adopted protocol, the support for some specific requirements of IoT applications may or may not be provided. Thus, a routing protocol should be projected to attend the needs of the applications considering the limitations of the device that will execute them.

Looking to attend the demand of routing protocols for LLNs and, consequently, for IoT networks, the Internet Engineering Task Force (IETF) has designed and standardized the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL). This protocol, although being robust and offering features to fulfill the need of several applications, still presents several faults and weaknesses (mainly related to its high complexity and memory requirement), which limits its adoption in IoT scenarios. An alternative to RPL, the Lightweight On-demand Ad Hoc Distance-vector Routing Protocol – Next Generation (LOADng) has emerged as a less complicated routing solution for LLNs. However, the cost of its simplicity is paid for with the absence of adequate support for a critical set of features required for many IoT environments. Thus, based on the challenging open issues related to routing in IoT networks, this thesis aims to study and propose contributions to better attend the network requirements of IoT scenarios. A comprehensive survey, reviewing state-of-the-art routing protocols adopted for IoT, identified the strengths and weaknesses of current solutions available in the literature. Based on the identified limitations, a set of improvements is designed to overcome these issues and enhance IoT network performance. The novel solutions are proposed to include reliable and efficient support to attend the needs of IoT applications, such as mobility, heterogeneity, and different traffic patterns. Moreover, mechanisms to improve the network performance in IoT scenarios, which integrate devices with different communication technologies, are introduced.

The studies conducted to assess the performance of the proposed solutions showed the high potential of the proposed solutions. When the approaches presented in this thesis were compared with others available in the literature, they presented very promising results considering the metrics related to the Quality of Service (QoS), network and energy efficiency, and memory usage as well as adding new features to the base protocols. Hence, it is believed that the proposed improvements contribute to the state-of-the-art of routing solutions for IoT networks, increasing the performance and adoption of enhanced protocols.

Keywords

Internet of Things, Routing Protocols, Low power and Lossy Networks, RPL Routing Protocol, LOADng Routing Protocol, Wireless Sensor Networks, Mobile Ad hoc Networks, Radio Frequency Identification, RFID Anti-collision Protocols, Directed Diffusion, Fuzzy Systems, Ant Colony Optimization, Ant System, Network Performance, Quality of Service.

Contents

List of Figures

Chapter 2

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

Chapter 3

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

Chapter 4

LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks

Chapter 5

Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios

Chapter 6

Multicast Improvement for LOADng in Internet of Things Networks

List of Figures

List of Tables

Chapter 2

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

Chapter 3

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

Chapter 4

LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks

Chapter 5

Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios

Chapter 6

Multicast Improvement for LOADng in Internet of Things Networks

Acronyms

6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ACK	Acknowledgment
ACO	Ant Colony Optimization
AEC	Average Energy Consumption
AES	Average Energy Spent per Delivered Data Bit
A-LLN	Agricultural LLN
AMI	Advanced Metering Infrastructure
AN	Associated Node
AODV	Ad-hoc On-demand Distance Vector
AODV-RPL	AODV routing-based RPL
AP	Access Point
ARSSI	Average RSSI
BA-LLN	Building Automation LLN
BMRF	Bidirectional Multicast RPL Forwarding
BMS	Building Management Systems
BRPL	Backpressure RPL
BWSN	Biomedical Wireless Sensor Networks
C1-G2	EPC Global UHF Class-1 Generation-2 Standard
CAOF	Context-Aware Objective Function
CARF	Context-Aware Routing Metric
CCI	Channel Check Interval
CCR	Check Channel Rate
CI	Computational Intelligence
CLRPL	Context-aware and Load-balancing RPL
CMO	Control Message Overhead per Delivered Data Message
COB	Control Bit Overhead per Delivered Data Bit
CODB	Control Bit Overhead per Delivered Data Bit
Co-RPL	Corona RPL
CPU	Central Processing Unit
CSMA	Carrier Sense Multiple Access
CV	Coefficient of Variation
DAG	Directed Acyclic Graphs
DAO	Destination Advertisement Object
DD	Directed Diffusion
DIM	Delay Iterative Method
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DODAG	Destination-Oriented DAG
DQCA-OF	Delivery Quality- and Context-Aware Objective Function
DSR	Dynamic Mobile Ad hoc Network On-demand Routing
DT-RPL	Diverse Traffic RPL
DualMOP-RPL	Dual Mode of Operation RPL
DYMO	Dynamic Source Routing
DYMO-Low	Dynamic MANET On-demand for 6LoWPAN

EAOF	Energy-Aware Objective Function
EKF	Extended Kalman Filter
EKF-MRPL	EKF for Mobile RPL
ELT	Expected Lifetime
EMA-RPL	Energy-efficient and Mobility-aware RPL
EPC	Electronic Product Code
ERAOF	Energy Efficient and Path Reliability Aware Objective Function
ERGID	Emergency Response IoT based on Global Information Decision
ER-RPL	Energy-efficient Region-based Routing Protocol
ESB	Energy Spent per Delivered Data Bit
ESDB	Energy Spent per Delivered Data Bit
ESMRF	Enhanced SMRF for IPv6-based LLN
ETX	Expected Transmission count
EUI	Extended Unique Identifier
FL	Fuzzy Logic
FQA	Fuzzy Q-Algoritm
FSBRC	Fuzzy System-Based Route Classifier
FSELC	Fully Simplified Exponential Lifetime Cost
FUZZY OF	Fuzzy-based Objective Function
GOAFR	Greedy Other Adaptive Face Routing
GPS	Global Positioning System
HA-LLN	Home Automation LLN
HC	Hop Count
Hi-Low	Hierarchical routing for 6LoWPAN
HYDRO	Hybrid Routing protocol for LLNs
IC	Internet-Connected
ID	Identifier
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IN	Internet-connected Node
IoT	Internet of Things
IP	Internet Protocol
IRC	Internet Route Cache
ITU-T	International Telecommunication Union - Telecommunication Standardization Sector
L^2AM	Lifetime and Latency Aggregatable Metric
LBR	LLN Border Router
LCO	Link Color Object
LLN	Low power and Lossy Networks
LOAD	6LoWPAN Ad hoc On-demand Distance Vector Routing
LOADng	Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation
LOADng-CTP	LOADng Collection Tree Extension
LOADng-IoT	LOADng for IoT
LOADng-IoT-Mob	LOADng for IoT with Mobility
LPM	Low Power Mode
LQI	Link Quality Indicator

LQI_WL	Link Quality Indicator Weaklinks
LRRE	Live Routes - Residual Energy
LTE	Long-Term Evolution
M_RREP	Multicast RREP
M_RREQ	Mutlicast RREQ
MAC	Media Access Control
MANET	Mobile Ad-hoc Networks
MAX-LQI	Maximum LQI
MBCR	Minimum Battery Cost Routing
MCU	Microcontroller Unit
M-LOADng	Multicast LOADng
MMBCR	Min-Max Battery Cost Routing
MN	Mobile Node
MNB	Maximum Number of Broadcasts
MO	Measurement Objects
mod-RPL	modified RPL
MOP	Mode of Operation
MP2P	Multipoint-to-point
MPL	Multicast Protocol for LLN
MRHOF	Minimum Rank with Hysteresis Objective Function
MRO	Message Request Object
mRPL	mobility-enabled RPL
NDM	Neighbor Disjoint Multipath
NH	Number of Hops
NLB	Network Load Balance
OF	Objective Function
OF0	Objective Function Zero
OF-FL	Objective Function Fuzzy Logic
OS	Operational System
OSPF	Open Shortest Path First
P2MP	Point-to-multipoint
P2P	Point-to-point
P2P-RDO	P2P Route Discovery Option
P2P-RPL	Reactive Discovery of Point-to-Point Routes in LLN
PAOF	Parent-Aware Objective Function
PDR	Packet Delivery Ratio
PLC	Powerline Communications
PLL	Percentage of Packets with Low Latency
PLR	Packet Loss Rate
QoS	Quality of Service
QoS_RPL	Quality of Service RPL
QSR	Query Success Rate
RAM	Random Access Memory
RC	Region Code
RDC	Radio Duty Cycling
RE	Remaining Energy
REL	Routing by Energy and Link Quality

REPC	Residual Energy Probability Choice
RERR	Route Error
RFC	Request for Comments
RFID	Radio Frequency Identification
RFO	Region Formation Object
RIP	Routing Information Protocol
RN	Reference Nodes
RoLL	Routing over Low Power and Lossy Networks
RPL	IPv6 Routing Protocol for Low Power and Lossy Networks
RREP	Route Reply
RREP_ACK	Route Reply Acknowledgment
RREQ	Route Request
RRM	Reply Retry Mechanism
RS	Reader-Sensor
RS-DR	RS with Dual Radio
RSM	Reply Suppressing Mechanism
RSN	RFID-Sensor Networks
RS-SDR	RS with Software-Defined Radio
RSSI	Received Signal Strength Indication
SA-A-WSN	Situation-Aware Adaptation Approach for Energy Conservation in WSN
SCAOF	Scalable Context-Aware Objective Function
SEEOF	Smart Energy Efficient Objective Function
SG	Smart Grid
SH	Smart Home
SMRF	Stateless Multicast RPL Forwarding
SoC	System-on-Chip
SPOF	Single Point of Failure
TIS	Tag Idenficiation Speed
UDGM	Unit Disk Graph Model
U-LLN	Urban LLN
WF	Weighed Forwarding
WG	Working Group
WSN	Wireless Sensor Networks

Chapter 1

Introduction

This thesis addresses the topic of routing protocols for Low Power and Lossy Networks (LLNs) in Internet of Things (IoT) scenarios. In this chapter, the focus and scope of this research are described. The problem definition, research objects, main contribution, thesis statement, and the organization of the thesis are presented.

1.1 Focus and Scope

The Internet of Things (IoT) paradigm has emerged in the last decade attracting the attention from both industry and the research community. The IoT concept proposes the possibility of objects ("things") communicating pervasively among themselves to provide facilities and automation in many application scenarios [1]. According to the Bain & Company [2], it is expected that IoT market will grow from the $235 billion spent in 2017 to around $ 520 billion in 2021. Until 2021, in comparison to 2017, the report reveals that the investments into the "System Integration" and "Network" areas should grow around 15% and 40%, respectively.

IoT scenarios are performed by smart devices that can be applied to perform daily activities, making intelligent decisions to make different aspects of people's lives easier [3]. Currently, the most significant set of IoT applications are focused on environmental monitoring, smart cities, healthcare, home automation, and industry [4]. Due to the particularities of several application scenarios, IoT networks can form complex structures composed by devices with different capacities. Network nodes can diverge in hardware features, manufacturer, and within communication technologies.

In general, IoT devices (the "things") adopt the use of micro-controller units (MCU) or System-on-Chip (SoC) solutions to perform data processing and wireless communications tasks. Thus, these nodes tend to have a severe hardware limitation related to low processing and memory capacities. Further, depending on the application scenario, the network devices can have mobility features and suffer from energy constraints (being battery equipped). Hence, due to the features of the nodes that compose them, IoT networks are commonly classified as Low Power and Lossy Networks (LNNs) [5]. IoT scenarios can integrate devices using different communication technologies in a shared network to allow the emergence of disruptive applications [6]. In this context, a prominent technology, frequently attributed for providing ubiquitous and pervasive features for IoT scenarios, is Radio Frequency Identification (RFID) [7]. RFID systems offer low-cost, highly reliable, and a fast, unique identification capacity to IoT scenarios, allowing the application for tracking and monitoring the presence of devices in the environment [8].

Regardless of the utilization scenarios and available technologies in a network, IoT applications aim to provide appropriated services to users. They should require fulfilling several

Quality of Service (QoS) metrics related to energy efficiency, latency, reliability, availability, and security [9]. Thus, based on these features, the choice of routing protocol becomes a key factor to allow networks to attend QoS requirements of the IoT applications [10].

Routing protocols are responsible for creating and maintain the path among the network nodes and further perform data packet forwarding. Hence, the network's performance becomes highly dependent on routing protocol behavior [11]. Based on the importance of this element, the Internet Engineering Task Force (IETF) has charted the Routing over Low Power and Lossy Networks (RoLL) working group (WG) to study the requirements of LLN applications and design a new routing solution to fulfill them. Thus, in an initial effort, the RoLL WG presented the routing requirement for a set of LLN applications related to industrial, home automation, building automation, and urban scenarios. Then, based on the identified needs, the RoLL designed the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [12], which was defined as the standard routing protocol for LLNs. RPL proposes a proactive routing solution in which a central device (also known as a border router) is responsible for constructing a routing tree optimized for the collection of data sent from the nodes to a central device.

Although RPL has been defined as a standard for LLNs and, then, the "standard" routing protocol for IoT networks [13], several studies have contested the protocol's efficiency and its capacity to attend the requirements of IoT applications. Iova *et al.* [13] conducted a critical study to evaluate the behavior of RPL in IoT environments. The authors identified several issues mainly related to limited support to point-to-point (P2P) and point-to-multipoint (P2MP) unreliable mobility support and a high memory footprint that limits the device's heterogeneity and the network's scalability. Further, the authors highlighted that RPL was designed looking mainly at the multipoint-to-point (MP2P) traffic pattern in which the data packets flow from the nodes to a central device. However, the most disruptive IoT applications can require efficient and reliable support for both P2P and P2MP. To better attend these traffics patterns, the RPL-P2P [14] and Multicast Protocol for LLNs (MPL) [15] enhancements were designed, but both approaches require increasing memory usage, which can provoke the emergence of other negative effects within the network's performance. Finally, Iova *et al.* endorsed that the emergence of new routing solutions for IoT networks could surpass the drawbacks of RPL.

Based on the referred RPL weaknesses and several other limitations presented by the protocol discussed in the literature [16] [17], a new less-complex routing solution has emerged as an alternative to the RoLL's solution. The Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation (LOADng) [18]. It presents a reactive routing solution in which the path among the nodes are built on demand. Thus, the constrained resources of LLN devices are only consumed when nodes need to exchange data messages. Different to RPL, LOADng is not a standard routing protocol for LLN but remains "under-draft" in an IETF track. However, highlighting its relevance, LOADng was included in the G3-PLC ITU-I standard for communication in smart grid scenarios [19].

When compared to RPL, the main advantage of LOADng is its lightweight core and reactive behavior, which allows the protocol to better fit the hardware limitation of low power devices and the support for P2P data transmission. In contrast, this on-demand work feature can imply the increase in end-to-end delay, affecting the network QoS. Further, due to its simplified implementation, this protocol does not present any support for P2MP traffic and mobility support

is implicitly attended only by the reactive features of the protocol. Therefore, it is noticed the main routing solutions for LLN may not completely meet the demands of IoT applications. Thus, this thesis addresses the performance study of routing protocols and propose contributions to fulfill the identified gaps in the state-of-the-art of routing for IoT. The following subsection is dedicated to present problem definition.

1.2 Problem Definition

In IoT scenarios, a routing protocol should be able to provide the network's requirements according to the application. Although can change, the needs of IoT applications generally goes through the support of heterogeneous and mobile devices, reasonable QoS levels, different traffic pattern support, and energy efficiency. Thus, considering the applications' requirements and the limited computational and energy resources of the network nodes, the task of routing protocol becomes hardest once it should be light, power efficient, highly reliable, and to demand low processing capacity.

The IETF, through the RoLL WG, proposed the RPL for attending the demands of LLN applications, in which the majority of IoT scenarios are inserted. However, since the first draft proposal, several studies have indicated the limitations and weaknesses of RPL [13] [16] [20] [21] [22] [23]. As an alternative to the main fault of RPL, which includes highly complex and demands a lot of memory usage, the LOADng was designed. This proposal, although presenting a lightweight core that can be executed by devices with extremely restricted hardware, still suffers from the absence of support for important features required by the majority of IoT applications.

The standard LOADng implementation can support peer-to-peer (P2P) data well but does not offer an efficient solution for P2MP and MP2P traffic patterns. Further, due to its simplicity, LOADng does not present any specific mechanism for mobility support. Thus, the adoption of LOADng in mobile IoT scenarios can require a high overhead that implies decreasing the network QoS and energy efficiency. Also, since the protocol works in a reactive way, the high number of control messages required during path construction phases is a problem for the LOADng adoption [11]. Finally, in a general view of IoT scenarios, the integration of devices with different communication technologies in a shared network is also an essential task for the current routing protocols, not only for LOADng.

1.3 Research Objectives

The main objective of this thesis includes the study and behavior analysis of the available routing protocols for IoT networks and, considering their limitations, proposes new approaches aiming to improve network performance and attend to the routing requirements of IoT applications.

To attain this global objective, the following partial objectives have been identified:

- Perform a systematic review of the state-of-the-art of routing protocols for IoT networks to investigate the most relevant approaches;

- Identify the primary strengths and weaknesses of the main routing approaches for low power networks in the related literature;
- Design a new routing solution to provide feasible support for heterogeneous IoT networks that integrates different communication technologies;
- Propose a novel mechanism that can increase the autonomy and capacity of self-adaption of the LOADng in heterogeneous IoT scenarios;
- Propose a solution for mobility support using LOADng in (mobile) IoT networks;
- Create a new mechanism to efficiently support P2MP and MP2P traffic patterns with a low overhead in LOADng;
- Evaluate the performance of the proposed routing solutions in comparison with the most relevant approaches available in the literature.

1.4 Main Contributions

The first contribution of this thesis is a comprehensive systematic review of the state-of-the-art of routing solutions for IoT networks conducted to identify the main approaches. The work initially introduces an overview of the recommendation documents presented by the IETF, indicating the routing requirements for several groups of applications in LLNs. Next, the work describes the current two main routing protocols for IoT networks (RPL and LOADng) and the most relevant improvements introduced to them in the related literature. The work identified that available solutions are designed to cover the limitations and improve these two main protocols. However, these new solutions continuously present requirements or limitations that make their adoption limited for IoT scenarios. This survey was published on *Sensors Journal* (MDPI, Switzerland) [24].

The second contribution is the proposal of a framework to enhance the performance of routing protocols in heterogeneous IoT networks, in which devices with different communication technologies, specifically IEEE 802.15.4 and RFID, can be a part of the environment. The contribution proposes the use of fuzzy systems to improve the route classification mechanism of the routing protocol and enhance the reading process of RFID tag readers. An ant colony optimization (ACO) algorithm is adopted to optimize the knowledge base of the fuzzy system. It was concluded that, for the studied scenarios, the proposed framework was able to increase the network QoS and improve the efficiency and speed of the RFID tags reading process. This contribution was published in the *Journal of Network and Computer Applications* (JNCA), from Elsevier [25].

The third contribution was published in a special issue entitled “Sensor Systems for Internet of Things” from *Sensors Journal* (MDPI, Switzerland) [26]. This work introduces a new route discovery mechanism for LOADng protocol allowing resource-limited network nodes to find gateways to forward data messages to the Internet. The proposed solution is complemented with a new routing table entry caching mechanism and a new code for control error messages. It was concluded that the adoption of the proposed solution can contribute to reduce the network control overhead, increase the packet delivery reliability, and improve energy efficiency.

The fourth contribution of this thesis proposes a mechanism to improve the performance of LOADng in mobile IoT scenarios (providing mobility support). It enhances the LOADng route discovery processes to allow the nodes to store additional important data about the availability of their neighbors. The nodes use the stored data to track the topology changes in the function of the devices' mobility and reduce data messages forwarding through broken paths. Besides, the proposed mechanism can take advantage of the nodes' movements to shorten already known routes and reduce the number of transmissions required to deliver a data message. It also introduces a new routing metric to enable LOADng to create paths based on the reliability of the links and the distance among nodes. In comparison to other relevant solutions on the topic, the introduced proposal showed the ability to offer a more efficient performance in several metrics related to QoS and energy consumption in different scenarios. This contribution was published in the journal *IEEE Access* [27].

Finally, the last contribution introduces a novel multicast (P2MP) support feature to the LOADng protocol. The proposal allows LOADng to create a multicast routing tree to forward data messages from a central device to specific set nodes that joined a multicast group. During the process of tree building, the proposed solution uses mechanisms to reduce the control overhead, ensure the delivery of essential reply control messages, and reduce the number of entries in the routing table. Further, the built routing tree can also be used to send packets from nodes to the sink, reducing the need to perform expensive route creation processes and enable the support for both P2MP and MP2P traffic. The experiments, performed using a real prototype, concluded that the proposed solution can add new features to the LOADng in an efficient way and make the protocol more appropriate for IoT scenarios. This contribution was published in the journal *Measurement*, from Elsevier [28].

1.5 Thesis Statement

The behavior of routing protocols represents a critical factor that can affect the efficiency, reliability, and performance of IoT networks. Further, according to the adopted routing protocol, the requirements of IoT applications may not be entirely attended. Thus, routing solutions should be designed to deliver a feasible performance considering the specific needs of IoT applications regarding Quality of Service, mobility, and different traffic pattern support. The protocols should also consider the heterogeneity and constraints of IoT devices, which commonly suffer from severe hardware and energy constraints. Currently, routing solutions tend to require complex deployments with considerable memory usage that lead to a partial adoption of them in real environments. Aiming to reduce the hardware requirements, novel lightweight protocols have emerged in last years but still suffering from the absence of important features for IoT applications. Offering self-adaption capability, mobility enhanced support, and a wide variety of traffic patterns supported can elevate the lightweight protocols' functionalities and performance delivering more stability, reliability, and efficiency. Thus, the enhancements for the newest less-complex solutions can improve the adoption of these protocols turning them able to attend the demands of the challenged IoT network scenarios.

1.6 Document Organization

This thesis comprises seven chapters, which are organized as follows. The first chapter introduces and focuses on the topic of the study, presents the work's motivation, identifies the problem delimitation, defines the thesis objectives, highlights the main contributions, and includes the thesis statement. The document's organization is also included in this chapter. Except for this, and the concluding chapter, all other chapters are based on papers published in, or submitted to, international journals.

Chapter 2 presents the survey paper entitled "Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications." The paper provides an in-depth review of the state-of-the-art of routing protocols and discusses and points out the strengths and weaknesses of the main routing solutions for IoT networks. This work presents the relevant guidelines and future directions for the design of new routing solutions for IoT.

Chapter 3, entitled "A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs," presents a fuzzy system-based framework for improving the performance of routing protocols in IoT networks. The proposal was designed considering heterogeneous networks and introduces a novel RFID-tag reading algorithm.

In Chapter 4, entitled "LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks," an enhanced version of LOADng is presented. It aims to allow nodes to discover routes to Internet-connected devices efficiently and autonomously.

Chapter 5, entitled "Multicast Improvement for LOADng in Internet of Things Networks", the Multicast LOADng improvement is introduced. The developed mechanisms allow the LOADng to create multicast routing trees and perform the routing of multicast data messages. The proposed solution was implemented in a real testbed and compared with other LOADng improvements in different scenarios.

In Chapter 6 ("Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios"), the LOADng protocol is enhanced for mobility support in IoT networks. The designed solution involves the improvement of route discovery processes and the harnessing of control messages to track topology changes caused by nodes' mobility.

Chapter 7 summarizes the main conclusions of the thesis and suggests future work that can emerge from the conducted work.

In the Appendices are presented three contributions, two published in international journals and one in an international conference. In Appendix A, entitled "Performance Evaluation of Routing Metrics in the LOADng Routing Protocol," a performance evaluation of LOADng routing protocol, considering different routing metrics, is conducted. The study varied several routing metrics in scenarios with MP2P and P2P data traffic. This contribution was published on the *Journal of Communications Software and Systems* from the Croatian Communications and Information Society in cooperation with the University of Split [29].

Appendix B, entitled "ERAOF: A New RPL Protocol Objective Function for Internet of

Things Applications," introduces a new objective function for RPL based on node energy and link quality aiming to optimize the routing process for IoT networks. The proposed solution was compared with the standard IETF objective function in networks with a variable number of nodes and different topologies. This contribution was published on the proceedings of the *2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)* from IEEE [30].

Appendix C ("A Proposal for IoT Dynamic Routes Selection Based on Contextual Information") proposes a set of RPL objective functions that can be adopted according to the context of the IoT application. In the approach, three routing metrics are combined in different ways, including the use of a fuzzy system, to generate five different objective functions. The set of created solutions are compared to the IETF standard approaches through computational simulations. This contribution was published on *Sensors Journal* (MDPI, Switzerland) [31].

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] A. Bosche, D. Crawford, D. Jackson, M. Schallehn, and C. Schorling, "Unlocking opportunities in the internet of things," Bain & Company, Tech. Rep., 2018.

[3] B. Nour, K. Sharif, F. Li, S. Biswas, H. Moungla, M. Guizani, and Y. Wang, "A survey of internet of things communication using icn: A use case perspective," *Computer Communications*, vol. 142-143, pp. 95–123, 2019.

[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[5] J. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis, "Connecting low-power and lossy networks to the internet," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 96–101, April 2011.

[6] K. Gama, L. Touseau, and D. Donsez, "Combining heterogeneous service technologies for building an internet of things middleware," *Computer Communications*, vol. 35, no. 4, pp. 405–417, 2012.

[7] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. Mccann, and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," *IEEE Wireless Communications*, vol. 20, no. 6, pp. 91–98, December 2013.

[8] I. E. d. B. Filho, I. Silva, and C. M. D. Viegas, "An effective extension of anti-collision protocol for rfid in the industrial internet of things (iiot)," *Sensors*, vol. 18, no. 12, 2018.

[9] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of things applications: A systematic review," *Computer Networks*, vol. 148, pp. 241–261, 2019.

[10] B. P. Santos, O. Goussevskaia, L. F. Vieira, M. A. Vieira, and A. A. Loureiro, "Mobile matrix: Routing under mobility in iot, iomt, and social iot," *Ad Hoc Networks*, vol. 78, pp. 84–98, 2018.

[11] J. Tripathi, J. C. de Oliveira, and J. Vasseur, "Proactive versus reactive routing in low power and lossy networks: Performance analysis and scalability improvements," *Ad Hoc Networks*, vol. 23, pp. 121–144, 2014.

[12] R. Alexander, A. Brandt, J. Vasseur, J. Hui, K. Pister, P. Thubert, P. Levis, R. Struik, R. Kelsey, and T. Winter, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, Mar. 2012.

[13] O. Iova, P. Picco, T. Istomin, and C. Kiraly, "Rpl: The routing standard for the internet of things... or is it?" *IEEE Communications Magazine*, vol. 54, no. 12, pp. 16–22, December 2016.

[14] M. Goyal, E. Baccelli, M. Philipp, A. Brandt, and J. Martocci, "Reactive Discovery of Point-to-Point Routes in Low-Power and Lossy Networks," RFC 6997, Aug. 2013.

[15] J. Hui and R. Kelsey, "Multicast Protocol for Low-Power and Lossy Networks (MPL)," RFC 7731, Feb. 2016.

[16] T. Clausen, U. Herberg, and M. Philipp, "A critical evaluation of the ipv6 routing protocol for low power and lossy networks (rpl)," in *2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2011, pp. 365–372.

[17] H. S. Kim, J. Ko, D. E. Culler, and J. Paek, "Challenging the ipv6 routing protocol for low-power and lossy networks (rpl): A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2502–2525, Fourthquarter 2017.

[18] T. Clausen, J. Yi, A. Niktash, Y. Igarashi, H. Satoh, U. Herberg, C. Lavenu, T. Lys, and J. Dean, "The lightweight on-demand ad hoc distance-vector routing protocol-next generation (loadng)," Working Draft, IETF Secretariat, Internet-Draft draft-clausen-lln-loadng-15.txt, 2016.

[19] T. Clausen, J. Yi, and U. Herberg, "Lightweight on-demand ad hoc distance-vector routing - next generation (loadng): Protocol, extension, and applicability," *Computer Networks*, vol. 126, pp. 125–140, 2017.

[20] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek, "Dualmop-rpl: Supporting multiple modes of downward routing in a single rpl network," *ACM Transactions on Sensor-Networks (TOSN)*, vol. 11, no. 2, pp. 39:1–39:20, Mar. 2015.

[21] H. Fotouhi, D. Moreira, and M. Alves, "mrpl: Boosting mobility in the internet of things," *Ad Hoc Networks*, vol. 26, pp. 17–35, 2015.

[22] E. Borgia, "The internet of things vision: Key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.

[23] D. Airehrour, J. Gutierrez, and S. K. Ray, "Secure routing for internet of things: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 198–213, 2016.

[24] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, J. Al-Muhtadi, and V. Korotaev, "Routing protocols for low power and lossy networks in internet of things applications," *Sensors*, vol. 19, no. 9, 2019.

[25] J. V. Sobral, J. J. Rodrigues, R. A. Rabelo, J. C. L. Filho, N. Sousa, H. S. Araujo, and R. H. Filho, "A framework for enhancing the performance of internet of things applications based on rfid and wsns," *Journal of Network and Computer Applications*, vol. 107, pp. 56–68, 2018.

[26] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, K. Saleem, and V. Furtado, "Loadng-iot: An enhanced routing protocol for internet of things applications over low power networks," *Sensors*, vol. 19, no. 1, 2019.

[27] J. V. Sobral, J. J. P. Rodrigues, R. A. Rabêlo, K. Saleem, and S. A. Kozlov, "Improving the performance of loadng routing protocol in mobile iot scenarios," *IEEE Access*, vol. 7, pp. 107 032–107 046, 2019.

[28] J. V. Sobral, J. J. Rodrigues, R. A. Rabêlo, and J. Al-Muhtadi, "Multicast improvement for loadng in internet of things networks," *Measurement*, vol. 148, p. 106931, 2019.

[29] J. V. V. Sobral, J. J. P. C. Rodrigues, N. Kumar, C. Zhu, and R. W. Ahmad, "Performance evaluation of routing metrics in the loadng routing protocol," *Journal of Communications Software and Systems*, vol. 13, no. 2, pp. 87–95, 2017.

[30] N. Sousa, J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, and P. Solic, "Eraof: A new rpl protocol objective function for internet of things applications," in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, July 2017, pp. 1–5.

[31] H. d. S. Araújo, R. H. Filho, J. J. P. C. Rodrigues, R. d. A. L. Rabelo, N. d. C. Sousa, J. C. C. L. S. Filho, and J. V. V. Sobral, "A proposal for iot dynamic routes selection based on contextual information," *Sensors*, vol. 18, no. 2, 2018.

Chapter 2

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

This chapter consists in the following paper:

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Jalal Al-Muhtadi, and Valery Korotaev

Sensors, MDPI, ISSN: 1424-8220, 2019.

DOI: doi.org/10.3390/s19092144

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 3.275

ISI Article Influence Score (2019): 0.530

Journal Ranking (2019): Q1 - 15/64 (Instruments & Instrumentation)

Journal Ranking (2019): Q2 - 77/266 (Engineering, Electrical & Electronic)

sensors

MDPI

Review

Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

José V. V. Sobral [1,2], Joel J. P. C. Rodrigues [1,3,4,5,6,*,†], Ricardo A. L. Rabêlo [4], Jalal Al-Muhtadi [5] and Valery Korotaev [6]

1 Instituto de Telecomunicações, Universidade da Beira Interior, 6201-001 Covilhã, Portugal; jose.sobral@it.ubi.pt
2 Federal Institute of Maranhão (IFMA), São Luís-MA 65010-030, Brazil
3 National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí-MG 37540-000, Brazil
4 Federal University of Piauí, Teresina-PI 64049-550, Brazil; ricardoalr@ufpi.edu.br
5 College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 12372, Saudi Arabia; jalal@ccis.edu.sa
6 ITMO University, St. Petersburg 197101, Russia; korotaev@grv.ifmo.ru
* Correspondence: joeljr@ieee.org; Tel.: +55-35-3471-9200
† Current address: Av. João de Camargo, 510-Centro, Santa Rita do Sapucaí-MG 37540-000, Brazil.

Received: 23 March 2019; Accepted: 5 May 2019; Published: 9 May 2019

Abstract: The emergence of the Internet of Things (IoT) and its applications has taken the attention of several researchers. In an effort to provide interoperability and IPv6 support for the IoT devices, the Internet Engineering Task Force (IETF) proposed the 6LoWPAN stack. However, the particularities and hardware limitations of networks associated with IoT devices lead to several challenges, mainly for routing protocols. On its stack proposal, IETF standardizes the RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) as the routing protocol for Low-power and Lossy Networks (LLNs). RPL is a tree-based proactive routing protocol that creates acyclic graphs among the nodes to allow data exchange. Although widely considered and used by current applications, different recent studies have shown its limitations and drawbacks. Among these, it is possible to highlight the weak support of mobility and P2P traffic, restrictions for multicast transmissions, and lousy adaption for dynamic throughput. Motivated by the presented issues, several new solutions have emerged during recent years. The approaches range from the consideration of different routing metrics to an entirely new solution inspired by other routing protocols. In this context, this work aims to present an extensive survey study about routing solutions for IoT/LLN, not limited to RPL enhancements. In the course of the paper, the routing requirements of LLNs, the initial protocols, and the most recent approaches are presented. The IoT routing enhancements are divided according to its main objectives and then studied individually to point out its most important strengths and weaknesses. Furthermore, as the main contribution, this study presents a comprehensive discussion about the considered approaches, identifying the still remaining open issues and suggesting future directions to be recognized by new proposals.

Keywords: Internet of Things; low-power and lossy network; LOADng; routing protocol; RPL

1. Introduction

In the near future, it is expected that all things will be able communicate among themselves through the Internet. It is easy to perceive this evolution in a time when things are more and more connected to the global computer network. This reason leads to the emergence of a new paradigm called the Internet of Things (IoT) [1]. According to [2], the core idea of the IoT is the pervasive

presence of a variety of objects and things around us that can exchange information to reach a common objective. According to [3], in 2025, Internet nodes will be present in people's daily routines in the form of furniture, papers, televisions, refrigerators, and food packaging.

According to [4], IoT is a multidisciplinary domain that encompasses a vast number of topics, including purely technical challenges and a mix of technical and social tasks, as well as social and enterprise efforts. Ambient and personal care monitoring, industrial planning monitoring, including agriculture, smart environments, and smart cities are examples of IoT applications [5–7].

Wireless Sensor Networks (WSN) are an essential component of the IoT because they enable the development of applications with a capacity to sense several variables related to the environment in which the devices are inserted [8]. WSN is a specific kind of LoWPAN (Low-power Wireless Personal Area Network) composed by nodes equipped with different types of sensors (e.g., temperature, humidity, and luminosity). To begin with, the interoperability between WSN and the Internet was limited due to the absence of IP communication infrastructure. Thus, seeking to solve this problem, several works were dedicated to proposing a structure that could provide the use of IP over LoWPAN [9].

Aiming to provide a standard solution for this problem, in October 2004, IETF (Internet Engineering Task Force) proposed a Working Group (WG) named 6LoWPAN (IPv6 over Low-power Wireless Personal Area Networks). The 6LoWPAN working group's objective was to create ways to enable the use of IPv6 over IEEE 802.15.4 networks. Thus, 6LoWPAN suggested the inclusion of an adaption layer in the IP stack between the network and data link layer. This new adaption layer permits the fragmentation and defragmentation of IPv6 packets in IEEE 802.15.4 frames, realizing the compression of the IPv6 head [10]. After the publication of six RFC (Request For Comments) documents (RFC 4919 [10], RFC 4944 [11], RFC 6282 [12], RFC 6568 [13], RFC 6606 [14], RFC 6775 [15]), the IETF defined the end of the 6LoWPAN working group as January 2014. The contributions of 6LoWPAN WG allowed the use of an IP-based network of tiny devices and, consequently, the existence of IoT applications [16]. Through the work of 6LoWPAN WG, a new working group was created to study routing solutions for the emerging Low-power and Lossy Networks (LLN) [17].

Created in February 2008, the Routing over Low-power and Lossy networks (RoLL) working group initially defined the routing requirements for urban LLNs [18], industrial LLNs [19], home automation LLNs [20], and building automation LLNs [21]. In parallel to the definition of the routing requirements, the RoLL working group initiated a study to verify whether the IETF standard routing protocols could supply the identified requirements [22]. Through the first conclusions, the RoLL detected that the existent routing solutions could not satisfy the requirements of LLNs. Since then, the working group has studied new routing solutions to provide the specific requirements of several types of LLN applications [17,23]. In March 2012, the RoLL working group defined the IPv6 Routing Protocol for LLNs (RPL) as the standard routing protocol for LLN. Presented in the RFC 6550 [24], RPL is a proactive tree-based routing protocol that builds a directed acyclic graph between the leaf nodes and the border route (sink node). Although considered as the standard routing protocol for IoT networks [25], since its creation, RPL has presented several drawbacks, and new solutions have been emerging to solve them.

The routing protocols for low power networks are summarized in several survey works. The works in [26–28] presented surveys about initial routing protocols for 6LoWPAN. A study about a secure routing protocol for IoT was presented in [29]. Considering the Smart Grid (SG) as a kind of IoT application inside the smart city domain [30,31], the authors in [32,33] showed survey studies about the routing protocols and network challenges for SG. Most recently, [34–36] have committed to surveying the existent routing solutions, considering improvements purely for RPL.

The weaknesses of the current solutions have motivated the emergence of an increasing number of new protocols and adaptations that seek to improve the network performance and avoid faults. Further, with the RPL-based improvements, it is also possible to find routing solutions for IoT networks based on other routing protocols, e.g., the AODV (Ad-hoc On-demand Distance Vector) routing protocol [37]. Thus, this work aims to present an extensive survey about the existent routing solutions

for low-power networks that can be used in IoT applications. Different from the current surveys existent in the literature, this work does not only consider RPL-based approaches, but all relevant solutions introduced during recent years. Thus, the main contributions of this work are summarized as follows:

- Presents the routing requirements identified by the IETF for LLN applications.
- Overviews the initial routing protocols considered to be adopted for LLNs.
- Identifies the issues most studied by the current proposals for routing in IoT scenarios.
- Presents a comprehensive survey showing the approach and functioning of the most relevant routing solutions for IoT/LLNs.
- Describes the studied solutions in summarizing tables and discusses the strengths and weakness of current routing solutions.
- Indicates relevant open issues and future directions to be appreciated by new routing approaches for IoT networks.

The rest of this work is organized as follows. Section 2 presents the routing requirements for applications over low-power networks. Section 3 elaborates the most-studied routing protocols for low-power and lossy networks, and Section 4 describes the most relevant routing solutions available in the literature to fulfill the requirements of IoT/LLN applications. A comprehensive discussion about the studied approaches, lessons learned, and the open issues and guidelines to be considered during the design of new solutions are addressed in Section 5. Section 6 concludes the paper.

2. Routing Requirements of Low-Power Networks and IoT Applications

The routing requirements of IoT/LLN networks may vary according to the specificities of applications. Thus, the RoLL working group initially presented four informational RFCs describing the necessities for different groups of LLN. In [18] were presented the routing requirements for Urban LLNs (U-LLN). This kind of LLN provides the network structure for applications executed in urban environments such as smart-metering, pollution and meteorological monitoring, and general purposes for smart cities. The document describes that U-LLNs can present three different types of devices: sensors, actuators, and routers. The actuator and sensor nodes have the capacity of performing actions and measure physical data in the environment in which they are deployed, respectively. On the other hand, routers have the function of prolonging the network's lifetime, balancing energy consumption, building the sensing infrastructure, and providing links to the Internet.

The number of sensor and actuator nodes is, generally, higher than the number of routers. It is expected that the number of sensors is between 10^2 and 10^7, deployed randomly or, e.g., along a road or river. The nodes have strong hardware constraints and a limited energy source. The communication distance between two nodes can vary from a few meters to one kilometer. It is common that the communication between a sensor/actuator and a border router needs to be performed through several hops, requiring multihop routing protocols. The communication reliability between the network nodes can be affected by different aspects, such as wireless channel effects, collision in the MAC layer, and interference [18].

The RoLL presented the routing requirements for Industrial LLNs in [19]. Security, monitoring, and control in industrial environments are some examples of applications supported by this group of LLNs. Thus, wireless devices should provide easy installation and maintenance, low power, and high reliability. Existing industrial LLN applications have, generally, networks with 10–200 nodes. The packet-sending rate demand ranges from 1/s to 1/h, with an average of 1/min. It is expected that critical alarm packets can be delivered with lower latency than normal messages. Hence, the routing protocol must support different routing metrics, including link quality and throughput. Further, industrial LLN applications can require the sending of messages for specific groups of nodes or more than one base station. In this case, multicast forwarding support becomes a critical routing requirement for these networks. The capacity of support in the addition of new nodes, real-time adaption to link

failures and route recomputing, and mobility support (with speeds up to 35 km/h) are other routing requirements for industrial LLNs.

Through RFC 5826 [20], the RoLL working group has defined the routing requirements for Home Automation LNNs (HA-LLN). Home automation scenarios encompass a great set of applications such as energy conservation and consumption optimization, window shade control, and healthcare [38]. Although the HA applications commonly use wired networks and power-line communication, the use of wireless networks can facilitate application expansion and possible upgrades. The routing requirements for HA-LLNs applications may vary according to the use case. For example, based on the above-mentioned informational RFC, a healthcare HA-LLN application requires a constraint-based routing, with mobility support and low convergence time. On the other hand, an alarm system HA-LLN application requires a network able to provide high scalability and convergence time. Further, the reliable support for different traffic patterns represents an indispensable feature for routing protocols in residential applications. Summarizing, a routing protocol able to support all HA-LLN applications must be ready to provide constraint-based routing and support to mobility, scalability, convergence time, manageability, and stability.

In [21], the routing requirements for Building Automation LLNs (BA-LLNs) were described. BA applications include the monitoring of several aspects such as fire alarms, elevator systems, control access, and intrusion detection systems [39]. In this context, wireless devices can increase the flexibility and reduce the deployment costs of applications. The network structure for BA applications should support at least two thousand nodes, including sensors, actuators, controllers, and user interface devices. A complete Building Management Systems (BMS) can be composed from different, smaller applications executed in several subnetworks that are connected through a backbone. Both the traffic pattern and the device density, as well as several other network peculiarities can diverge for each subnetwork. Therefore, the routing protocols must provide a self-organizing structure with zero-configuration to make the installation and replacement of devices accessible. Further, it must also provide scalability and support resource-constrained devices, mobility, addressing, manageability, dynamic route selection, and security.

During the definition of the requirements for the different LLN types, IETF studied its existent standard routing protocols. The objective was to identify whether they could attend to the routing requirements of low-power networks [22]. This study has considered the routing protocols for wired and wireless networks. Among these are OSPF (Open Shortest Path First) Version 2 [40], RIP (Routing Information Protocol) Version 2 [41], AODV [37], DYMO (Dynamic Mobile ad hoc network On-demand routing) [42], and DSR (Dynamic Source Routing) [43]. It was established that different basic criteria were derived from common requirements of LLNs applications. Thus, if at least one of the criteria was not met, the routing protocol was considered as not sufficient.

Based on the conclusions of the IETF study and with the necessity of solving the routing problems existent in LLNs, several extensions and new approaches have emerged. The next section presents the most relevant ones. This study has the objective of presenting a survey of routing solutions that may be employed in networks for IoT/LLN applications. Thus, only routing protocols for wireless networks with support for IPv6 are considered.

3. Routing Protocols for Low-Power and Lossy Networks

3.1. Initial Approaches

Following the presentation of the 6LoWPAN problem statement and goals through RFC 4919 [10], several routing protocols were proposed. The initial intention of the solutions was to offer routing solutions to provide multihop communication among IEEE 802.15.4 devices with support to IPv6. The main protocols, which are published, mainly, by means of Internet-Drafts (I-D), are briefly introduced in the following paragraphs.

The Hierarchical routing for 6LoWPAN (Hi-Low) is a hierarchical-based routing protocol that proposes the use of the dynamic address assignment scheme [44]. The basic idea of Hi-Low is to create a routing tree in which each node has a unique short address with 16 bits. In the Hi-Low topology, the network nodes are divided into three types: coordinator, router (parent), or end device (child). Each node of the network maintains a table with information about its neighbors such as its short address, device type, depth, and others. In the Hi-Low operation, a node initially seeks and tries to join an already existent network. If it finds a node that is already in the network, the node that is already part of the network should define a short address to the new node. If the network is not found, the node assumes the function of coordinator, creates a new network, and defines its short address as zero. A parent (or coordinator) node must define the address of its children nodes using Equation (1), in which *MC* is a parameter that defines the maximum number of children that a node can have and *addressParent* is the address of the parent. Hi-Low nodes frequently send beacon messages to update the neighboring information in the table. In the routing operation, a Hi-Low node uses the destination address to calculate whether the message should be sent to its parent or its child.

$$childAddress = (MC * addressParent) + 1 \quad (1)$$

The 6LoWPAN Ad hoc On-demand Distance vector routing (LOAD) is a simplified version of the well-known AODV developed for 6LoWPAN [45]. During the protocol functioning, it creates a mesh network topology without considering the IPv6 network layer. LOAD supports both the EUI-64 address and 16-bit short address. Seeking to reduce the size of control packets, LOAD does not use destination sequence numbers. Thus, to avoid loops, only the destination of an RREQ (Route Request) message can create an RREP (Route Reply) as a response. Although not mandatory, a LOAD node may use local repair when it detects a link break during data packet forwarding. When the local repair fails, a LOAD node may generate an RERR (Route Error) message to inform the originator of the data packet that it was not possible deliver the message to the destination. Different from AODV, an RERR message of LOAD indicates just one unreachable address. Due to limitations of network devices and seeking to reduce the computational resource usage, LOAD does not use a precursor list like in AODV. Furthermore, LOAD allows the use of the LQI (Link Quality Indicator) as a routing metric in addition to the hop distance. Although there are different ways of using LQI as a routing metric, LOAD adopts the LQI to avoid routes that have a link quality lower than an initially-defined threshold. Besides, LOAD does not use HELLO messages or passive acknowledgments.

In [46], the authors proposed a Dynamic MANET On-demand for 6LoWPAN (DYMO-low) routing protocol. DYMO-low is a routing protocol based on DYMO and, consequently, on the well-known AODV. DYMO-low uses three types of control messages: RREQ, RREP, and RRER (like in AODV). The protocol was fully projected for use over IEEE 802.15.4 devices. Thus, routing messages should not be fragmented. The choice of the best path to forward a message is done considering the LQI value and route cost. HELLO messages (used in AODV) are not used in DYMO-low. In contrast, the protocol uses IEEE 802.15.4 acknowledgment messages to determine if a neighboring node is reachable. Aiming to avoid loops, DYMO-low uses a simplified sequence number with 16-bits of length (DYMO sequence numbers are 32-bit). RREP must only be sent by the destination node of an RREQ. In DYMO-low, RERR messages indicate just one unreachable destination and are generated when: (i) an entry of the route table expires; (ii) a data packet cannot be successfully delivered to the next hop; or (iii) a broken link is detected.

A Hybrid Routing protocol for LLNs (HYDRO) was proposed in [47,48]. Hydrois a routing protocol for LLN, created to attend collection-based and point-to-point traffic. The hybrid approach combines local agility with centralized control. The network nodes create Directed Acyclic Graphs (DAGs) as default routes to the border router. On the other hand, the border router aggregates information about the nodes to create a global view of the network. This information is sent by the nodes through specific control messages or through opportunistic piggybacks in data packets. By creating a triangle between two nodes, the border router is responsible for forwarding the

point-to-point traffic in the network. When a node wants to send a data message to another node in the network (point-to-point traffic), the packet is sent to the border router that should forward the packet to the desired destination. When an active point-to-point flow is detected, Hydro uses a special control message to optimize the message exchange between these two nodes. Thus, the data packets can be forwarded through the shortest paths without the use of a border router. Each node maintains statistical information about its neighbors. This information is used to measure link quality and assist in route selection.

These routing protocols were defined before the specification of the routing requirements of LLN (presented in Section 2). As they are only simplifications of already existent protocols, they were unable to meet the requirements identified by the RoLL working group. As already mentioned, after identifying that the existent routing protocols defined by IETF were unable to supply the demands of LLN, the RoLL working group started the development of RPL. A description of RPL is provided in the next subsection.

3.2. RPL

RPL is a tree-based routing protocol created by the RoLL working group and defined by IETF as the standard routing protocol for LLNs [24]. RPL organizes the network topology in a Destination-Oriented Graph (DAG), which is composed of one or more Destination-Oriented Directed Acyclic Graphs (DODAG). Each DODAG represents a routing tree created by a root node, also known as a sink node or LBR (LLN Border Router). To construct the DODAG, RPL uses the Objective Functions (OF) that adopt routing metrics to calculate the best path between nodes and the DODAG root [49]. Thus, the routing structure created by the RPL is a logical topology built on a physical topology according to the used OF.

In the RPL operation, the network nodes initially build the DODAG. Thus, in this first step, the root node starts the process of DODAG construction, sending a DIO (DODAG Information Object) message to its neighbors. This message carries various relevant information, such as node rank, mode of operation, OF, and metrics. Each node that receives a DIO message should process it and decide whether or not to join the DODAG according to the used OF [50]. If a node chooses to join the DODAG, it has an upward path to the root node. At this moment, the node computes its rank, refreshes its neighbor table, and chooses its preferred parent, which will be used to forward messages to the DODAG root. Usually, the preferred parent is the neighbor with the lowest rank computed according to the OF. Some nodes may be configured to provide routes to other nodes. If a node is set to be a router, it should update and resend the DIO messages to its neighbors. If the node does not resend DIO, it is defined as a leaf in the routing tree. Each node that receives the DIO message should process it and continue the operation until all network nodes can be reached [51].

RPL permits a new node to join the network at any time. In this case, the new node uses a DIS (DODAG Information Solicitation) message to request a DIO message of a node already incorporated in the DODAG. Through the reception of the DIO message, the new node selects its preferred parent according to the OF. Furthermore, a DIO message is used in DODAG maintenance. Based on the Trickle Timer [52], DIO messages are periodically sent by the nodes seeking to maintain network stability.

RPL allows both upward (from the node to its parent) and downward (from the node to its children) routes. Upward routes are naturally created during the initial process of DIO sending. However, the use of downward routes requires the handling of DAO (Destination Advertisement Object) messages. The DODAG nodes process the DAO messages according to the RPL Mode Of Operations (MOP), which are presented below. Independent of the MOP used, DAO messages may require reception confirmation, which should be done using DAO-ACK messages.

Although it is designed for the Multipoint-to-Point (MP2P) traffic pattern, RPL also admits the data forwarding using Point-to-Multipoint (P2MP) and Point-to-Point (P2P) [53]. In MP2P, the nodes send data messages to the root, creating an upward flow (Figure 1a). In P2MP, sometimes termed as multicast, the root sends data messages to the other nodes, producing a downward flow (Figure 1b).

In P2P, a node sends messages to the other node (non-root) of DODAG; thus, both upward and downward forwarding may be required (Figure 1c).

RPL defines four MOPs that should be used considering the traffic pattern required by the application and the computational capacity of the nodes. In the first, MOP 0, RPL does not maintain downward routes; thus, consequently, only MP2P traffic is enabled. In non-storing MOP (MOP 1), downward routes are supported, and the use of P2P and MP2P is allowed. However, all downward routes are maintained in the root node. Thus, the total downward traffic should be initially sent to the DODAG root and subsequently be forwarded to its destination as shown in Figure 2a. In storing without multicast MOP (MOP 2), downward routes are also supported, but are different from MOP 1; the nodes maintain, individually, a routing table constructed using DAO messages to provide downward traffic. Hence, downward forwarding occurs without the use of the root node, as illustrated in Figure 2b. Storing with multicast MOP (MOP 3) has a functioning similar to MOP 2 plus the possibility of multicast data sending. This type of transmission permits the non-root node to send messages to a group of nodes formed using multicast DAOs.

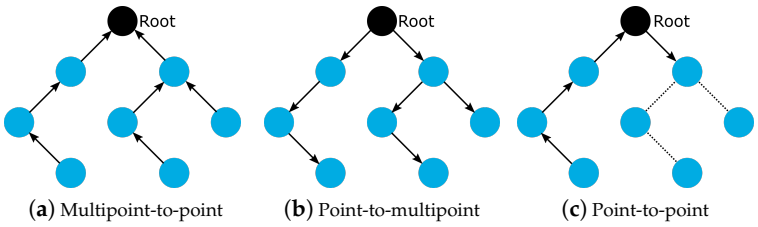


Figure 1. Traffic patterns supported by RPL. Lines with arrows indicate the traffic flow, while dotted lines without arrows indicate the links of the routing topology.

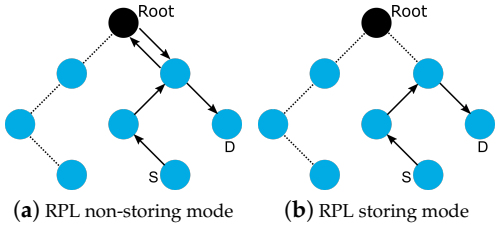


Figure 2. P2P message forwarding using Mode Of Operations 1 (MOP 1) and MOP 2. Lines with arrows indicates the traffic flow, while dotted lines without arrows indicate the links of routing topology.

In RPL, the whole process of DODAG construction is highly dependent on the OF considered by the network. Furthermore, the network performance is directly linked to the process of route selection. According to the metrics and constraints applied to select the best path between a node and a root, it is possible to increase or decrease the network performance. Furthermore, as already explained, some applications require different performance of the routing protocol. For example, e-health applications need low latency and high reliability [54]. On the other hand, some applications may require low energy consumption and mobility, such as animal monitoring applications. Thus, different types of applications can require different OFs. To fulfill the application requirements, each OF should contemplate metrics or constraints that better attend to these conditions. Based on the information of OFs, the node computes the weight of each path to the root. Hence, the selection of preferred parent and the rank calculation are performed according to OF. IETF defines, as the standard, two different objective functions: Objective Function zero (OF0) and Minimum Rank with Hysteresis Objective Function (MRHOF).

Objective Function zero (OF0) [55] is defined in the RFC 6552 and is designed to find the shortest path to the root. Using the DOI message information, OF0 knows the rank of each candidate

neighbor. OF0 selects, as the preferred parent, the candidate neighbor with the best rank (the shortest). Additionally, OF0 stores a backup feasible successor to use as an alternative to the preferred parent. All the upward traffic received by the node is routed to the root using the preferred parent or the backup successor, according to the link conditions. OF0 does not attempt to perform any load balancing. The backup successor is used to forward a packet to the root node when it is not possible to use the preferred parent. The rank of a node is obtained through the sum of the rank of its preferred parent and the value of *rank_Increase*, which is computed using a specific equation and predefined variables [55].

The Minimum Rank with Hysteresis Objective Function (MRHOF) [56] was developed to select the path with the lowest path cost without promoting excessive changes of the preferred parent. Thus, MRHOF uses two mechanisms to fulfill this purpose. The first seeks to find the path with minimum cost, while the second performs the hysteresis. Using the hysteresis mechanism, a candidate parent is selected as the preferred parent only when it has a path cost smaller than the current preferred parent, minus a given threshold [57]. Usually, MRHOF uses the ETXrouting metric to compute the cost of each path. However, it can utilize any additive routing metric described in [58]. During the network functioning, the path cost is re-computed periodically. Nevertheless, the preferred parent selection process is executed just when the path cost for a neighbor is refreshed or a new neighbor is inserted in the neighbors' table. As mentioned previously, the replacement of the preferred parent needs to attend to the hysteresis condition to avoid constant parental changes.

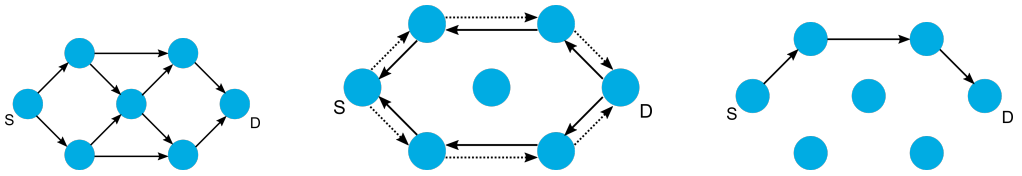
3.3. LOADng

The LOADng (Lightweight On-demand Ad hoc Distance-vector routing protocol-next generation) is a reactive routing protocol based on AODV and adapted for LLNs. LOADng was presented in an Internet-Draft to IETF in October 2011 [59]. Since then, LOADng has received various updates, and its last version was published in July 2016 [60]. Similar to AODV, LOADng only creates a route when a node needs to send a data message to another node. The process of route creation (also known as route discovery) is performed through the use of control messages adapted from AODV. The RREQ (Route Request) message is used during the route discovery process to find a path to the desired destination. The RREP (Route Reply) message is used by the destination of an RREQ to attend to the received route creation request. To allow the creation of bi-directional paths, an RREP can require a reception confirmation. In this case, an RREP_ACK (Route Reply Acknowledgment) message is used to answer the sender of the received RREP. Further, the LOADng also can use RERR (Route Error) messages to inform problems detected during the data message forwarding. In general, RERR is used when an intermediate node does not know the destination of a data message.

During the route discovery process of LOADng, the nodes should store a set of information about the other nodes of the network. Thus, each node has a routing set, a pending acknowledgment set, and a blacklisted neighbor set. The routing set, which works as a routing table, is formed by a set of entries created during the route discovery process. Each entry of the routing set contains information such as the destination address of the route, the address of the next hop to the destination, and the number of hops to reach the destination. The pending acknowledgment set is used to store data about the RREP messages sent with the requirement of an acknowledgment. As a complement to this last, the blacklisted neighbor set is employed to maintain the address of the nodes that have not replied with an RREP that required a reply (an RREP_ACK).

In the protocol functioning, when a node S needs to send a data message to a destination D, it first verifies its routing set, seeking a path to the desired destination. If the route is found, the node should use it to send the message. Otherwise, S must start the route discovery process. Thus, S creates an RREQ message with the same destination as the data message (node D). In sequence, S broadcasts the RREQ message to all its neighbors. Each node that receives the RREQ should verify several fields to identify whether it is valid for processing. This verification also avoids the creation of loops. The information carried in the message is used to update the routing set of nodes. In the sequence, the node refreshes the message fields and, if necessary, forwards it to the other nodes using broadcast

transmission. The RREQ message is forwarded by the intermediate nodes until reaching the destination *D* (Figure 3a). When *D* receives the RREQ, an RREP message is generated and sent to reply to the request originated by *S*. The RREP is forwarded in unicast using the path stored in the routing set during the RREQ broadcasts. If required by *D*, each intermediate node that receives the RREP should send an RREP_ACK to the previous hop to confirm the RREP reception (Figure 3b). When *S* receives the RREP, the route discovery process is finished and the data message can be sent through the created path (Figure 3c).



(**a**) RREQ transmission (**b**) RREP and RREP_ACK transmission (**c**) Data message transmission

Figure 3. LOADng control message transmissions. Arrows indicate the message flow. (**a**) Transmission of Route Request (RREQ) messages flooding all network nodes. (**b**) Transmission of Route Reply (RREP) and RREP_ACK messages, where bold lines indicate RREP messages that are forwarded through the path constructed by RREQ and dotted lines indicate RREP_ACK messages that are optionally sent after reception of each RREP, but not routed. (**c**) Transmission of data messages through the path selected by Snode after receiving RREP.

The RREQ and RREP messages carry fields to inform about the hop count and hop limit. Thus, so that these messages are always received, the node should increment the hop count and decrement the hop limit. The hop count field is used to inform about the number of hops traveled by the message since its originator. The hop limit field is implemented to indicate the number of remaining transmissions of the message. Thus, when a message is received with the hop limit equal to zero, it should not be forwarded to other nodes. Both messages also can transport another routing metric that should be computed and updated according to its specifications.

As mentioned above, LOADng is based on the well-known AODV. However, to permit the protocol to be executed on devices with severe hardware restrictions, some simplifications were made. In the LOADng, the routing table does not store the whole path to reach a destination. Besides, the control messages do not carry the address of each node traveled. Furthermore, only the destination of an RREQ message can reply to it using an RREP. Thus, the use of "intermediate RREP" mechanisms is not allowed. Finally, LOADng permits the use of a wide variety of addressing schemes (e.g., IPv6, IPv4, and Rime), and allows the use of different routing metrics as an alternative to the hop count.

4. Enhanced Routing Solutions for Internet of Things Networks

Currently, the routing solutions proposed for IoT/LLN are performed with the improvement of existent and well-known base protocols. Although several protocols have been used as inspiration, the principal and most recent approaches are based on the RPL and LOADng. Figure 4 presents a taxonomy of studied routing solutions, divided according to their proposals and motivation. Each considered solution is presented and described in the following subsections. Furthermore, at the end of each subsection, a summary table is shown to present the highlights of each approach, such as base protocol, objectives, descriptions, strengths, weaknesses, and application scenarios. The objective column presents the aims of the routing solution. The description column shows the method used by the proposal to reach its objectives. The strengths and weaknesses columns present, respectively, the main contributions and limitations of the solution. Finally, the application column shows the type of application to which the solution should, or intends to, be applied.

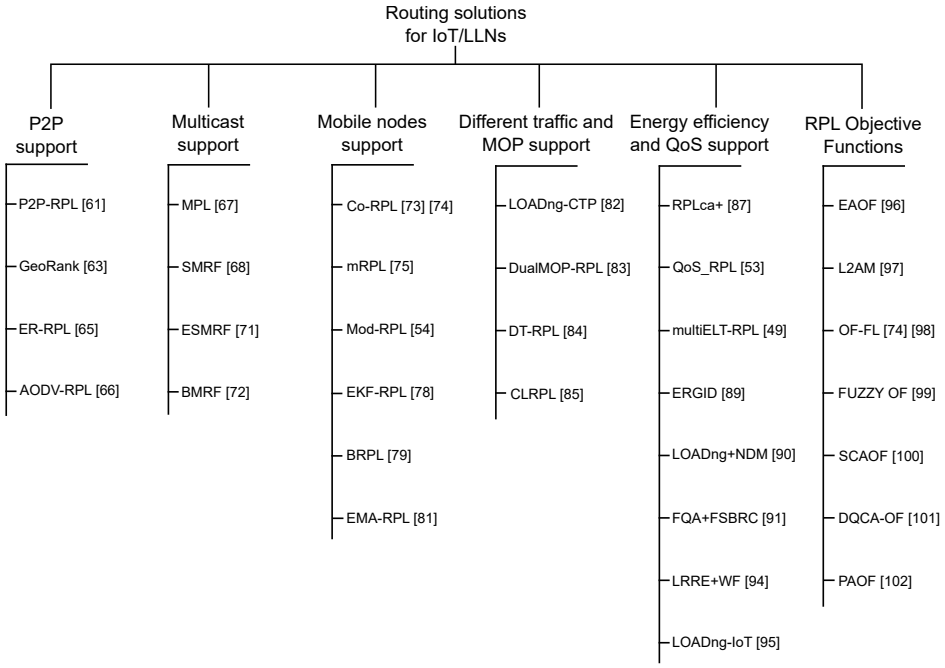


Figure 4. Taxonomy of routing solutions for IoT/Low-power and Lossy Networks (LLNs). ER, Energy-efficient Region-based; AODV, Ad-hoc On-demand Distance Vector; MPL, Multicast Protocol for Low-power and lossy networks; ESMRF, Enhanced Stateless Multicast RPL Forwarding for IPv6-based low-lower and lossy networks; BMRF, Bidirectional Multicast RPL Forwarding; BRPL, Backpressure RPL; DT, Diverse Traffic; CLRPL, Context-aware and Load balancing RPL; ELT, Expected Lifetime; ERGID, Emergency Response IoT based on Global Information Decision; NDM, Neighbor Disjoint Multipath; FQA, Fuzzy Q-Algorithm; FSBRC, Fuzzy System-Based Route Classifier; WF, Weighed Forwarding; EAOF, Energy-Aware Objective Function; OF-FL, Objective Function Fuzzy Logic; SCAOF, Scalable Context-Aware Objective Function; DQCA, Delivery Quality- and Context-Aware; PAOF, Parent-Aware Objective Function; Co, Corona.

4.1. P2P Communication Support

Peer-to-peer communication represents an import traffic pattern in LLNs, mainly in IoT scenarios. Although the default RPL implementation supports P2P traffic, the approach introduced by the IETF standard can be considered very costly in some situations. Thus, in April 2010, RoLL WG started the proposal of a new solution to improve the P2P supported by RPL. Hence, in August 2013, the Reactive Discovery of Point-to-Point Routes in Low-Power and Lossy Networks (P2P-RPL) was presented as an IETF standard in the RFC 6997 [61]. P2P-RPL is an extension of RPL that allows the creation of P2P routes on demand, working as a reactive routing protocol. Thus, a source node that wants to send a P2P data message should start a route discovery process, creating a temporary routing tree. The source, which acts as the root of the temporary DODAG, sends a P2P Route Discovery Option (P2P-RDO) across the entire network requesting a route to the destination node. Each node that receives the P2P-RDO must first check the message destination, determine if it joins in the DODAG, and select its temporary preferred parent. Furthermore, if the node is not the destination, it should forward the P2P-RDO. The P2P-RDO is forwarded across the network until it reaches its destination node or accomplishes its maximum number of hops. When the destination receives the P2P-RDO, it must generate a P2P Discovery Reply Object (P2P-DRO) to answer the route request. The P2P-DRO is forwarded back through the preferred parent chosen during the P2P-RDO forwarding. The intermediate nodes should store the reverse route used to deliver the P2P-DRO for a future data message forwarding. The source of P2P-RDO, after receiving the P2P-DRO, should start the sending of the P2P data message through

the route created. Complementarily, P2P-RPL permits the use of a mechanism to verify the necessity of starting a route discovery for P2P data sending. This mechanism, introduced in the RFC 6998 [62], uses Measurement Objects (MO) that assess the quality of the P2P route created by the default RPL. Whether or not it can fulfill the application routing requirements, the node should use it. Otherwise, the node should initiate the process of P2P route discovery (as previously described) seeking to attend to the application constraints. Figure 5 exemplifies the routes that can be created using both RPL and P2P-RPL for the transmission of a P2P data message.

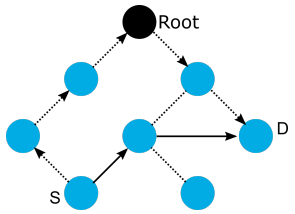


Figure 5. Transmission of a P2P message from node S to node D. The dotted lines with an arrow represent the path created by RPL, while the solid lines with an arrow represent the path established by P2P-RPL.

A geographic routing approach for 6LoWPAN was presented in [63]. The proposed GeoRank is based on RPL and GOAFR (Greedy Other Adaptive Face Routing) [64]. The GOAFR algorithm is able to seek optimal or sub-optimal routes in networks with high link density. However, considering networks with low link density, RPL can demonstrate a better performance than GOAFR. Thus, seeking to mix the better of two approaches, the authors proposed GeoRank. The approach aims to improve the P2P support of 6LoWPAN and reduce the amount of control messages required in this type of traffic. In its operation, the GeoRank algorithm initially calculates the distance between the source node and the destination based on the list of DODAG roots. After, the root with the lowest absolute angle difference between the source and the destination is selected as the anchor node. Subsequently, the algorithm tries to perform the routing of packet using greedy forwarding. In this mode, the node holding the message tries to send it to its neighbor with one hop to reach the message destination. If this is not possible, the algorithm assumes the GeoRank mode and then forwards the packet to the preferred parent in the path to arrive at the selected anchor. This forwarding process should occur until the message reaches a node closer to the destination than the anchor node. If this condition is attained, the greedy forwarding mode is assumed again. Otherwise, the message reaches the anchor and is forwarded in the face routing mode of GOAFR until it comes to its destination. In GeoRank, due to the use of geometric calculation, all nodes of the network must be static or be equipped with GPS. The use of mobile nodes is possible when they can communicate with a static node using just one hop. Furthermore, the message to be routed must store information about the position of its anchor. Although the authors used smart lighting system as an example of application for using GeoRank, they highlighted that the proposed solution might not meet the requirements of this kind of application.

Considering that the majority of P2P routing protocols, including P2P-RPL, flood the network with control packets to create routes, provoking great overhead and energy consumption, Zhao et al. [65] proposed the Energy-efficient Region-based Routing Protocol (ER-RPL). The approach aims to allow an energy-efficient P2P communication without harming the network's reliability. ER-RPL presents a hybrid approach, mixing proactive and reactive features based on RPL. The main idea of the protocol is to segment the network into different regions and realize the P2P route discovery, just considering the areas in which the nodes are located. To this end, ER-RPL requires the existence of location-aware nodes (e.g., equipped with GPS) in the network, named Reference Nodes (RN). The protocol uses the coordinates of RNs to partition the network into different regions and to calculate the distance between nodes and RNs. Afterwards, the nodes, based on the distance to RNs, should select their

regions. A binary number, named the Region Code (RC), identifies each region. This process occurs in an initial stage of the protocol when the RNs floods the network with Region Formation Object (RFO) control messages. For a P2P communication, the source node first should send a P2P route request to the destination node using the root node, i.e., the default route of RPL. This request is made using a control message named Message Request Object (MRO). After receiving the MRO, the destination node should verify whether the existent path is satisfactory based on the route cost. If valid, the destination node, through the reverse path and using an MRO, should inform the source node to start the sending of data. Otherwise, the destination should initiate the process of region-based route discovery. In this process, the destination node uses MRO to create a temporary DODAG just considering the regions between it and the source. Subsequently, this DODAG is used to send the data packets. Note that, at this stage, RN does not realize any task. Thus, ER-RPL functioning, although complex, can reduce the network energy consumption by avoiding the flooding of the whole network with the P2P route discovery packet. Further, the approach allows the sending of P2P messages without the use of a root node, creating near-optimal paths that can fulfill the reliability requirements of applications. For the other traffic patterns, such as MP2P, ER-RPL uses the structure built by default RPL functioning.

Although P2P-RPL can allow improved P2P data traffic for RPL networks, it still lacks in some areas. The protocol admits that the links among the nodes are symmetric, i.e., the quality of the linkage from Node A to Node B is the same (or almost the same) from B to A. Hence, the P2P messages exchanged between these nodes use a single path. However, in a practical environment, this symmetry may not be reached and, consequently, may not satisfy the routing requirements of a large variety of IoT applications. Thus, the Ad hoc On-demand Distance Vector routing-based RPL (AODV-RPL) [66] emerges as an improvement of RPL to support the P2P traffic pattern considering both the symmetric and asymmetric link during the path discovery process. As P2P-RPL, the peer-to-peer routes in AODV-RPL are constructed on-demand. A source node begins the route creation process to a destination when the current path does not fulfill the application requirements or the desired path does not exist. Thus, the source node must send DIO-RREQ (DIO with Route Request content) messages to create an RREQ-instance (DODAG created by the originator of an RREQ message) and find a path to the desired destination node. DIO-RREQ has a field to carry information about whether the route is symmetric or asymmetric (*S* field). Each node that receives the DIO-RREQ must verify the condition of the bidirectional link, update the *S* field, if necessary, and choose whether to join in the RREQ-instance. In this case, if a link is set as asymmetric, all routes that use this link are considered asymmetric. When DIO-RREQ reaches its destination, the node, based on the *S* field, must choose how to answer the route creation request. The destination node can also wait for a predefined period to receive DIO-RREQs through other routes. If the *S* field of the message indicates that the route created is symmetric, the destination node should unicast a DIO-RREP (DIO with Route Reply content) to the DIO-RREQ originator through the created path. Therefore, due to being symmetric, this single path can be used for data transportation in both directions. Otherwise, in case the route created is asymmetric, the destination node should start the creation of an RREP-instance (DODAG created by the destination of an RREQ message) multicasting DIO-RREP to its neighbors. DIO-RREPs are forwarded realizing the link symmetry verification, similarly to DIO-RREQs, until reaching the DIO-RREQ originator that, after receiving it, should initiate the sending of data messages through the path created by the DIO-RREP. Notice that when AODV-RPL does not find a symmetric path between two nodes, it can create a pair of DODAGs (RREQ-instance and RREP-instance) with a single-route discovery process. Thus, the P2P data message exchange can be performed using two different paths: a route from A to B and another from B to A. However, AODV-RPL is still an IETF Internet-Draft and can suffer modifications until its complete definition. Furthermore, to the best of the authors' knowledge, no other paper in the literature realizes a performance evaluation study about the efficiency of AODV-RPL.

Table 1 presents a summary of the above-described routing solutions.

Chapter 2. Routing Protocols for Low Power and Lossy Networks in Internet of Things Applications

Table 1. Comparison among studied approaches for P2P communication support. GOAFR, Greedy Other Adaptive Face Routing.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
P2P-RPL [61]	RPL	- Improve P2P support of RPL	- Creates new P2P routes on demand as an alternative to P2P routes built by RPL	- Offers the creation of alternative P2P routes to fulfill the application routing requirements - Avoids the use of a root node to forward P2P messages	- Floods the network with control packets that can increase the overhead and energy consumption	- Applications that require P2P traffic
GeoRank [63]	RPL and GOAFR	- Improve P2P support and reduce the amount of control messages	- Combines RPL with a geographic routing approach to reduce control message in P2P communication	- Avoids the use of DAO messages - Improves scalability, reducing the memory usage	- Changes the default RPL control messages - Requires static nodes or equipped with GPS	- Smart street lighting systems (as an example)
ER-RPL [65]	RPL	- Provide an energy-efficient and reliable P2P communication	- Performs a region-based P2P route discovery to save energy and create direct paths	- Avoids the P2P control message flooding by the whole network, hence producing a considerable energy savings - Improves the P2P packet delivery ratio	- Requires some location-aware nodes (e.g., with GPS) - Complex approach - Introduces new control messages in addition to default RPL messages	- Applications that require P2P traffic
AODV-RPL [66]	RPL and AODV	- Improves the support for the P2P traffic pattern of RPL	- Creates paired DODAGs for P2P message exchange through asymmetric routes	- Considers the bidirectional link condition during route discovery - Can reduce the size of control messages	- The current version is still in draft form - Has not presented performance results yet	- Applications that require P2P traffic

4.2. Multicast Communication Support

Multicast communication represents a fundamental transmission type for several IoT applications. As pointed out in Section 2, an industrial environment application running over an LLN can require the sending of alert messages for a predefined group of nodes or base stations. To achieve this, the routing protocol should realize reliable multicast transmissions.

Proposed in 2010, the Multicast Protocol for Low-power and lossy networks (MPL) [67], which was introduced in parallel to RPL and initially called trickle multicast, was the first effort of IETF to provide IPv6 multicast support for LLN. Defined as an IETF standard in 2016 through RFC 7731, the MPL uses a flooding mechanism controlled by trickle timers [52] to perform multicast message sending without the necessity to maintain routing tables. Although the MPL is an independent protocol, its solution is sometimes considered an extension to provide multicast support for RPL. In MPL, all network nodes can receive a multicast message depending on whether they are inserted in a multicast group. As all nodes can broadcast the received multicast packets, the implementation of a mechanism for avoiding duplicated messages is necessary. Thus, MPL includes a sequence number in the packet header for distinguishing each multicast message, besides storing the most recently-received messages in a buffer. After receiving a multicast message, the node should verify whether it was already received by checking its sequence number or presence in the buffer. If the node identifying the message is already received, it should discard it. Otherwise, the node must record the message sequence number, store it in the buffer, process the payload, and multicast it to the other nodes. MPL uses trickle to organize both control and data messages exchanged among the nodes. The nodes can use control messages for informing their neighbors about their set of sequence numbers or buffer. When a node identifies that a neighbor has not yet received a message, it redefines its trickle interval to the minimum value to quickly disseminate the packet missed by the neighbor. Thus, due to storing the most recent packets in a buffer and to allow local retransmission of lost packets, MPL can offer high reliability for the packet delivery ratio.

Although MPL can provide high reliability, it can also present a high end-to-end latency and communication overhead. Thus, trying to mitigate these drawbacks, Oikonomou et al. proposed the Stateless Multicast RPL Forwarding (SMRF) [68]. The proposal was constructed over RPL and uses the "storing mode of operation with multicast support" (multicast RPL or MOP 3) of the IETF standard solution to offer enhanced results when compared with the default approach. SMRF proposes a cross-layer mechanism that improves the functioning of Radio Duty Cycling (RDC) protocols for multicast forwarding operations. As an example, the authors cited ContikiMAC [69], a duty cycling mechanism present in the Contiki OS [70] and widely used in IoT/6LoWPAN networks. ContikiMAC presents low radio usage for reducing the power consumption of the nodes. Thus, the radio spends the majority of its time sleeping (turned off), waking-up (turning on), and periodically trying to identify a possible packet transmission. The duration of the period during which the radio is turned on is called CCI (Channel Check Interval). A node, to send a message, realizes several transmissions of the same packet during a time interval intending to reach an awake receiver. This repetitive transmission is referred to as a packet train. In a unicast transmission (Figure 6), a receiver node detects a packet sent during its CCI and starts the reception window until it receives the whole message. Afterwards, the receiver must send an acknowledgment to the sender, which finishes the packet train transmission. Differently, during a broadcast transmission (Figure 7), the sender must send a message during the entire packet train interval. Note that, due to a possible disturbance in the communication of other nodes, the transmission of acknowledgment packets is not used in ContikiMAC broadcast transmission. A receiver node cannot immediately forward the received packet due to possible collisions with the sender that may still be transmitting. For that reason, SMRF proposes the use of a short delay between the message reception and its forwarding for allowing the use of ContikiMAC broadcast and avoiding possible collisions (Figure 8). This behavior prevents the use of multiple ContikiMAC unicasts to the task of multicast transmission, commonly used by multicast RPL, and contributing to the reduction of energy consumption. Furthermore, multicast RPL does not specify any control mechanism to avoid

the reception and transmission of duplicated multicast packets. Thus, SMRF constrains that a node must only process multicast packets received from its preferred parents.

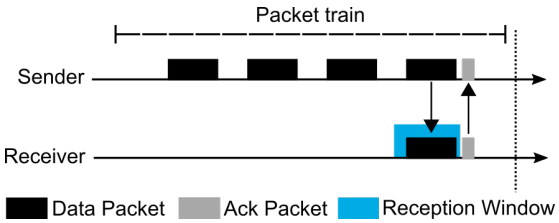


Figure 6. ContikiMAC unicast packet transmission.

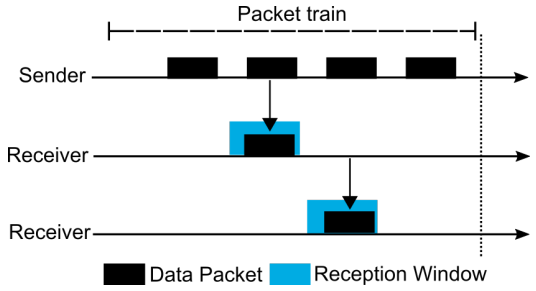


Figure 7. ContikiMAC broadcast packet transmission.

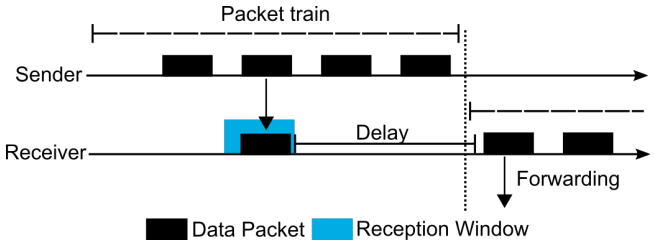


Figure 8. SMRF mechanism for multicast packet transmission.

SMRF, although showing several benefits when compared with standard RPL multicast, still presents some limitations. One of the main drawbacks of SMRF is the lack of support for upward multicast. Due to its mechanism for avoiding the processing of duplicated packets, as mentioned above, SMRF only allows nodes to process packets sent from its preferred parent. Thus, messages received from children nodes are neither processed nor routed. Therefore, to allow the nodes to transmit multicast messages both upward and downward, the Enhanced Stateless Multicast RPL Forwarding for IPv6-based low-lower and lossy networks (ESMRF) was proposed [71]. ESMRF proposes that multicast messages are first sent to the root node that, afterwards, should realize the message forwarding to its final destination. Thus, when a node wants to send a multicast message, it should encapsulate its content and destinations in an ICMPv6 (Internet Control Message Protocol Version 6) packet. This packet, also named a delegation packet, is sent in unicast to the root node (using the default RPL upward mechanism), which is in the top of the tree structure and can communicate with all network nodes. After receiving the delegation packet, the root node must extract the multicast content and send it to the destinations indicated. The process of forwarding multicast messages performed by the root node is similar to that proposed by SMRF. Thus, by using the same principle of SMRF, ESMRF also avoids the processing of duplicated packets.

ESMRF allows, using a simple approach, both upward and downward multicast communication, solving the gap of SMRF. However, the proposed solution suggests the use of the root node to forward all multicast messages. This approach becomes too expensive in a large routing tree, provoking communication overhead and high end-to-end latency. Thus, to mitigate the problem found in SMRF with a more robust method than proposed by ESMRF, Lorente et al. [72] introduced a new multicast routing protocol named Bidirectional Multicast RPL Forwarding (BMRF). BMRF seeks to merge the best features of RPL and SMRF. Thus, the approach offers three different modes for packet sending. In unicast mode, the sender node verifies the nodes interested in the multicast message according to the routing table, and then, the message is sent in unicast to each one, as proposed by RPL multicast. In broadcast mode, the sender verifies the group of nodes that want to receive the multicast message, and then, the communication is performed as in the broadcast proposed by SMRF. In the last mode, mixed-Tmode, the sender performs the transmission according to a T threshold that decides between multiple unicasts or broadcasts. T is defined according to the number of children interested in the message and the RDC rate.

During BMRF functioning, when a node wants to send a multicast message, it sends the packet both upwards and downwards, except for the root node, which should not be sent upwards, and leaf nodes, which should not be sent downwards. The upward sending is performed using unicast transmission. Otherwise, the downward sending is performed according to the BMRF mode. This behavior is necessary for avoiding duplicated packets. After receiving a multicast message, a node can process it depending on whether the packet was received from above or below. If the message was received from above, the node just should process it if the sender is its preferred parent (checked via MAC address). The node also consults its interest in the message to send it to the upper layer and verifies its routing table to forward the message to its children nodes. If the message was received in broadcast, the node should insert a delay in forwarding it (as in SMRF). This forwarding should be done according to the BMRF mode selected. If the multicast message was received from below and indicated with a unicast MAC address, the node should process the message once it was sent in unicast by a child node. Thus, the receiver node checks its interest in the message and then forwards it in unicast to the children inserted into the multicast group. The node also sends the message upwards, in unicast, if it is not the root. Hence, BMRF allows all network nodes to send and receive multicast messages, regardless of its position in the routing tree. Additionally, BMRF supports dynamic multicast group registration and avoids delivery disorder.

A summary of routing solutions for multicast support described in this subsection is presented in Table 2.

Table 2. Comparison among studied approaches for multicast communication support.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
MPL [67]	-	- Provide multicast data forwarding in LLNs	- Uses the flooding mechanism governed by the trickle algorithm	- Promotes a high packet delivery ratio due to buffer messages and retransmits it locally when necessary	- Can provoke a communication overhead - Low storing capacity of devices can limit the buffer size, causing a performance reduction	- Applications that require multicast transmission
SMRF [68]	RPL	- Enhance the multicast data forwarding provided by RPL	- Cross-layer approach that optimizes the ContikiMAC for multicast transmission	- Reduces the energy consumption occasioned by multiple unicast transmission in multicast RPL - Avoids processing of duplicated multicast packets	- Only allows downward multicast transmission - Can provoke high end-to-end latency	- Applications that require multicast transmission
ESMRF [71]	RPL and SMRF	- Permit both upward and downward multicast data forwarding in SMRF	- Encapsulates multicast messages and sends them to the root node to perform the forwarding	- Solves the gap of SMRF by allowing upward multicast traffic	- Oversees the communication by using the root node to forward all multicast traffic - Can cause high end-to-end latency	- Applications that require multicast transmission
BMRF [72]	RPL and SMRF	- Enhances both upward and downward multicast data forwarding in SMRF	- Merges the features of RPL and SMRF to offer different transmission modes used according to the needs	- Can reduce the number of radio transmissions and energy consumption - Increase the packet delivery ratio	- Slightly higher memory consumption - Increases the end-to-end latency - Wrong parameter adjustment can provoke low performance	- Applications that require multicast transmission

4.3. Mobile Nodes' Support

In addition multicast transmission, the support for mobile nodes also represents an important task for the majority of applications in IoT environments, mainly in urban and industrial scenarios. Thus, an efficient mobility support protocol should provide fast, continuous, and reliable communication among mobile and static nodes. Hence, in [73,74], the authors presented a variation of RPL based on the Corona mechanism (Co-RPL), which aims to ensure the quality of service in LLNs with mobile nodes. Co-RPL uses the default RPL control messages with additional fields to fulfill its purpose. All network nodes maintain a table with information about their neighbors (ID, DAG, corona, and link quality). When a node receives a DIO message, it selects its best parent, defines its corona ID, and computes its rank based on the objective function. The corona ID is determined through the result of the sum of the lowest corona ID among its neighbors plus one. After updating the DIO, the node resends the message in a broadcast. If a router node receives other DIO messages, it must select the preferred parent among the neighbors with the lowest corona ID. Whenever the corona ID of a neighbor changes or a new neighbor is detected, the node should execute the neighbor discovery process to update the corona IDs. Co-RPL proposes a new path recovery mechanism. When a node cannot reach the message destination, it sends the data packet to any node of the upper corona ID and sends a DIS message to its children to pause the sending of the data packet. This process occurs until the router node finds a new preferred parent. If a node cannot send the data packet to any neighbor of the high corona ID, it must inform the previous hop to pause the sending of data packets.

Fotouhi et at. [75] proposed the mRPL, a solution to enhance the mobility support in RPL. The proposed solution integrates the default RPL with smart-HOP [76,77], a hands-off mechanism initially developed for WSN. In mRPL, smart-HOP uses RPL-control messages like beacons. Considering the existence of two different types of nodes, which are Mobile Nodes (MN) and serving Access Points (AP), the smart-HOP mechanism is divided into two phases. In the data transmission phase, an MN sends a sequence of n DIS beacons to a serving AP. After receiving the n DIS, the serving AP calculates the Average Received Signal Strength Indication (ARSSI) value based on the n received messages and sends the obtained value to MN inside a DIO message. If the MN detects that the received ARSSI is lower than a defined threshold, the discovery phase begins. In this phase, an MN sends DIS control messages periodically until it receives a DIO message from an AP with an ARSSI greater than the defined threshold. When this condition is fulfilled, the MN returns to the data transmission phase with a new preferred parent. An MN can also start the discovery phase after detecting that its serving AP parent is not sending DIO messages. It is important to note that the messages exchanged in these two phases are disassociated from the data messages and control packets of RPL. Thus, mRPL nodes can coexist and interoperate in the same network when nodes execute default RPL.

Considering the necessity of mobility support for healthcare and medical applications, Gara et al. [54] proposed an adaption of RPL named mod-RPL. The approach contemplates applications executed over networks containing both mobile and static nodes. Thus, to attend to the routing requirements, the mod-RPL configures all mobile nodes as leaves of the DAG created by RPL. Therefore, mobile nodes are unable to send DIO messages and, consequently, be a parent of other nodes or perform routing tasks. The operation of the static nodes is equal to the default RPL. However, mobile nodes must not create or forward DIO messages, avoiding the creation of sub-DODAG and connection loss with possible child nodes. All the others' operations performed by mobile nodes (DIS and DAO sending, parent selection, and rank computation) are done like in the default RPL. In the end, mod-RPL regulates the transmission of control messages in the same way as the default RPL considering that the mobility in healthcare applications is not high.

Another routing solution for supporting mobile nodes in LLNs was proposed by Bouaziz et al. in [78]. Based on RPL, the Extended Kalman Filter for Mobile RPL (EKR-MRPL) aims to reduce the signaling overhead and energy consumption caused by constant changes of attachment points of mobile nodes. EKF-MRPL assumes that MNs do not execute any routing task. Thus, MNs just send and receive data from their preferred parent. EKF-MRPL operates following the basic concepts of RPL to construct the routing tree and execute the message forwarding. However, the new approach gives particular attention to the movement of MNs. This proposal is divided into three phases: movement detection, reaction, and notification. During the first phase, when MN is sending data messages to its Preferred Parent (PP), the PP should compute the RSSI for each packet received. Whether or not the RSSI value degrades below a predefined threshold, PP identifies that MN is moving away and should indicate it to find a new association node, i.e., a new PP. The current PP realizes this indication using a DIS message (flag = 1). After receiving the indication, MN starts the second phase. During this phase, MN should consider its direction and neighbor list to choose the new PP that can offer greater linkage time. For this purpose, MN broadcasts DIS (flag = 2) messages requesting the reception of DIO messages (flag = 1) from its static neighbors. After receiving DIO messages, MN should compute the RSSI of each neighbor. The computed RSSIs are used by the Extended Kalman Filter (EKF) to predict the MN movement trajectory. Posteriorly, MN should select its new PP considering its future movement and start the third phase. In this last phase, MN should send a DAO message to notify its neighbor that it was selected as the new PP. Furthermore, if possible, the MN should transmit a DAO "no-path" message to its previous PP to inform the disconnection. Notice that, between Phases 2 and 3, the MN can stop the sending of data messages to avoid packet loss and energy consumption. Hence, EKF-MRPL, through its movement prediction mechanism that allows MNs to select PPs with greater linkage time, reduces the number of preferred parent changes, contributing to the reduction of control message usage. Consequently, the protocol helps to reduce energy consumption and increase packet delivery.

Aiming to provide mobile support to LLN with highly dynamic network traffic, Tahir et al. proposed the Backpressure RPL (BRPL) [79]. BRPL, according to the authors, is the first routing protocol for LLN that merges RPL and the concepts of backpressure routing [80] to provide support to mobility and adaptability to various throughputs. As RPL, BRPL allows the use of multiple logical topologies (multiple DAGs) that can be created according to different OFs. However, contrary to RPL, each BRPL node maintains a queue of buffered packets for each DAG. Information such as RPL rank, queue length, and maximum queue length are exchanged among the nodes using DIO messages. Thus, each node that received a DIO message should update the information about the packet sender in the neighbor table. In the packet forwarding, the link weight of each neighbor is computed considering both its rank value and queue length. BRPL also uses a θ (theta) parameter to represent the tradeoff among the variables employed in the link cost computation (length of queues and rank of nodes). The θ value, which varies between zero and one, is dynamically adjusted by the QuickThetaalgorithm. QuickTheta defines the θ value according to the congestion level and the mobility of nodes. The congestion level is obtained by the ratio between the queue length of the node and its maximum queue. The mobility of the node, defined as β and computed by QuickBeta, is calculated in accordance with the changes in the neighbor table. Thus, the θ value is automatically adjusted for different traffic loads and topology changes provoked by mobility. θ also defines the switching of the routing strategy of BRPL. When its value is equal to one, BRPL realizes the packet forwarding using only the objective function of RPL. Otherwise, when θ is nearest to zero, BRPL tends to perform the forwarding task inspired by backpressure routing concepts. This dynamic strategy allows BRPL to achieve a significant packet loss reduction as detriment of an increased end-to-end delay.

An energy-efficient and mobility-aware routing protocol based on RPL was presented in [81]. The proposal, named EMA-RPL, considers low power networks composed by static nodes and MNs. To avoid the broken routes caused by the nodes' movement, the MNs are always defined as a leaf in the RPL tree. This feature prohibits MNs from performing the data message routing received from other nodes. MNs send and receive data messages through a static node, named the Associated Node (AN). Moreover, an AN is responsible for monitoring, predicting the movement, and defining a new AN to the MNs. The EMA-RPL functioning is divided into three different phases: mobility detection, reaction and prediction, and notification. The first phase is started when an MN moves away from its AN. The AN performs monitoring and detects the movement of MN nodes based on the RSSI value measured when data messages are received. If the measured RSSI is lower than a predefined threshold, the reaction and prediction phase is triggered. In this phase, the AN sends control messages searching for a new static node able to be used as the AN of the MN. When the new AN is found, the notification phase is started to inform the MN. Thus, the previous AN sends a control message to the MN informing about the change. After receiving this message, the MN should confirm the message reception and perform the AN change on its routing table. By avoiding MN sending several control messages monitoring its movements, EMA-RPL can reduce the energy and computational resource usage of mobile devices. However, in a dense and high mobility scenario, this approach can overload the static nodes and reduce the network performance. Furthermore, the proposal requires several changes in the structure of RPL control messages, which can make the interoperability of the EMA-RPL difficult with other approaches and the standard RPL implementation.

Table 3 presents a summary of the above-mentioned routing solutions for mobile nodes' support.

4.4. Different Traffic and Mode of Operations' Support

IoT/LLN applications can require the use of different data traffic patterns during its execution. As an example, two nodes can exchange messages among them creating P2P traffic, and a few moments later, both need to send data to a central node generating MP2P traffic. Furthermore, the central node can send data messages to a specific group of nodes using multicast communication (or P2MP). Thus, based on this requirement, the routing protocols should provide support for different traffic patterns, preferentially permitting that all these can be used simultaneously.

The main feature of LOADng is P2P traffic support. However, the most significant limitation of the protocol is the lack of efficient support for MP2P and P2MP. Seeking to reduce this drawback, Yi and Clausen [82] proposed the LOADng Collection Tree Extension (LOADng-CTP). The introduced improvement allows the LOADng to function as a proactive routing protocol and to perform the creation of a routing tree for the collection of data messages from the leaf nodes to the root (similar to what happens in the RPL). The LOADng-CTP introduces two new flags on the RREQ messages (RREQ_TRIGGER and RREQ_BUILD) and a new HELLO message. At the begin of protocol functioning, the root node floods the network using an RREQ_TRIGGER message (RREQ with the RREQ_TRIGGER flag set to true) to indicate the interest in constructing the routing tree. The nodes that receive the RREQ_TRIGGER answer it using the HELLO messages. Each receiver of a HELLO should set the message sender as bidirectional on its routing set. After a predefined time, the root node floods the network again using an RREQ_BUILD message (RREQ with the RREQ_BUILD flag set to true). The receivers of an RREQ_BUILD verify whether the message was received from a neighbor set as bidirectional, and if true, a new route entry to the root is inserted in the routing set. After this procedure, the collection tree is built, and the upward traffic (MP2P) can be performed more efficiently. Optionally, LOADng-CTP also permits the construction of a reverse path to facilitate the downward traffic. For this, each node that receives an RREQ_BUILD should answer it using an RREP message. Thus, LOADng-CTP overcomes one of the main limitations of LOADng, allowing it to support different traffic patterns appropriately.

Table 3. Comparison among the studied approaches for mobile nodes' support.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
Co-RPL [73,74]	RPL	- Improve the mobility support of RPL	- Routing solution based on the corona mechanism	- Presents an alternative mechanism to path recovery - Reduces the packet loss ratio, the average energy consumption, and the end-to-end delay	- Requires changes in the default RPL messages - Requires extension of the routing table	- General mobile wireless sensor network applications
mRPL [75]	RPL and smart-Hop	- Provide a fast and reliable mobility support in RPL	- Integrates the smart-Hop mechanism with RPL	- Presents a mechanism for collision and loop avoidance - Interoperable with default RPL - Reduces packet loss rate and delay - Source code available for Contiki OS	- Short increment in the control messages' length - Increases the number of exchanged control messages	- Wireless clinical monitoring applications
Mod-RPL [54]	RPL	- Adjust RPL for hybrid networks (mobile and static nodes)	- Modification of RPL to limit the operation of mobile nodes	- Reduces the use of control messages - Interoperable with default RPL	- Forbids the use of mobile nodes as routers - Only slow mobile nodes are considered	- Healthcare and medical applications
EKF-RPL [78]	RPL	- Improves mobility support of RPL and reduces the signaling overhead of mobile nodes	- Uses the extended Kalman filter to predict the movement of mobile nodes and to reduce the parent changes	- Reduces the control message overhead and energy consumption - Mobile nodes select the parent that can offer a higher linkage time	- Can increase the end-to-end latency - Does not consider important information as energy and link quality in the parent selection	- Application with mobile nodes
BRPL [79]	RPL and backpressure routing	- Enhances RPL to support mobility and dynamic traffic load	- Combines RPL with backpressure routing concepts to distribute the network resources adaptively	- Provides a considerable packet loss reduction - Can coexist in a network already running default RPL	- Increases the end-to-end delay significantly	- Application with mobile nodes and variable throughput
EMA-RPL [81]	RPL	- Increases the lifetime of mobile devices and improves network connectivity	- Monitors RSSI to predict the movement of the mobile nodes and promote the change of the preferred parent	- Use of power and computational resources of mobile nodes is reduced - Does not require the use of additional hardware for mobile detection (e.g., GPS)	- Mobile nodes cannot route packets from other nodes - Requires several additional fields on standard RPL control messages	- Healthcare applications

On the RPL, the supported traffic pattern is defined according to the utilized MOP, as described in Section 3.2. However, the choice of ideal MOP to be adopted also involves the consideration of the computational resources of the devices. Choosing the MOP with support to different traffic patterns (i.e., storing MOP) can require more hardware capacity and prevent "weaker" devices from running the protocol. In this case, a more reasonable solution could be to create a network with a different MOP being executed by the nodes with different capacities and making the "weaker" nodes work only as leaves. However, the presented scenario could face several interoperability problems once control messages are used and processed differently according to the MOP. Furthermore, as indicated on the RPL official document, each MOP should be adopted individually by the whole network. Thus, considering the limitations that may emerge for the using of different MOP simultaneously, Ko et al. have presented the DualMOP-RPL [83], an improved version of RPL that supports both storing and non-storing MOP at the same time without reducing network performance. The protocol, to reach this objective, introduces five different enhancements in the RPL. The first seeks to prevent the network partition by allowing the "weaker" nodes, initially defined as leaves for running an MOP different from the root, to work as routers. The second enhancement allows non-storing nodes to send hop-by-hop DAOs, contrary to what happens in RPL, where non-storing nodes only send DAOs to the root. Thus, storing mode nodes should process and store DAO messages received from storing and non-storing nodes. The third and fourth enhancements suggest modifications in DAO message fields and adaptations in their processing for both storing and non-storing nodes. Thus, the DAO message can be fully and correctly understood by both modes, avoiding problems of ignoring fields. The fifth enhancement, defined as optional, modifies a flag inside of DAO message to inform whether the DODAG root uses storing mode or not. The information of this flag should be recorded in the routing table of storing mode nodes to optimize the construction of downward routes. Thus, DualMOP-RPL, with this set of improvements, enables the use of both storing and non-storing mode nodes without decreasing the network performance and optimizes the use of computational resource devices.

Considering that the solution designed to allow the use of different MOP on RPL is more focused on providing interoperability, Kim et al. [84] have proposed an improvement for enhancing the support of RPL to diverse traffic. In their work, the authors realized several experiments with the RPL protocol, identifying its performance decline in scenarios with downward traffic. Thus, a modification of RPL was proposed, named DT-RPL (Diverse Traffic-RPL), for providing a stable bidirectional communication among the network nodes. The approach explores the messages exchanged among the nodes to deliver link quality information during downward packet sending. In default RPL functioning, the nodes only update their link quality with an upward packet. Using DT-RPL, the downward packets carry link quality information in a specific reserved space inside of the IEEE 802.15.4 MAC header. Thus, the receiver of a downward packet computes the link quality and updates this information about its parent. This mechanism makes the routing table updating faster and avoids the constant triggering of global repair during the transmission of downward packets. Hence, DT-RPL makes possible the use of different traffic patterns and also contributes to the packet delivery ratio increasing and overhead control.

To overcome the problem existent in RPL related to dynamic load networks, Taghizadeh et al. [85] proposed a new routing protocol called Context-aware and Load-balancing RPL (CLRPL). The protocol aims to reduce the packet loss ratio and increase the network lifetime in LLN with heavy throughput and highly-variable traffic. The CLRPL approach is divided into three parts. The first, named CAOF (Context-aware Objective Function), is a new composed objective function that computes the rank of each node based on ETX, residual energy in both the node and its parent, and the rank of the parent. This information, which piggybacks on the DIO messages, is stored by the node in an array after being received. After calculating the rank of each DIO sender based on the stored information, CAOF uses an algorithm to sort the nodes by rank from best to worst. Thus, DIO with the best rank is broadcast first to avoid the thundering herd phenomenon [86]. The second part is the Context-Aware Routing Metric (CARF). CARF uses information about the status of the queue chain in the path, the rank of the

node (computed using CAOF), and an index of network traffic dynamicity to obtain a value used in parent selection. As in CAOF, the information used by CARF is carried in the DIO. The third part is a parent selection mechanism that chooses the parent with the lower value computed by CARF. Whether or not the two candidate parents have the same CARF value, the mechanism selects the parent with the lower number of children nodes. The three parts are directly linked and exercise cooperative work. Thus, by considering the workload of the paths together with energy and link quality information, CLRPL can reduce energy consumption and improve the packet loss rate.

The routing solutions to support different traffic and MOP discussed above are summarized in Table 4.

4.5. Energy Efficiency and QoS Support

Reliability for data message sending and acceptable latency are requirements of all IoT applications. Due to power restrictions of various IoT devices, it is required that the packet sending task can be executed with the shortest energy expenditure possible. Therefore, a feasible QoS (Quality of Service) and reasonable energy consumption are considered as the most crucial features for all routing protocols for LLNs. Thus, in [87], Ancillotti et al. designed and developed a new cross-layer implementation of RPL, seeking to improve the data transmission reliability. The proposed solution, named RPLca+, is composed of two specialized libraries, the first for link quality estimation and the second for neighbor table management. The goal of the first library is to optimize the RPL operations through a hybrid link-monitoring framework to estimate the link quality with low overheads. Thus, the link quality library is composed of three methods of link quality estimation that are dynamically activated according to the information of the node and the characteristics of the link. The second library consists of a set of techniques for the management of neighbor tables. These techniques describe policies for insertion and replacement of entries in the RPL and IP tables seeking to manage better the neighbor's information considering the memory limitation of the network nodes. Furthermore, the proposed implementation provides a synchronization mechanism between the RPL and IP neighbor tables to improve the consistency of the stored information. This synchronization is done in a unidirectional way in which the inserts and replacements realized in the RPL table are reflected in the IP table, but are not contrary. Finally, it is important to note that the proposal is interoperable with the default RPL implementation since it does not change the RPL control message fields.

A routing metric based on the transmission delay and remaining energy was proposed in [53], titled QoS_RPL (Quality of Service RPL). The authors used an Ant Colony Algorithm (ACO) [88] seeking to better fulfill the requirements of energetic efficiency and QoS in LLNs. During routing protocol functioning, the information about energy and delay are piggybacked on the control messages. This information is computed and updated by each node that receives a packet. Furthermore, the approach uses the pheromone (of ACO) as a metric in the process of route selection. The pheromone information is updated whenever a route is used to forward a data packet, realizing the process of path reinforcement. Thus, the most used paths tend to be better evaluated by the proposed approach. However, the authors also implemented negative reinforcement to prevent the use of suboptimal routes and allow the adoption of other possible, better paths. The QoS_RPL has presented the capacity of reducing the delay and consumed energy. Nonetheless, the packet delivery ratio has shown a slight reduction compared with RPL using the ETX metric.

Table 4. Comparison among studied approaches for different traffic and MOP support. CTP, Collection Tree Extension.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
LOADng-CTP [82]	LOADng	- Permit LOADng to efficiently support MP2P traffic	- Introduces proactive features in LOADng for creating a bidirectional routing tree	- Significantly reduces the overhead and latency of LOADng in MP2P traffic - Easy implementation over LOADng core	- Introduces extra fields on default LOADng messages and a new HELLO message - Implementation requires new data structures that can increase the memory usage	- Not specified
DualMOP-RPL [83]	RPL	- Enable the use of both storing and non-storing MOP in a single RPL network	- Introduces a set of modification in RPL control messages and enhances its processing mode	- Solves interoperability problems existent between the two MOPs	- Increases the complexity of control messages' processing - Modifies the structure of the standard RPL control message	- Applications with heterogeneous devices
DT-RPL [84]	RPL	- Improves the reliability of RPL for different traffic patterns	- Allows a bidirectional measurement of link quality during the message exchange	- Improves the RPL performance during downward-centric communication - Does not require significant changes in the default RPL	- Does not consider energy information for load balancing - Can present limitations in scenarios with many nodes	- Not specified
CLRPL [85]	RPL	- Enhances support to heavy and highly dynamic load LLNs	- Creates new mechanisms to consider information about node workload, energy, and link quality in the parent selection process	- Uses relevant information in the parent selection - Reduces the changes of the preferred parent - Reduces energy consumption and increases PDR	- Requires an extra memory consumption and message sorting - Can increase the end-to-end delay	- Application with heavy and dynamic traffic load

A novel proposal for improving the network energy balancing seeking to maximize the lifetime of nodes was presented in [49]. The approach is based on RPL, and the authors focused on the creation of a routing solution that considers the estimation of energy consumption. Using a mechanism to measure the Expected Lifetime (ELT) of nodes and exploring multipaths, the proposed approach tries to avoid the use of bottlenecks (nodes with less energy) and equalizes the power consumption. Each node should compute its ELT based on the traffic expectation generated by itself and its children, the possible necessary retransmissions, the time during the transmissions, and the transmission power of its radio. The authors also suggested the use of a mechanism to limit the parental exchange. Thus, the quantity of a control message is reduced, contributing to fewer transmissions and energy depletion.

Qiu et al. proposed the ERGID (Emergency Response IoT based on Global Information Decision), a routing protocol that aims to provide both an efficient emergency response and reliable data transmission for IoT applications [89]. The proposal is based on two different mechanisms. The first, the Delay Iterative Method (DIM), classifies the nodes of a candidate route according to a global delay estimation. This mechanism is used to mitigate the problem of ignoring a valid path. Furthermore, DIM seeks to ensure real-time communication for emergency response applications and performs periodical updates in the routing table. The second mechanism, called Residual Energy Probability Choice (REPC), enables the use of residual energy information during the process of next node selection to forward a message. Thus, through the composition of these mechanisms, ERGID gives a low end-to-end delay (provided by DIM) and an efficient energy consumption distribution (supplied by REPC).

A Neighbor Disjoint Multipath scheme attached to the LOADng routing protocol (LOADng + NDM) was introduced in [90]. The proposed scheme seeks to reduce the problems caused by local node failures and a lousy channel condition in specific areas. During its functioning, LOADng + NDM first creates a primary path between the source and destination nodes, which is frequently the shortest route. In sequence, the scheme tries to build a set of backup paths. These alternative routes need to attend to some requirements, such as: none of its nodes can compose the primary path, except to source and destination nodes; and none of its nodes can be a neighbor of the nodes that form the primary path, except to source and destination nodes. Using this approach and considering applications with P2P traffic, LOADng + NDM can overcome the default LOADng regarding throughput and latency. It is also important to emphasize that the proposed neighbor disjoint multipath scheme can be used with different routing protocols, including the well-known RPL.

A framework for co-operating with routing protocols and providing QoS and energy efficiency for IoT networks was presented in [91]. Considering IoT scenarios that merge RFID and IEEE 802.15.4 devices, the proposed solution presents two mechanisms to improve the performance of both the routing protocol and the RFID tag-reading process. The first mechanism, named the Fuzzy Q-Algorithm (FQA), is a tag anti-collision protocol especially developed for IoT scenarios in which the tag reading process is realized for LLN devices equipped with RFID readers. FQA uses a fuzzy system to adjust the Q value [92] dynamically according to the tags' density and improve the reading process, making it faster and reliable. The second mechanism, called the Fuzzy System-Based Route Classifier (FSBRC), is a fuzzy system designed for joining four routing metrics (node energy, hop count, tags' density, and LQI) and defines a unique value that determines the route quality. The FSBRC uses the routing protocol control messages for transporting the metrics, which after being received, are used in the path quality computation. The nodes should select the path with higher route quality value to forward the data packets. Thus, the framework integrates both mechanisms to enhance the performance of IoT networks with heterogeneous devices and different wireless technologies. In the paper, the authors applied the proposed approach in the Directed Diffusion (DD) [93] routing protocol. However, they emphasized that the solution can be used with different protocols such as RPL and LOADng.

Sasidharan and Jacob [94] have proposed a new routing metric and a multipath forwarding mechanism for LOADng aiming to improve the network energy efficiency, load balancing, and data

message exchange reliability. The introduced composite routing metric, named LRRE, is based on three primary metrics: hop count, residual energy, and live routes. LRRE uses different parameters to allow the adjustment of the influence of each primary metric in the route value computation. Thus, α, β, and γ are used to tune the weight of residual energy, live routes, and hop count, respectively. Each weight is multiplied by its corresponding primary metric, and in the sequence, all the values are summed to obtain the LRRE value. The computed value is carried inside of default LOADng control messages (RREQ and RREP), not requiring additional fields. Hence, the cost of a path is the sum of the received LRRE value inside the message plus the value computed by the node. In addition to LRRE, the authors have also proposed adaption for LOADng that allows a node to create more than one route to the same destination. Thus, in the proposed functioning adaption, when an RREQ message is received, the node checks if the number of existing paths to the RREQ originator is lower than three. If true, a new route is added to the routing table. Otherwise, if the route indicated by the received RREQ is better than the one already existent, the new route should replace it. Finally, the authors' proposal also included a new Weighed Forwarding (WF) mechanism to allow the routing of data messages through the paths with the best LRRE values. The proposed set of improvements allows LOADng to construct multiple paths to a unique destination and provide load balancing and reliability in the message exchange process. However, this approach can increase memory usage and affect the network scalability, restricting the proposal adoption in large-scale IoT applications.

Due to implementation costs, it is expected that in several IoT scenarios, not all devices can have the capacity for direct communication with the Internet. In this case, the nodes without this capacity can use Internet-Connected (IC) devices as a gateway (or bridge) to communicate with external services and applications. Based on this specific heterogeneous IoT scenario, Sobral et al. [95] have proposed LOADng-IoT. The proposal introduces a new route discovery mechanism for the nodes to find IC nodes automatically without the need for a predefined gateway configuration. Thus, a node that wants to find a device to use as its gateway (i.e., an IC device) uses an RREQ-IoT message. The RREQ-IoT is broadcast and forwarded as a normal RREQ from default LOADng. However, the RREQ-IoT does not have a specific destination and can be replied by any device with an Internet connection in the network. When an IC node receives the RREQ-IoT, an RREP-IoT message is generated and sent to the request originator. The RREP-IoT is forwarded through the reverse path created by the RREQ-IoT until its destination. Each intermediate node that receives the RREP-IoT should add the message originator as a gateway to the Internet. Thus, in case there is a future need to send the message to the Internet, a path to an IC device is already known. LOADng-IoT also introduces a route cache mechanism to reduce the overhead of the route discovery process and a new error message used to inform when an IC node has lost its Internet connection. Thus, the proposal allows LOADng to better attend to the requirements of the QoS and energy efficiency needed by several IoT applications.

Table 5 shows a summary of studied solutions for energy efficiency and QoS support in IoT/6LoWPAN networks.

4.6. RPL Objective Functions

The RPL protocol uses objective functions to determine the network topology and the process of preferred parent selection. In its RFC document, RPL does not mandate the use of a specific OF and suggests that this choice should be made based on the application requirements. However, IETF has defined two initial OFs, OF0 and MRHOF, which can attend to simple routing requirements. Nonetheless, these OFs can present some limitations, e.g., they do not consider energy information during route selection. Thus, several works have proposed alternative OFs for RPL considering different routing metrics.

Table 5. Comparison among studied approaches for energy efficiency and QoS support.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
RPLca+ [87]	RPL	- Improves the reliability of data delivery in RPL	- Cross-layer approach for link quality estimation and routing table management	- Provides a dynamic link quality estimator - Provides the policies for the management of routing tables - Improves the packet delivery rate	- Presents implementation overhead - Increases the energy consumption	- Advanced metering infrastructure
QoS_RPL [53]	RPL	- Improves energy efficiency and QoS in LLNs	- Composite routing metric based on delay, remaining energy, and pheromone (ACO)	- Reduces the delay and energy consumption	- Decreases the packet delivery ratio - Can provoke a disturbance in the load balancing - Increases the control message overhead	- Not specified
multiELT-RPL [49]	RPL	- Maximizes the network lifetime and creates an energy balanced topology	- Computes the expected node lifetime and uses a multipath RPL modification	- Increases the node lifetime - Prevents the early necessity of routing topology reconfiguration	- Increases the memory usage - Modifies the structure of default RPL control messages	- Not specified
ERGID [89]	SPEED	- Provides low delay and load balancing for emergency response applications	- Uses the estimation of global delay of nodes and information about the energy consumption	- Chooses routes based on global information - Reduces the average delay and packet loss rate without increasing energy consumption	- Uses a high number of control messages to maintain delay information updated - Requires frequent calculations and routing table update	- Emergency response applications
LOADng+ NDM [90]	LOADng	- Reduces problems caused by local failures in the nodes or radio interferences	- Creates a set of alternative neighbor disjoint routes	- The main scheme can be adapted for different routing protocols - Avoids the usage of network regions with local problems	- Does not consider any information about energy consumption or link quality in the route selection - Increases the use of control messages and the complexity of the protocol	- Not specified
FQA + FSBRC [91]	DD	- Improves QoS and energy efficiency in IoT networks composed by RFID and LLN devices	- Designs a new tag-reading protocol integrated with a fuzzy system-based route classifier that uses four routing metrics to define the path quality	- Considers IoT devices equipped with RFID readers - Allows the consideration of RFID tags' density in the node area during the route selection process	- The approach is based on fuzzy systems that can be complex for IoT devices - The improved reading tags' algorithms requires changes in the standard protocol	- Applications that englobe RFID and LLN devices

Table 5. *Cont.*

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
LRRE + WF [94]	LOADng	- Improves the network lifetime, load balancing, and reliability	- Introduces a new composite routing metric and a new multipath route discovery and forwarding mechanism for LOADng	- Permits LOADng to construct multiple paths between source and destination nodes - Provides load balancing and improves network energy efficiency	- Proposed multipath adaption can increase memory usage and affect the network scalability - Bad adjusting of the parameters of the proposed routing metric can decrease the network performance	- Machine-to-machine communication applications
LOADng-IoT [95]	LOADng	- Allows LOADng to better discover and maintain routes in heterogeneous networks	- Introduces a new route discovery mechanism dedicated to creating paths to Internet-connected nodes	- Permits the nodes to find gateways to the Internet without a previous configuration - Presents a solution to reduce the overhead generated during the route discovery	- Requires the insertion of an extra field on the default LOADng control messages - Route cache mechanism can increase the memory usage	- Applications where nodes have different Internet connection capacities

The Energy-Aware Objective Function (EAOF) was introduced in [96]. EAOF seeks to increase the network lifetime and reliability of Biomedical Wireless Sensor Networks (BWSN). According to the authors, a BWSN should be able to work without human intervention for an extended period to further require reliable levels of QoS and efficient energy consumption. Thus, EAOF uses ETX and Remaining Energy (RE) to compute the best path to the root node aiming to satisfy the requirements of BWSN. EAOF performs the best parent selection in three steps and using two configurable parameters: MAX_ETX and MIN_ENER. In the first step, the proposed OF uses MAX_ETX as a threshold to define the maximum value that a potential parent should have to be considered in the next step. Thus, a node only can be considered as a candidate to be the best parent if it has an ETX value to the sink lower than MAX_ETX. In the next step, EAOF compares the rank of the candidate with the rank of the node. In the last step, the node with the highest RE is selected. EAOF introduces some hysteresis through the MIN_ENER parameter. Hence, a potential node only should be selected as the new preferred parent if the difference between its RE and the RE of the current parent is higher than MIN_ENER. This way, EAOF seeks to select the preferred parent with lower ETX and higher RE.

Some approaches do not propose a completely new OF, but improve how the rank of the nodes is computed, introducing a new routing metric. In [97], the authors proposed an additive composite metric called the Lifetime and Latency Aggregatable Metric (L^2AM). L^2AM aims to provide balanced energy consumption considering the reliability of data transmission along the path. To this end, the proposed routing metric merges a link reliability metric (i.e., ETX) with a new energy consumption metric termed Fully Simplified Exponential Lifetime Cost (FSELC). FSELC represents the power cost that each node needs to pay to send a message. During the RPL functioning, the nodes use DIO messages both for transporting information about the metrics and for informing the L^2AM value computed for each neighbor. The L^2AM value must be summed along the path to obtain the overall route cost. When a node needs to send a packet to the root, it should select the path with the lowest summation of L^2AM. Thus, the proposed metric allows the node to use routes that are reliable and energy efficient.

A fuzzy-based OF was presented in [74,98]. The proposed OF-FL (Objective Function Fuzzy Logic) aims to create a robust OF that can fulfill different routing requirements through consideration of several network metrics. In the work, the authors highlighted that only one, or a simple combination of, routing metrics cannot be sufficient to create a reliable path between a sender and a receiver. Thus, OF-FL uses a fuzzy system to combine four primary routing metrics (end-to-end delay, hop count, link quality, and node energy) and obtain a value that indicates the quality of a neighbor. As any OF, the information used to compute the rank of nodes piggybacks onto DIO messages. Hence, the nodes that receive a DIO execute the fuzzy system for computing the rank of the message sender neighbor. After, looking for the set of neighbors and its respective qualities, the OF-FL selects the best preferred parent based on the application requirements. Thus, OF-FL permits RPL, considering different network aspects during the route construction phase.

Another fuzzy-based OF was introduced in [99]. The proposal sought to merge different routing metrics to optimize QoS and energy consumption. The FUZZYOF combines three routing metrics, which are the delay, ETX, and energy, to obtain a fourth value termed quality that represents the path cost. Delay is the average time it takes for a packet to reach its destination. ETX, as already introduced, is the expected number of transmissions for a message to be delivered successfully. The energy metric expresses the lowest energy level among the nodes that compose a path. Different from the previously-presented OF-FL, FUZZY OF obtains the quality of a route using two fuzzification processes. The first considers the two metrics: delay and ETX. As a result of the first process, an output called QoS is produced. This QoS output, together with the energy metric, are the two inputs of the second fuzzification process, which provides the path quality value. Notice that the result of both fuzzification processes is a maximizable value. Thus, the node should select, as its preferred parent, the neighbor candidate with the highest quality value. In FUZZY OF, the rank value of a node is obtained through

the sum of the rank of its preferred parent with respective computed quality. Thus, the proposed OF allows RPL to construct a routing topology considering both QoS and energy aspects.

Araujo et al. [100] also adopted fuzzy systems to design a new set of context-aware objective functions for RPL. The authors aimed to provide a solution that can dynamically adjust its functioning to better attend to the requirements of the applications. Thus, they defined four Delivery Quality- and Context-Aware Objective Functions (DQCA-OF) considering the metrics of ETX, consumed energy, and the number of hops. A priority level was also defined for each metric used in DQCA-OFs. As an example, DQCA-OF1 was designed with the additive combination of the routing metrics ETX and number of hops. Each one of these metrics has a priority level that varies among high (1), medium (3), and low (5). Thus, the DQCA-OF proposal stored the routing needs of applications inside a database to adjust the priority level of metrics according to the requirements of each scenario. The authors also proposed a fuzzy system to combine the metrics of DQCA-OFs, alternatively. Thus, for each DQCA-OF exists a DQCA-OF (FL) that merges the routing metrics using the proposed fuzzy system to classify the routes. Therefore, the authors created a set of eight OFs that can be dynamically chosen according to the IoT application requirements for improving energy usage and QoS.

An objective function, developed for a specific implementation of RPL for Agricultural LLNs (A-LLNs), is presented in [101]. The Scalable Context-Aware Objective Function (SCAOF) introduces context-aware features in RPL to fulfill the QoS requirements of A-LLNs. The proposed OF combines Link Color Object (LCO) metrics and an additive metric composed of ETX and RE (remaining energy). The LCO is used to propagate information about the links, such as workload, hardware, and availability. The RE metric is an additive metric that represents the average remaining energy of the nodes that compose a path. All this information is obtained locally by the nodes or exchanged using DIO messages. To select its preferred parent, a sensor node seeks its parents with the appropriated LCO, and afterwards, it chooses the one with the lowest second metric (composed of ETX and RE). Next, the node should compute its rank based on ETX and RE and subsequently forward it to its neighbors. Hence, SCAOF provides the combined use of several routing metrics to fulfill the requirements of A-LLNs, mainly QoS and energy efficiency.

Gozuacik et al. [102] proposed the Parent-Aware Objective Function (PAOF), an objective function that aims to offer network load balancing for LLNs. PAOF uses both ETX and parent count to perform the rank computing and preferred parent selection. The parent count metric represents the number of potential preferred parents of the node. These two metrics are combined in a lexical way. In the comparison between two candidate nodes, PAOF first verifies the ETX modular difference between them. Whether it is smaller than *MinHopRankIncrease* or not, the node should select the best parent to be the candidate with the lowest parent count. *MinHopRankIncrease* is a default variable value of RPL that defines the minimum value increased in the rank for each parent of the node. Thus, although PAOF considers two routing metrics, the primary decision is based on the ETX, the second metric being used just in case of a significant difference among ETX values of candidates nodes.

The main features of the OFs studied during this subsection are shown in Table 6.

Table 6. Comparison among studied RPL objective functions. L^2AM, Lifetime and Latency Aggregatable Metric.

Proposal	Base Protocol	Objectives	Description	Strengths	Weaknesses	Target Applications
EAOF [96]	RPL	- Provides energy efficiency and balance	- Lexical composite OF based on ETX and remaining energy	- Increases the network lifetime - Easy implementation	- Packet delivery rate is decremented	- Biomedical WSN applications
L^2AM [97]	RPL	- Performs energy consumption balancing and improves reliability	- Additive composite OF based on energy and ETX	- Compatible with default RPL - Easy implementation - Improves network lifetime	- Packet delivery ratio is not studied	- Not specified
OF-FL [74,98]	RPL	- Provides a configurable routing decision with a fuzzy system	- OF based on fuzzy systems that mix end-to-end delay, ETX, node energy, and hop count	- Routing decisions are made considering different network aspects - Presents improvement in the end-to-end delay, network lifetime, and packet loss ratio	- Implementation of a fuzzy system can extend the memory usage - The definition of fuzzy parameters is not trivial and can strongly affect the network performance	- Not specified
FUZZY OF [99]	RPL	- Improves the QoS of RPL using fuzzy systems	- OF based on fuzzy systems that mix energy, delay, and ETX	- Reduces the packet loss ratio - Reduces the end-to-end delay	- Implementation of a fuzzy system can extend the memory usage - The definition of fuzzy parameters is not trivial and can strongly affect the network performance	- Not specified
DQCA-OF [100]	RPL	- Attends to the routing requirements of various IoT applications	- Set of OFs based on ETX, number of hops, and energy consumption combined using fuzzy systems	- Can change its features in running time to attend to the routing requirements - Improves network QoS	- Implementation of a fuzzy system can extend the memory usage - Requires previous knowledge of applications	- Not specified
SCAOF [101]	RPL	- Enhances the applicability of RPL in agricultural LLN, providing QoS	- Additive and lexical composite OF based on energy, link color, and ETX	- Performance evaluation realized in a real testbed - Allows the extension of network lifetime - Increases network reliability and efficiency	- Complex approach - Requires an extended version of RPL	- Agricultural LLNs
PAOF [102]	RPL	- Provides load balancing	- Lexical composite OF based on parent number and ETX	- Distribution of work load of nodes - Increases the network lifetime	- Does not consider node energy information	- Not specified

5. Discussion, Lessons Learned, and Open Issues

The last section presented the most relevant and recent approaches for improving the performance of routing layers in IoT/LLN networks. At the end of each subsection, comparison tables summarizing the most relevant characteristics (strengths and weaknesses) of the described protocols were included. Thus, it was possible to observe several relevant aspects of the current approaches.

The studied routing solutions for LLNs are, in general, based on the RPL routing protocol. This is justified due to RPL being an already consolidated and well-known solution. Furthermore, RPL, can initially better attend to the routing requirements presented by IETF when compared to other standard routing protocols. Furthermore, some approaches use solutions based on the combination of RPL with other already existent proposals. These approaches seek to solve the RPL drawbacks with the benefits of other previously-known proposals. Although these approaches can improve the standard RPL, the final algorithm tends to be long and complicated, requiring a higher storage and processing capacity of the nodes. According to the results presented in each work, the majority of the studied solutions can improve the packet delivery rate (or decrease the packet loss) and reduce energy consumption. Thus, it is clear that the RPL has various problems with packet delivery reliability and energy efficiency.

The broad adoption of RPL makes the most recent approaches attempt to overcome the existing limitations in the IETF routing standard. The costly support to P2P communication, provided by RPL, motivated the creation of new approaches such as P2P-RPL and AODV-RPL. These are, as well as LOADng, inspired by AODV. However, the inclusion of a reactive mechanism into RPL can increase the control message overhead and contribute to the increase of energy consumption. As alternatives, other solutions for P2P support improvement are based on geographic routing. These proposals tend to reduce the control message overhead and provide scalability, a critical requirement for several IoT applications. However, the necessity of location-aware nodes (e.g., equipped with GPS) can make its adoption very expensive.

Another feature defined continuously as a requirement of LLN applications is the support to multicast transmission. Industrial, home automation, and urban and building applications (as shown in Section 2) can present the necessity of realizing transmission for pre-defined groups of nodes. MPL, SMRF, ESMRF, and most recently, BMRF, are the main approaches to allow devices with limited computational resources to perform multicast transmission with high reliability and low energy consumption. However, these solutions tend to execute coordinated floods of messages or present cross-layer approaches. Such characteristics make their adoption costly regarding packet overhead and, in the case of cross-layer approaches, cause the dependency of using a specific MAC protocol [103] and the increasing end-to-end latency.

A considerable part of the current and future IoT/LLN applications admits the use of mobile devices. However, when neglected, the topology changes provoked by the movement of the nodes can cause strong adverse effects in network performance. Thus, mobile nodes support emerges as a requirement for all groups of IoT applications. As the current version of RPL does not present robust support for mobility, several works have been introduced to overcome this limitation. The approaches are very diverse and range from the use of merging RPL with another already existent solution to the adoption of movement prediction mechanisms. Although these approaches can improve mobility support when compared with RPL, they usually increase end-to-end latency and require extensions in the routing tables or increase the control messages' length.

The variety of IoT applications is unbounded, as well as its requirements of throughput and traffic flow. The main drawback of LOADng is the weak support of MP2P traffic. To surpass this limitation, LOADng-CTP introduces new features to allow protocol to work based on collection tree routing. In contrast, RPL presents four different kinds of MOP that can be adopted, one at a time, according to the traffic requirements of the application and hardware of the devices. However, some applications could take advantage of using more than one MOP to improve their operation or to allow new functionalities. Thus, a dual-MOP solution was proposed to attend to this necessity.

Furthermore, IoT applications can change their traffic pattern and load according to specific events or times. These requirements have motivated the creation of DT-RPL and CLRPL. These solutions can offer a robust self-adjustment capacity to attend to applications' requirements. In contrast, the protocol complexity, size of control messages, and latency can be widely increased. In the same way, a higher hardware capacity of devices is required.

Energy efficiency and QoS are requirements of the majority of IoT applications. Thus, LLN routing solutions should provide reliability for packet delivery with low latency and efficient energy consumption. The current proposals try to meet these necessities by using cross-layer approaches, multipath solutions, and computational intelligence algorithms, among others. The literature also presents solutions based on protocols different from RPL, such as ERGID, LOADng + NDM, LRRE + WF, and LOADng-IoT. Usually, OFs for RPL are proposed to attend to these same requirements. The studied OFs can improve the performance of RPL when compared with the standard OFs (e.g., OF0 and MRHOF). The OFs that consider a higher number of routing metrics tend to have better results. However, the implementation of a composed metric may not be trivial. Some solutions use fuzzy systems to perform the composition of routing metrics as an alternative to the conventional lexical and additive methods. Although fuzzy system-based OFs allow domain experts to be able to map their experience and decision-making processes to define the quality of a path, their implementation tends to be complex and requires more computational resources compared with traditional composition methods.

The presented discussion about the studied routing solutions has identified that the main constraints of the current routing solutions are the increment of algorithm complexity, the extension of message length, the increment of the use of control messages, and the implementation overheads. Furthermore, the majority of current solutions are evaluated employing computational simulations. Although simulations allow a higher variation of scenarios and number of nodes, their results may not represent the real performance of the approach. Beyond that, the current approaches designed for specific applications do not consider the routing requirements presented by IETF and RoLL for each application group. Although these considerations are not mandatory, it is important to note that they can represent the right direction towards a satisfactory functioning of a routing solution.

The development of routing solutions that can provide a stable service for the application layer of the IoT network is not an easy task. The different application requirements demand an acceptable trade-off among the various aspects of the network. Thus, some open issues and guidelines suggested for consideration in the creation of new approaches are as follows:

- Considering devices' heterogeneity: The current routing solutions are, usually, designed and evaluated considering just one or two kinds of devices. However, one of the first assumptions of IoT applications is the interoperability among heterogeneous devices. Different hardware configurations can affect the performance of routing protocols and the rate of message processing. Thus, it is critical that the development of a routing solution for IoT be conducted considering the diversity of hardware devices.
- Seek a satisfactory trade-off among the network aspects: As mentioned in the course of this work, the requirements of the 6LoWPAN routing protocol can vary according to the applications. Some applications can demand reliability, while others want energetic efficiency. However, the requirements of applications are frequently concurrent among them. As an example, the improvement of the network load balancing can affect the end-to-end delay negatively. Thus, routing solutions must seek a satisfactory trade-off among the concurrent requirement of the applications.
- Robust solutions with low complexity: The approaches currently presented seek to cover drawbacks, improve the weaknesses of an existent solution, or attend to as many requirements as possible. However, the design of an enhanced or "perfect" solution can generate unexpected results. With the increment of algorithm complexity, devices can be overloaded, causing a decrement to the network performance. With this, the approach needs to be robust to consider all

the relevant requirements in the scope of the applications and, at the same time, be simplistic and light for execution in devices with low computational power.

- Consider the IETF's recommendations: Although the consideration of the IETF specifications is not mandatory, several efforts were demanded to define the routing requirements for the different types of IoT/6LoWPAN applications. When considering the study approaches in the overview of this work, the routing solutions developed for specific applications do not consider IETF's recommendations. The consideration of IETF's guidelines can allow the proposal for a satisfactory behavior, high acceptability, and the future standardization of the solution.
- Interoperability with base protocols: The majority of studied routing solutions are based on already existent routing protocols. Thus, they use almost all the structure of the base protocols as RPL and AODV. However, a small number of current approaches provide support and interoperability with the protocol on which they are based. This gap makes the adoption of these new solutions difficult in an already functioning IoT network, because it would require updating the firmware of the nodes. Thus, it is recommended that new solutions can interact with the base protocols, allowing the coexistence of both in the same network. A way to attend to this requirement is by avoiding significant changes in the fields and processing the mode of control messages.
- Compatibility of existent OSs and boards: The current state-of-the-art presents a large set of boards (Arduino, WSN430 open node, M3 open node, TelosB, Zolertia Z1) that are constantly used in IoT scenarios. In recent years, several Operation Systems (OS) [104,105] were especially developed for the IoT context, e.g., Contiki OS [70] and RIOT OS [106]. It is desired that the new routing solution can be executed by the most recent adopted boards and OS for IoT. To this end, the proposal should be exhaustively tested through emulation, simulation, and real testbeds (although this can be expensive). The RENODE™ [107] can be helpful to reach these features.
- Make available and reproducible proposals: Although it is possible to find several routing proposals in the literature, the number of these that can be entirely reconstructed and studied is low. Commonly, the routing solution proposals are superficially described, and many details are suppressed. Thus, it is difficult to perform an accurate reproduction of the presented approaches. Providing flowcharts, algorithms, message structures, and mathematical equations helps the proposal to be better understood and makes it implementable. Because of this, the approach can be considered for comparison with another proposal or utilized for real IoT application. It is also essential to show a full description of the testbed or simulation parameters to allow a fair comparison among the solutions. It is desirable that future approaches can make public their implementations, simulation scripts, and trace logs, whenever possible.
- Self-adaption to the application context: Some applications can present different needs according to the time or type of data message. As an example, at a given time, an application can require low latency and may continue to need high reliability or lower energy consumption. Thus, routing solutions should self-adapt to the temporary context of applications. This feature can be obtained with the creation of cross-layer solutions in which the application layer indicates their particularities to the routing layer. However, it is important to highlight that some application can never require this characteristic. In this case, they can adopt a straightforward approach.

6. Conclusions

The creation of the 6LoWPAN working group by IETF allowed the development of a solution to provide the use of IPv6 over low power devices. This contribution boosted the emergence of IoT applications. However, since the definition of LLNs, the routing in the networks with low power devices is considered a significant challenge. Seeing the importance of studying appropriate routing solutions, IETF created the RoLL working group. Initial conclusions of a study developed by the new working group showed that the existent IETF standard routing protocols were unable to provide

the routing requirement of LLNs. Thus, RoLL began the development of a new routing solution considering the specificities identified by the group for different LLN applications.

Although the RoLL working group has defined RPL as the standard routing protocol for LLNs, several works have shown that the solution presents severe limitations and drawbacks. Thus, various new solutions studied in the course of this document have emerged in recent years trying to mitigate the existent routing issues. The currently-proposed enhancements are designed, in general, to solve problems related to P2P and mobility support, multicast data forwarding, energy efficiency, and QoS improvement. However, it is important to note that these new routing solutions are for the IoT/LLN network, as they are not limited to RPL enhancements. LOADng and its advances have emerged as a potential solution, especially for P2P communication, though it is much less studied than RPL in the recent literature.

The comprehensive analyses and discussions about the studied routing strategies have allowed the identification of the evolution of state-of-the-art solutions. However, it is also possible to point out various limitations of current proposals, which shows that they are still unable to fulfill the IoT routing requirements feasibly. Finally, several research directions and guidelines were presented for consideration in the design of new proposals.

Author Contributions: J.V.V.S. collected and performed a deep analysis and reviewed the related literature on the topic, wrote the first draft of the document, performed the comparison study, and identified some open research issues. J.J.P.C.R. supervised all of the study, consolidated the comparison analysis and open issues, and reviewed the structure and the first draft. R.A.L.R., J.A.-M., and V.K. reviewed the text carefully, verified the comparison study, and reviewed the identified open issues. J.V.V.S., J.J.P.C.R, R.A.L.R., J.A.-M., and V.K. contributed equally to the scope definition, motivation, and focus of the paper.

Acknowledgments: This work has been supported by the Brazilian National Council for Research and Development (CNPq) via Grant Nos. 201155/2015-0 and 309335/2017-5, by the National Funding from the FCT (Fundação para a Ciência e a Tecnologia) through the UID/EEA/500008/2019 Project, by the Government of the Russian Federation, Grant 08-08, by RNP with resources from MCTIC, Grant No. 01250.075413/2018-04, under the Radiocommunication Reference Center (Centro de Referência em Radiocomunicações (CRR)) project of the National Institute of Telecommunications (Instituto Nacional de Telecomunicações (Inatel)), Brazil, and by the International Scientific Partnership Program ISPP at King Saud University through ISPP #0129.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
2. Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [CrossRef]
3. Council, N. *Disruptive Civil Technologies: Six Technologies with Potential Impacts on Us Interests Out to 2025*; Conference Report CR; SRI Consulting Business Intelligence: Menlo Park, CA, USA, 2008.
4. Gluhak, A.; Krco, S.; Nati, M.; Pfisterer, D.; Mitton, N.; Razafindralambo, T. A survey on facilities for experimental internet of things research. *IEEE Commun. Mag.* **2011**, *49*, 58–67. [CrossRef]
5. Vermesan, O.; Friess, P. *Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems*; River Publishers: Aalborg, Denmark, 2013.
6. Díaz, M.; Martín, C.; Rubio, B. State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing. *J. Netw. Comput. Appl.* **2016**, *67*, 99–117. [CrossRef]
7. Domingo, M.C. An overview of the Internet of Things for people with disabilities. *J. Netw. Comput. Appl.* **2012**, *35*, 584–596. [CrossRef]
8. Perera, C.; Zaslavsky, A.; Christen, P.; Georgakopoulos, D. Context aware computing for the internet of things: A survey. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 414–454. [CrossRef]
9. Rodrigues, J.J.; Neves, P.A. A survey on IP-based wireless sensor network solutions. *Int. J. Commun. Syst.* **2010**, *23*, 963–981. [CrossRef]

10. Montenegro, G.; Schumacher, C.; Kushalnagar, N. *IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*; RFC 4919; IETF Secretariat: Fremont, CA, USA, 2007. [CrossRef]
11. Montenegro, G.; Hui, J.; Culler, D.; Kushalnagar, N. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*; RFC 4944; IETF Secretariat: Fremont, CA, USA, 2007. [CrossRef]
12. Thubert, P.; Hui, J. *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*; RFC 6282; IETF Secretariat: Fremont, CA, USA, 2011. [CrossRef]
13. Kim, E.; Kaspar, D.; Vasseur, J. *Design and Application Spaces for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)*; RFC 6568; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
14. Kim, E.; Kaspar, D.; Gomez, C.; Bormann, C. *Problem Statement and Requirements for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing*; RFC 6606; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
15. Bormann, C.; Shelby, Z.; Chakrabarti, S.; Nordmark, E. *Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)*; RFC 6775; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
16. Ishaq, I.; Carels, D.; Teklemariam, G.K.; Hoebeke, J.; Abeele, F.V.d.; Poorter, E.D.; Moerman, I.; Demeester, P. IETF standardization in the field of the internet of things (IoT): A survey. *J. Sens. Actuator Netw.* **2013**, *2*, 235–287. [CrossRef]
17. Ko, J.; Terzis, A.; Dawson-Haggerty, S.; Culler, D.E.; Hui, J.W.; Levis, P. Connecting low-power and lossy networks to the internet. *IEEE Commun. Mag.* **2011**, *49*, 96–101.
18. Watteyne, T.; Winter, T.; Barthel, D.; Dohler, M. *Routing Requirements for Urban Low-Power and Lossy Networks*; RFC 5548; IETF Secretariat: Fremont, CA, USA, 2009. [CrossRef]
19. Pister, K.; Phinney, T.; Thubert, P.; Dwars, S. *Industrial Routing Requirements in Low-Power and Lossy Networks*; RFC 5673; IETF Secretariat: Fremont, CA, USA, 2009. [CrossRef]
20. Porcu, G.; Buron, J.; Brandt, A. *Home Automation Routing Requirements in Low-Power and Lossy Networks*; RFC 5826; IETF Secretariat: Fremont, CA, USA, 2010. [CrossRef]
21. Martocci, J.; Mil, P.; Riou, N.; Vermeylen, W. *Building Automation Routing Requirements in Low-Power and Lossy Networks*; RFC 5867; IETF Secretariat: Fremont, CA, USA, 2010. [CrossRef]
22. Tavakoli, A.; Dawson-Haggerty, S. Overview of Existing Routing Protocols for Low Power and Lossy Networks. In *Internet-Draft draft-ietf-roll-protocols-survey-07*; Work in Progress; Internet Engineering Task Force: Fremont, CA, USA, 2009.
23. Sheng, Z.; Yang, S.; Yu, Y.; Vasilakos, A.; Mccann, J.; Leung, K. A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities. *IEEE Wirel. Commun.* **2013**, *20*, 91–98. [CrossRef]
24. Alexander, R.; Brandt, A.; Vasseur, J.; Hui, J.; Pister, K.; Thubert, P.; Levis, P.; Struik, R.; Kelsey, R.; Winter, T. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*; RFC 6550; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
25. Iova, O.; Picco, P.; Istomin, T.; Kiraly, C. RPL: The Routing Standard for the Internet of Things... Or Is It? *IEEE Commun. Mag.* **2016**, *54*, 16–22. [CrossRef]
26. Oliveira, L.M.; De Sousa, A.F.; Rodrigues, J.J. Routing and mobility approaches in IPv6 over LoWPAN mesh networks. *Int. J. Commun. Syst.* **2011**, *24*, 1445–1466. [CrossRef]
27. Kumar, V.; Tiwari, S. Routing in IPv6 over low-power wireless personal area networks (6LoWPAN): A survey. *J. Comput. Netw. Commun.* **2012**, *2012*, 316839. [CrossRef]
28. Babu, H.R.; Dey, U. Routing Protocols In IPv6 enabled LoWPAN: A Survey. *Int. J. Sci. Res. Publ.* **2014**, *4*, 1–6.
29. Airehrour, D.; Gutierrez, J.; Ray, S.K. Secure routing for internet of things: A survey. *J. Netw. Comput. Appl.* **2016**, *66*, 198–213. [CrossRef]
30. Borgia, E. The Internet of Things vision: Key features, applications and open issues. *Comput. Commun.* **2014**, *54*, 1–31. [CrossRef]
31. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of things for smart cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [CrossRef]
32. Sabbah, A.I.; El-Mougy, A.; Ibnkahla, M. A survey of networking challenges and routing protocols in smart grids. *IEEE Trans. Ind. Inform.* **2014**, *10*, 210–221. [CrossRef]
33. Saputro, N.; Akkaya, K.; Uludag, S. A survey of routing protocols for smart grid communications. *Comput. Netw.* **2012**, *56*, 2742–2771. [CrossRef]

34. Oliveira, A.; Vazão, T. Low-power and lossy networks under mobility: A survey. *Comput. Netw.* **2016**, *107*, 339–352. [CrossRef]
35. Kim, H.S.; Ko, J.; Culler, D.E.; Paek, J. Challenging the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL): A Survey. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2502–2525. [CrossRef]
36. Kamgueu, P.O.; Nataf, E.; Ndie, T.D. Survey on RPL enhancements: A focus on topology, security and mobility. *Comput. Commun.* **2018**, *120*, 10–21. [CrossRef]
37. Das, S.R.; Perkins, C.E.; Belding-Royer, E.M. *Ad Hoc On-Demand Distance Vector (AODV) Routing*; RFC 3561; IETF Secretariat: Fremont, CA, USA, 2003. [CrossRef]
38. Alaa, M.; Zaidan, A.; Zaidan, B.; Talal, M.; Kiah, M. A review of smart home applications based on Internet of Things. *J. Netw. Comput. Appl.* **2017**, *97*, 48–65. [CrossRef]
39. Zarpelão, B.B.; Miani, R.S.; Kawakani, C.T.; de Alvarenga, S.C. A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* **2017**, *84*, 25–37. [CrossRef]
40. Moy, J. *OSPF Version 2*. RFC 2328; IETF Secretariat: Fremont, CA, USA, 1998. [CrossRef]
41. Malkin, G.S. *RIP Version 2*. RFC 2453; IETF Secretariat: Fremont, CA, USA, 1998. [CrossRef]
42. Chakeres, I.; Perkins, C. Dynamic MANET on-demand (DYMO) routing. In *Internet-Draft draft-ietf-manet-dymo-15*; Work in Progress; Internet Engineering Task Force: Fremont, CA, USA, 2008.
43. Hu, Y.C.; Maltz, D.A.; Johnson, D.B. *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*; RFC 4728; IETF Secretariat: Fremont, CA, USA, 2007. [CrossRef]
44. Kim, K.; Yoo, S.; Park, J.; Park, S.D.; Lee, J. Hierarchical routing over 6LoWPAN (HiLow). In *Internet-Draft draft-daniel-6lowpan-hilow-hierarchical-routing-01.txt*; IETF Secretariat: Fremont, CA, USA, 2007.
45. Kim, K.; Park, S.D.; Montenegro, G.; Yoo, S.; Kushalnagar, N. 6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD). In *Internet-Draft draft-daniel-6lowpan-load-adhoc-routing-03.txt*; IETF Secretariat: Fremont, CA, USA, 2007.
46. Kim, K.; Montenegro, G.; Park, S.; Chakeres, I.; Perkins, C. Dynamic MANET On-demand for 6LoWPAN (DYMO-low) Routing. In *Internet-Draft draft-montenegro-6lowpan-dymo-low-routing-03.txt*; IETF Secretariat: Fremont, CA, USA, 2007.
47. Tavakoli, A.; Dawson-Haggerty, S.; Hui, J.; Culler, D. HYDRO: A hybrid routing protocol for lossy and low power networks. In *Internet-Draft draft-tavakoli-hydro-01.txt*; IETF Secretariat: Fremont, CA, USA, 2009.
48. Dawson-Haggerty, S.; Tavakoli, A.; Culler, D. Hydro: A hybrid routing protocol for low-power and lossy networks. In Proceedings of the 2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm), Gaithersburg, MD, USA, 4–6 October 2010; pp. 268–273.
49. Iova, O.; Theoleyre, F.; Noel, T. Using multiparent routing in RPL to increase the stability and the lifetime of the network. *Ad Hoc Netw.* **2015**, *29*, 45–62. [CrossRef]
50. Clausen, T.; Herberg, U.; Philipp, M. A critical evaluation of the ipv6 routing protocol for low power and lossy networks (RPL). In Proceedings of the 2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Wuhan, China, 10–12 October 2011; pp. 365–372.
51. Vasseur, J.; Agarwal, N.; Hui, J.; Shelby, Z.; Bertrand, P.; Chauvenet, C. *RPL: The IP Routing Protocol Designed for Low Power and Lossy Networks*; Internet Protocol for Smart Objects (IPSO) Alliance: Santa Clara, CA, USA, 2011; Volume 36.
52. Levis, P.; Clausen, T.H.; Gnawali, O.; Hui, J.; Ko, J. *The Trickle Algorithm*; RFC 6206; IETF Secretariat: Fremont, CA, USA, 2011. [CrossRef]
53. Mohamed, B.; Mohamed, F. QoS Routing RPL for Low Power and Lossy Networks. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 971545. [CrossRef]
54. Gara, F.; Ben Saad, L.; Ben Ayed, R.; Tourancheau, B. RPL protocol adapted for healthcare and medical applications. In Proceedings of the 2015 International Wireless Communications and Mobile Computing Conference (IWCMC), Dubrovnik, Croatia, 24–28 August 2015; pp. 690–695.
55. Thubert, P. *Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)*; RFC 6552; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
56. Gnawali, O.; Levis, P. *The Minimum Rank with Hysteresis Objective Function*; RFC 6719; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
57. Palattella, M.R.; Accettura, N.; Vilajosana, X.; Watteyne, T.; Grieco, L.A.; Boggia, G.; Dohler, M. Standardized protocol stack for the internet of (important) things. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1389–1406. [CrossRef]

58. Barthel, D.; Vasseur, J.; Pister, K.; Kim, M.; Dejean, N. *Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks*; RFC 6551; IETF Secretariat: Fremont, CA, USA, 2012. [CrossRef]
59. Clausen, T.; Yi, J.; Lavenu, C.; Lys, A.; Niktash, A.; Igarashi, Y.; Satoh, H. The LLN On-demand Ad hoc Distance-vector Routing Protocol-Next Generation (LOADng). In *Internet-Draft Draft-Clausen-lln-Loadng-00.txt*; IETF Secretariat: Fremont, CA, USA, 2011.
60. Clausen, T.; Yi, J.; Niktash, A.; Igarashi, Y.; Satoh, H.; Herberg, U.; Lavenu, C.; Lys, T.; Dean, J. The lightweight on-demand ad hoc distance-vector routing protocol-next generation (LOADng). In *Internet-Draft draft-clausen-lln-loadng-15.txt*; IETF Secretariat: Fremont, CA, USA, 2016.
61. Goyal, M.; Baccelli, E.; Philipp, M.; Brandt, A.; Martocci, J. *Reactive Discovery of Point-to-Point Routes in Low-Power and Lossy Networks*; RFC 6997; IETF Secretariat: Fremont, CA, USA, 2013. [CrossRef]
62. Goyal, M.; Baccelli, E.; Brandt, A.; Martocci, J. *A Mechanism to Measure the Routing Metrics along a Point-to-Point Route in a Low-Power and Lossy Network*; RFC 6998; IETF Secretariat: Fremont, CA, USA, 2013. [CrossRef]
63. Barriquello, C.H.; Denardin, G.W.; Campos, A. A geographic routing approach for IPv6 in large-scale low-power and lossy networks. *Comput. Electr. Eng.* **2015**, *45*, 182–191. [CrossRef]
64. Kuhn, F.; Wattenhofer, R.; Zollinger, A. Worst-case optimal and average-case efficient geometric ad-hoc routing. In Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing, Annapolis, MD, USA, 1–3 June 2003; pp. 267–278.
65. Zhao, M.; Ho, I.W.H.; Chong, P.H.J. An Energy-Efficient Region-Based RPL Routing Protocol for Low-Power and Lossy Networks. *IEEE Internet Things J.* **2016**, *3*, 1319–1333. [CrossRef]
66. Anamalamudi, S.; Zhang, M.; Sangi, A.R.; Perkins, C.E.; Anand, S.; (Remy), B.L. Asymmetric AODV-P2P-RPL in Low-Power and Lossy Networks (LLNs). In *Internet-Draft draft-ietf-roll-aodv-rpl-03*; Work in Progress; Internet Engineering Task Force: Fremont, CA, USA, 2018.
67. Hui, J.; Kelsey, R. *Multicast Protocol for Low-Power and Lossy Networks (MPL)*; RFC 7731; IETF Secretariat: Fremont, CA, USA, 2016. [CrossRef]
68. Oikonomou, G.; Phillips, I.; Tryfonas, T. IPv6 Multicast Forwarding in RPL-Based Wireless Sensor Networks. *Wirel. Pers. Commun.* **2013**, *73*, 1089–1116. [CrossRef]
69. Dunkels, A. *The Contikimac Radio Duty Cycling Protocol*; Tech. Rep. T2011:13; Swedish Institute of Computer Science: Kista, Sweden, 2011.
70. Dunkels, A.; Gronvall, B.; Voigt, T. Contiki-a lightweight and flexible operating system for tiny networked sensors. In Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, Tampa, FL, USA, 16–18 November 2004; pp. 455–462.
71. Abdel Fadeel, K.Q.; El Sayed, K. ESMRF: Enhanced stateless multicast RPL forwarding for IPv6-based low-Power and lossy networks. In Proceedings of the 2015 Workshop on IoT challenges in Mobile and Industrial Systems, Florence, Italy, 18 May 2015; pp. 19–24.
72. Lorente, G.G.; Lemmens, B.; Carlier, M.; Braeken, A.; Steenhaut, K. BMRF: Bidirectional Multicast RPL Forwarding. *Ad Hoc Netw.* **2017**, *54*, 69 – 84. [CrossRef]
73. Gaddour, O.; Koubâa, A.; Rangarajan, R.; Cheikhrouhou, O.; Tovar, E.; Abid, M. Co-RPL: RPL routing for mobile low power wireless sensor networks using Corona mechanism. In Proceedings of the 2014 9th IEEE International Symposium on Industrial Embedded Systems (SIES), Pisa, Italy, 18–20 June 2014; pp. 200–209.
74. Gaddour, O.; Koubâa, A.; Abid, M. Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL. *Ad Hoc Netw.* **2015**, *33*, 233–256. [CrossRef]
75. Fotouhi, H.; Moreira, D.; Alves, M. mRPL: Boosting mobility in the Internet of Things. *Ad Hoc Netw.* **2015**, *26*, 17–35. [CrossRef]
76. Fotouhi, H.; Zúñiga, M.; Alves, M.; Koubâa, A.; Marrón, P. Smart-hop: A reliable handoff mechanism for mobile wireless sensor networks. In *Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 131–146.
77. Fotouhi, H.; Alves, M.; Zuniga Zamalloa, M.; Koubâa, A. Reliable and fast hand-offs in low-power wireless networks. *IEEE Trans. Mob. Comput.* **2014**, *13*, 2620–2633. [CrossRef]
78. Bouaziz, M.; Rachedi, A.; Belghith, A. EKF-MRPL: Advanced mobility support routing protocol for internet of mobile things: Movement prediction approach. *Future Gener. Comput. Syst.* **2019**, *93*, 822–832. [CrossRef]
79. Tahir, Y.; Yang, S.; McCann, J. BRPL: Backpressure RPL for High-Throughput and Mobile IoTs. *IEEE Trans. Mob. Comput.* **2018**, *17*, 29–43. [CrossRef]

80. Tassiulas, L.; Ephremides, A. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Trans. Autom. Control* **1992**, *37*, 1936–1948. [CrossRef]
81. Bouaziz, M.; Rachedi, A.; Belghith, A.; Berbineau, M.; Al-Ahmadi, S. EMA-RPL: Energy and mobility aware routing for the Internet of Mobile Things. *Future Gener. Comput. Syst.* **2019**, *97*, 247–258. [CrossRef]
82. Yi, J.; Clausen, T. Collection Tree Extension of Reactive Routing Protocol for Low-Power and Lossy Networks. *Int. J. Distrib. Sens. Netw.* **2014**, *10*, 352421. [CrossRef]
83. Ko, J.; Jeong, J.; Park, J.; Jun, J.A.; Gnawali, O.; Paek, J. DualMOP-RPL: Supporting Multiple Modes of Downward Routing in a Single RPL Network. *ACM Trans. Sens. Netw. (TOSN)* **2015**, *11*, 39. [CrossRef]
84. Kim, H.S.; Cho, H.; Kim, H.; Bahk, S. DT-RPL: Diverse bidirectional traffic delivery through RPL routing protocol in low power and lossy networks. *Comput. Netw.* **2017**, *126*, 150–161. [CrossRef]
85. Taghizadeh, S.; Bobarshad, H.; Elbiaze, H. CLRPL: Context-Aware and Load Balancing RPL for Iot Networks Under Heavy and Highly Dynamic Load. *IEEE Access* **2018**, *6*, 23277–23291. [CrossRef]
86. Hou, J.; Jadhav, R.A.; Luo, Z. Optimization of Parent-node Selection in RPL-based Networks. In *Internet-Draft draft-hou-roll-rpl-parent-selection-00*; Work in Progress; Internet Engineering Task Force: Fremont, CA, USA, 2017.
87. Ancillotti, E.; Bruno, R.; Conti, M. Reliable data delivery with the IETF routing protocol for low-power and lossy networks. *IEEE Trans. Ind. Inform.* **2014**, *10*, 1864–1877. [CrossRef]
88. Dorigo, M.; Birattari, M., Ant Colony Optimization. In *Encyclopedia of Machine Learning*; Springer US: Boston, MA, USA, 2010; pp. 36–39.
89. Qiu, T.; Lv, Y.; Xia, F.; Chen, N.; Wan, J.; Tolba, A. ERGID: An efficient routing protocol for emergency response Internet of Things. *J. Netw. Comput. Appl.* **2016**, *72*, 104–112. [CrossRef]
90. Hossain, A.K.M.M.; Sreenan, C.J.; Alberola, R.D.P. Neighbour-Disjoint Multipath for Low-Power and Lossy Networks. *ACM Trans. Sens. Netw.* **2016**, *12*, 23:1–23:25. [CrossRef]
91. Sobral, J.V.; Rodrigues, J.J.; Rabelo, R.A.; Filho, J.C.L.; Sousa, N.; Araujo, H.S.; Filho, R.H. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs. *J. Netw. Comput. Appl.* **2018**, *107*, 56–68. [CrossRef]
92. EPCglobal. *EPCTM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz–960 MHz Version 1.2.0*; EPCglobal: Brussels, Belgium, 2008.
93. Intanagonwiwat, C.; Govindan, R.; Estrin, D. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, MA USA, 6–11 August 2000; pp. 56–67.
94. Sasidharan, D.; Jacob, L. Improving network lifetime and reliability for machine type communications based on LOADng routing protocol. *Ad Hoc Netw.* **2018**, *73*, 27–39. [CrossRef]
95. Sobral, J.V.V.; Rodrigues, J.J.P.C.; Rabelo, R.A.L.; Saleem, K.; Furtado, V. LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks. *Sensors* **2019**, *19*, 150. [CrossRef]
96. Abreu, C.; Ricardo, M.; Mendes, P. Energy-aware routing for biomedical wireless sensor networks. *J. Netw. Comput. Appl.* **2014**, *40*, 270–278. [CrossRef]
97. Capone, S.; Brama, R.; Accettura, N.; Striccoli, D.; Boggia, G. An Energy Efficient and Reliable Composite Metric for RPL Organized Networks. In Proceedings of the 2014 12th IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Milano, Italy, 26–28 August 2014; pp. 178–184.
98. Gaddour, O.; Koubaa, A.; Baccour, N.; Abid, M. OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol. In Proceedings of the 2014 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), Hammamet, Tunisia, 12–16 May 2014; pp. 365–372.
99. Kamgueu, P.O.; Nataf, E.; Ndie Djotio, T. On design and deployment of fuzzy-based metric for routing in low-power and lossy networks. In Proceedings of the 2015 IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops), Clearwater Beach, FL, USA, 26–29 October 2015; pp. 789–795.
100. Araujo, H.D.S.; Filho, R.H.; Rodrigues, J.J.P.C.; Rabelo, R.D.A.L.; Sousa, N.D.C.; Filho, J.C.C.L.S.; Sobral, J.V.V. A Proposal for IoT Dynamic Routes Selection Based on Contextual Information. *Sensors* **2018**, *18*, 353. [CrossRef]
101. Chen, Y.; Chanet, J.P.; Hou, K.M.; Shi, H.; de Sousa, G. A Scalable Context-Aware Objective Function (SCAOF) of Routing Protocol for Agricultural Low-Power and Lossy Networks (RPAL). *Sensors* **2015**, *15*, 19507–19540. [CrossRef] [PubMed]

102. Gozuacik, N.; Oktug, S. Parent-Aware Routing for IoT Networks. In *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*; Springer: Cham, Switzerland, 2015; pp. 23–33.
103. Oliveira, L.; Rodrigues, J.J.P.C.; Kozlov, S.A.; Rabêlo, R.A.L.; Albuquerque, V.H.C.d. MAC Layer Protocols for Internet of Things: A Survey. *Future Internet* **2019**, *11*, 16. [CrossRef]
104. Hahm, O.; Baccelli, E.; Petersen, H.; Tsiftes, N. Operating Systems for Low-End Devices in the Internet of Things: A Survey. *IEEE Internet Things J.* **2016**, *3*, 720–734. [CrossRef]
105. Musaddiq, A.; Zikria, Y.B.; Hahm, O.; Yu, H.; Bashir, A.K.; Kim, S.W. A Survey on Resource Management in IoT Operating Systems. *IEEE Access* **2018**, *6*, 8459–8482. [CrossRef]
106. Baccelli, E.; Hahm, O.; Gunes, M.; Wahlisch, M.; Schmidt, T.C. RIOT OS: Towards an OS for the Internet of Things. In Proceedings of the 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Turin, Italy, 14–19 April 2013; pp. 79–80. [CrossRef]
107. Antmicro. RENODE. Available online: https://renode.io/ (accessed on 4 June 2018).

Chapter 3

A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

This chapter consists in the following paper:

A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

José V.V. Sobral, Joel J.P.C. Rodrigues, Ricardo A.L. Rabêlo, José C. Lima Filho, Natanael Sousa, Harilton S. Araujo, and Raimir Holanda Filho

Journal of Network and Computer Applications, Elsevier, ISSN: 1084-8045, 2018.

DOI: doi.org/10.1016/j.jnca.2018.01.015

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 5.570

ISI Article Influence Score (2019): 0.800

Journal Ranking (2019): Q1 - 6/53 (Computer Science, Hardware & Architecture)

Journal Ranking (2019): Q1 - 8/109 (Computer Science, Interdisciplinary Applications)

Journal Ranking (2019): Q1 - 7/108 (Computer Science, Software Engineering)

Chapter 3. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

Journal of Network and Computer Applications 107 (2018) 56–68

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

José V.V. Sobral [a,b], Joel J.P.C. Rodrigues [a,c,d,e,*], Ricardo A.L. Rabelo [f], José C. Lima Filho [f], Natanael Sousa [f], Harilton S. Araujo [d], Raimir Holanda Filho [d]

[a] Instituto de Telecomunicações, Universidade da Beira Interior, Covilhã, Portugal
[b] Federal Institute of Maranhão (IFMA), São Luís, Maranhão, Brazil
[c] National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, MG, Brazil
[d] University of Fortaleza (Unifor), Fortaleza, CE, Brazil
[e] ITMO University, Saint-Petersburg, Russia
[f] Federal University of Piauí, Ininga, Teresina, PI, Brazil

ARTICLE INFO

Keywords:
Fuzzy system
Internet of Things
Performance
Radio frequency identification
Wireless sensor networks

ABSTRACT

Internet of Things (IoT) has emerged as a new paradigm that allows different objects interconnection to create new smart services and applications. In this sense, key features such as traceability, unique identification, energy efficiency, heterogeneity of devices, scalability and ubiquity, are necessary at the network structure for efficient performance of IoT applications. As a single technology, it is not capable to provide all these features, then, the integration of Wireless Sensor Networks (WSNs) and Radio Frequency Identification (RFID) arises as an important approach for these solutions. In this paper, a framework aiming to provide the network features required for IoT applications is proposed, considering the challenges that comes from the integration of these two technologies. In this framework, two components are considered, the Fuzzy Q-Algorithm and the Fuzzy System-Based Route Classifier. The former comprises a fuzzy system to enhance an anti-collision protocol for RFID, and the latter is a fuzzy system that classifies routes and assists routing protocols in applications that use WSNs and RFID. Experiments show that framework provides the following benefits: improves the performance of RFID tags identification, lowers packet loss rates, decreases the nodes energy consumption, and shows improvements in network load balancing. Moreover, these benefits provide greater energy efficiency and quality of service for IoT applications.

1. Introduction

Internet of Things (IoT) is considered the biggest revolution in the industry of information and communication technologies by letting the pervasive presence of a variety of things/objects (Radio Frequency Identification tags, henceforth named RFID tags, sensors, actuators, smartphones, etc.) in a network, which are able to communicate and interact with each other without human intervention in order to cooperate on a common purpose (Al-Fuqaha et al., 2015; Stankovic, 2014; Gandotra et al., 2017). In order to turn the IoT paradigm a reality, it is necessary that network structure supports some key features, such as (Miorandi et al., 2012): heterogeneity; scalability; energy optimization; minimizing costs, features such as self-organization, self-adaptation, and self-reaction; privacy, security, and quality of service (QoS). To this end, two technologies are considered to play an important role on IoT (Atzori et al., 2010; Sheng et al., 2013; Agrawal and Das, 2011), RFID and Wireless Sensor Networks (WSNs). In particular, RFID provides platforms to identify every object in IoT environments (Fernández-Caramés et al., 2016; Bashir et al., 2017). On the other hand, as the smart objects can sense physical objects and the environment, WSNs should be responsible for data acquisition from the surroundings (Lin et al., 2017).

The integration of RFID and WSN technologies constitutes a promising solution to explore improved performances and new applications for IoT (Díaz et al., 2016; Al-Fuqaha et al., 2015). However, it stills some challenges to be overcome in order to achieve full integration of these two technologies into the so-called RFID-Sensor Networks (RSN),

* Corresponding author. National Institute of Telecommunications (Inatel), Av. João de Camargo, 510 – Centro, 37540-000, Santa Rita do Sapucaí, MG, Brazil.
E-mail addresses: jose.sobral@it.ubi.pt (J.V.V. Sobral), joeljr@ieee.org (J.J.P.C. Rodrigues), ricardoalr@ufpi.edu.br (R.A.L. Rabelo), jose.correia@live.estacio.br (J.C. Lima Filho), natanaelsousa@ufpi.edu.br (N. Sousa), harilton@edu.unifor.br (H.S. Araujo), raimir@unifor.br (R. Holanda Filho).

https://doi.org/10.1016/j.jnca.2018.01.015
Received 27 July 2017; Received in revised form 22 January 2018; Accepted 26 January 2018
Available online 6 February 2018

Chapter 3. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

and comply with the required characteristics for IoT applications.

The authors in (Mainetti et al., 2016) propose an IoT smart parking system application that uses the integration of different technologies (among these RFID and WSN) to collect, in real time, information about the use of parking space. The proposal uses a 6LLR (6LoWPAN Router Reader) node that combined with a 6LoWPAN node with an RFID tag reader. The results demonstrate the great potential of the proposal but were not realized studies to verify the performance of the network created.

In (Alfian et al., 2017) the authors present a proposal of integration between RFID and WSN for an e-pedigree food traceability system. The RFID system is responsible for tracking and identifying the products while WSN is used to collect environmental data. The integration of RFID and WSN happen in a web service that receives data from both technologies. Thus, RFID and WSN are not combined in a unique device. The authors have analyzed the network performance and noted that the proposed system is scalable, but the RFID network results are fully independent of the WSN results once they run separately.

A framework to provide tolerance to delays and optimized deployment of nodes in RFID-Sensor Networks (RSN) for IoT applications is proposed in (Al-Turjman et al., 2013). This framework considers two components that group different types of nodes and performs data routing. This routing technique is tolerant to delays. However, it does not consider the energy levels of the nodes and the quality of routes. In (Yang et al., 2007), the authors propose an integration model of RFID with WSNs, which maximizes the number of RFID readers. Therein the RFID readers are equipped with transducers that enable multi-hop wireless communication among readers and a sink node. The results obtained by simulation show greater load balancing. However, no metric includes the packet loss rate or the interference levels caused by the increased number of readers/transducers. Thus, the range of applications for this model is reduced, as the network cannot sense environmental data. In (Andreou et al., 2014), the authors present KSpot$^+$, a distributed framework for WSNs that seeks to decrease the inefficiencies of data reception and transmission, and reduce the packets size and the number of packets exchanged in the network. The framework is composed of three components that construct a balanced network topology, control the time of the sensor nodes monitoring, and facilitate the use of advanced queries. The performance evaluation results show that this framework can reduce the energy consumption and increase the network lifetime. Finally, in (Rajesh, 2013), a model of integration of WSN and RFID systems with active tags is designed for patients monitoring in hospitals. These active tags capture information from a patient, such as, body temperature and heart rate. However, this work does not investigate measurements of the prototype performance.

This paper proposes a framework suitable to provide relevant network features required for IoT applications. This framework is composed by two main components, namely, Fuzzy Q-Algorithm and Fuzzy System-Based Route Classifier. The former comprises a fuzzy system to integrate an anti-collision protocol for RFID, which improves the tags reading system turning it faster and more efficient. The latter comprises a fuzzy system to classify routes and assist routing protocols in applications that use WSN and RFID, which increases the performance of data exchange between nodes by reducing the packet loss rate, the load imbalance, and the energy consumption. Thus, the novelty provided by the proposed framework is to make available a set of mechanisms to improve the performance of a network that integrates RFID and WSN, once that the current integration approaches are focused just in the combination of the technologies without considering the network performance as a whole. Then, the main contributions of this work are the following:

- Reducing the energy consumption of the network components, decrease the quantity of packet loss and improve the network load balance using routing techniques based on computational intelligence that uses information about the network the estimate the route quality in a most efficient fashion;
- Increasing the tag identification speed and the query success rate during the tag reading process using a novel anti-collision protocol able to use the knowledge of human specialist related to the functioning of an RFID system.
- Providing an optimized energy solution and enhance the data exchange among ubiquitous wireless technologies involved in the IoT network using an improved routing protocol and an efficient tag reading process.
- Improving the performance and the tracking and location capacity of the tags in the IoT applications executed on networks that integrate WSN and RFID.
- Providing a most reliable network structure for the data transport of things and objects collected by the nodes that compose the IoT application.

The rest of the paper is organized as follows. Section 2 describes the fundamentals regarding WSN and RFID technologies and Section 3 describes the integration of RFID and WSNs for IoT. Section 4 describes the proposed framework and the experimental results are discussed at Section 5. Main conclusions are provided in Section 6.

2. Technical background

This section briefly discusses some fundamentals regarding WSN and RFID technologies, as well as some types of integration already presented in the literature.

2.1. RFID

Radio Frequency Identification (RFID) is a technology composed by different electronic devices with wireless communication capabilities to perform automatic identification tasks (Roberts, 2006). Thus, these devices have been widely used in tracking and access applications. RFID systems consider one or more readers and various RFID tags. During the full operation of an RFID system, a reader sends requests to the tags, which are attached to objects or people. Immediately after that, the tags answer with a message containing their ID. However, sending simultaneous responses from tags may cause data packet collisions (Klair et al., 2010). Therefore, the consideration of collisions is paramount when evaluating RFID systems (Kabir et al., 2015).

2.2. Wireless sensor networks

Wireless sensor networks (WSNs) enable applications development with monitoring capabilities (Gluhak et al., 2011; Perera et al., 2014). In this sense, a WSN can be defined as a collection of sensor nodes with limited capacity of storage, processing, and energy (Rashid and Rehmani, 2016). These sensor nodes acquire the data from the environment in which they operate, and can transmit the acquired data to other network nodes via wireless communications (Akyildiz et al., 2002; Hu and Cao, 2010). In addition, executing a collaborative function, it allows sensor nodes to provide data to be processed by sink nodes (Yick et al., 2008). However, reducing energy consumption in these sensor nodes represents one of the major challenges in WSN (Lin et al., 2012) because, in most cases, these nodes are deployed in a location with hard access turning it infeasible to replace the batteries that feed them (García-Hernández et al., 2007; Kulkarni et al., 2011).

3. RFID and WSNs integration

RFID systems are fast and convenient for the identification of the objects, on the other hand, WSN nodes can cooperate for collecting and distributing data messages in a multi-hop network. However, when compared with RFID systems and WSN individually, these integrated technologies present a highly promising potential. The integration of

Chapter 3. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

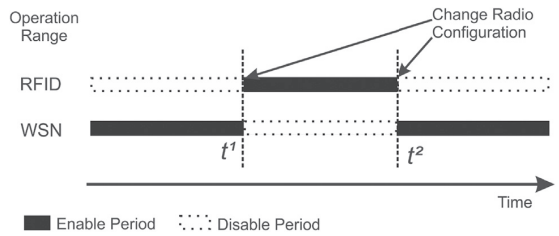


Fig. 1. RS-SDR architecture.

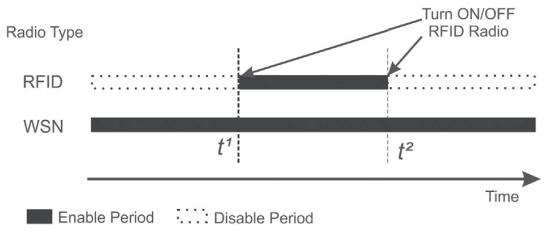


Fig. 2. RS-DR architecture.

RFID and WSNs brings a range of opportunities to exploit the advantages of both technologies to expand the applications for IoT.

Despite the benefits, the result of this integration can provoke several technicals, politics and operational challenges when compared with the involved technologies applied isolatedly (Luo et al., 2011). In the literature, four types of integration of RFID and WSN systems are proposed in (Liu et al., 2008), namely (Hussain et al., 2009; López et al., 2009; Abahsain et al., 2013):

1. Sensing units combined with RFID tags;
2. Sensor nodes interconnected to RFID tags;
3. Sensor nodes connected to RFID readers;
4. Set of RFID components and sensors integrated by an application.

Based on the aforementioned types of integration, this paper will consider the one in which RFID readers are connected to wireless sensor nodes (type 3) because this type, in particular, can meet a wide variety of IoT applications while keeping a low deployment cost. The issues inerrant to this kind of integration are directly related to the increased energy consumption provoked by additional power requirement of RFID components in the node, and the necessity of specific protocols aware of the node condition to better adjust the use of the hardware resources.

In this work, this integrator device that combines the features of a wireless sensor node with the capabilities of an RFID reader will be named Reader Sensor (RS) node. Thus, the device can, besides provide environmental sensing, offer the possibility of tracking and identification of RFID tags sending the collected data to a central node through multi-hop communication. The RS nodes can have two different architectures. In the first, RS node is equipped with just one communication interface. In the second, the RS node has two independent communication interfaces. Both architectures are presented in the following subsection.

3.1. RS nodes with software defined radio architecture

The RS nodes with Software-Defined Radio (RS-SDR) architecture use only one communication interface for exchanging messages with both WSN and RFID elements. The SDR is a radio that includes a transmitter whose operating parameters, such as frequency, modulation type, or maximum output energy can be changed via software without making any changes on hardware components responsible for radiofrequency transmission (Dillinger et al., 2005; Tuttlebee, 2003). The simultaneous communication with sensor nodes and RFID tags is not possible when this type of node is used, as shown in Fig. 1. Thus, when the reader node is collecting information from tags, it is unable to receive messages from other sensor or reader nodes available in the network.

3.2. RS nodes with Dual Radio architecture

The RS nodes with Dual Radio (RS-DR) architecture have two independent communication interfaces. The first communication interface is used to exchange messages between RS nodes and sensor nodes. A CC2420 radio (T. Instruments, 2420) can be used to perform this task. The second interface is responsible for the communication between RS nodes and RFID tags. Depending on the specifications of the tags to be identified, a CC1000 radio (T. Instruments, 1000) can be used to perform the reader-tag communication. The RS-DR can communicate with the elements of the WSN and the RFID at the same time because it has two independent communication interfaces. The usage of different communication interfaces of RS-DR nodes is shown in Fig. 2.

4. Proposed framework

The proposed framework is based on two components capable to improve the performance of networks that integrate WSNs and RFID for IoT applications. The first component, called the Fuzzy Q-Algorithm (FQA), addresses the acquisition of data from RFID elements by the application. It uses an anti-collision protocol for RFID tags based on the Electronic Product Code (EPC) Class 1 Generation 2 Global Standard (EPCglobal, 2008). The second component, called the Fuzzy System-Based Route Classifier (FSBRC), deals with an existing routing protocol to increase the performance of the message-delivery service. Both components of framework are based on computational intelligence techniques.

Fig. 3 illustrates a scenario which integrates RFID and WSNs for IoT applications where the proposed framework can be used. Thus, it includes the following elements:

- Reader-sensor nodes – they are elements that join RFID reader and sensor nodes functionalities in a WSN. Their task includes data collection stored in RFID tags, sensing the surrounding environment, and provide multi-hop communication with other network elements. The RS nodes can have a DR or an SDR architecture;
- Sensor nodes – they are elements that sense the environment and provide multi-hop communication with other network nodes. Considering that in a potential application some nodes do not need to realize the reading task, these nodes do not have the ability to collect information from RFID tags. However, they serve to collect environment data and as a route to convey data from tags obtained by an RS node;
- RFID tags – these elements identify things or objects and/or persons that have an ID. They communicate with RS nodes to inform the data recorded in their memory;
- Sink Node – it is a differentiated network element that usually does not present energy constraints. These nodes have greater storage and processing power to handle the information acquired by the other nodes. In order to share the data acquired by the network, Sink Node can be connected to the Internet, then it acts as a gateway (Zhu et al., 2010).

Following, the both above-mentioned components are explained in detail.

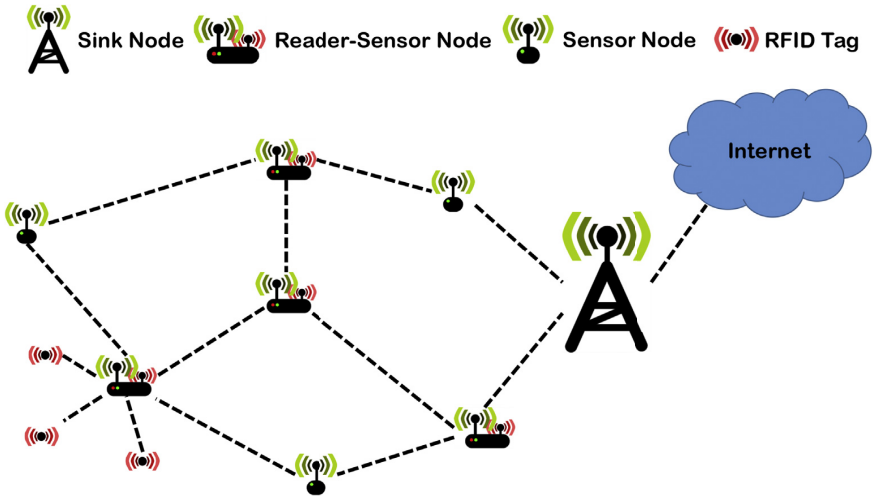


Fig. 3. Illustration of a network scenario that integrates WSN and RFID.

4.1. Fuzzy Q-Algorithm

The high number of packets exchanged between readers and RFID tags can affect the scalability of the network as well as the QoS required by IoT applications (Welbourne et al., 2009). In particular, QoS is affected by the packet loss commonly caused by collisions among simultaneous messages sent by tags. In order to minimize the problems caused by these collisions, RFID readers use anti-collision protocols to coordinate the sending of response messages from tags (Klair et al., 2010).

One of the most commonly used RFID anti-collision protocols is the EPC Global UHF Class-1 Generation-2 Standard (C1–G2) (EPCglobal, 2008; Wang et al., 2009). According to the specification of the EPC C1–G2 protocol, the tag identification process considers an inventory formed by several rounds (EPCglobal, 2008). The EPC C1–G2 does not specify the criteria to complete the inventory and start another one, although several authors consider the end of an inventory when all tags in the region are successfully identified (Zhu and Yum, 2011; Kim et al., 2007; Baloch and Pendse, 2013). The EPC C1–G2 protocol is based on slotted ALOHA protocol, in which tags send their data on specific slots (Bueno-Delgado and Vales-Alonso, 2011).

Fig. 4 shows the operation of the EPC C1–G2. An inventory is based on several rounds which, in turn, includes multiple slots. Upon receiving a QUERY command, the tags use the Q value to define its slots. The tags also define a slot counter that receives a random value between 0 and $2^Q - 1$. Each time a tag receives a QUERY_REP or QUERY_ADJUST command, it reduces its slot counter. When a tag slot counter reaches 0 after receiving a QUERY_REP or QUERY_ADJUST command, the tag must send an RN16 message (contains a 16-bits random number generated by the tag) to the reader. If two or more tags send its RN16 in the same slot, most probably will happen a collision among the messages and they may be lost. In this case, the slot is considered a collision slot S_c (Fig. 4a). When no tag sends RN16 in a slot, namely, no tag reaches the slot count value equals to 0, this slot is considered an empty slot S_e (Fig. 4b). Otherwise, if just one tag sending the RN16 message, the reader receives the RN16 and answer the tag with an ACK message. The ACK message must contain the 16-bit number obtained by the reader inside of the RN16 message. After receiving the ACK message, the tag must send its unique identification number (EPC code) (Fig. 4c). If the reader received the EPC successfully, a new command message is sent to continue the identification process.

The Q Algorithm is used by the EPC C1–G2 to fit the slots number in each round and reduce the number of collisions among messages sent by tags (Namboodiri et al., 2012). The idea of this algorithm is to change the protocol parameters according to the environmental conditions. Fig. 5 illustrates the operation of the Q-Algorithm. Therein,

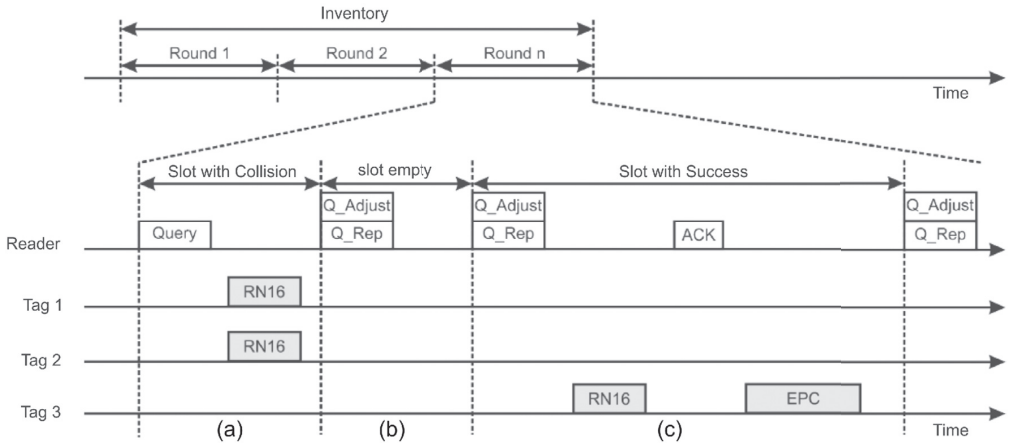


Fig. 4. EPC C1–G2 identification process.

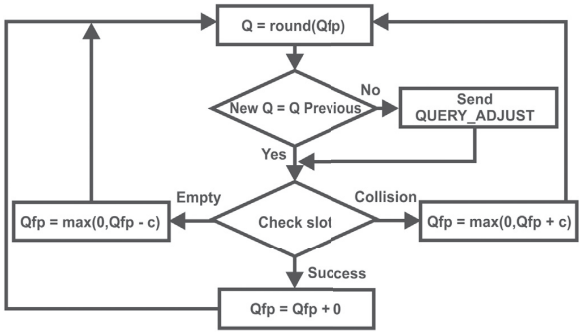


Fig. 5. Q-Algorithm operation.

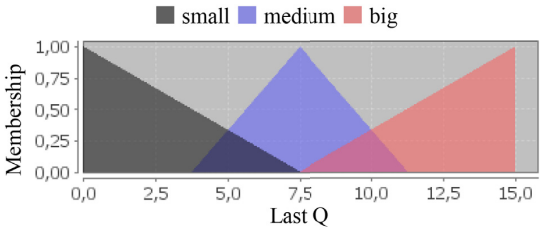
Q_{fp} consists in a floating-point representation of the Q value. The Q_{fp} value is increased or decreased, at each slot, based on a variable c, where $0.1 < c < 0.5$. If there is any collision on the slot, the Q_{fp} value is increased by c. If the slot is empty, the Q_{fp} value is decreased by c. Otherwise, in case of success in the slot, the Q_{fp} value is not changed. In all cases, the Q_{fp} value may not be higher than 15 or lower than 0. After each slot, the Q_{fp} value is rounded to its nearest integer and compared to Q. If the values are different, the reader sends a *QUERY_ADJUST* command to adjust the Q value of the tags. However, if the value of c is too small, the Q adjustment process can be very time-consuming. On the other hand, if the value of c is high, the Q value will be changed rapidly and many adjustment messages are then required. In a scenario with few tags, where the value of Q is initially high, the c value must be high so the number of slots is readily suited to the amount of tags. Likewise, with an initially small Q in a setting with many tags, the value of c should be high enough to raise the number of slots. In IoT applications, there can be scenarios in which the amount of tags (fixed to things) is very high at a time and very low shortly after. This type of scenario is common in applications with mobile things (DaCosta, 2013). Furthermore, some proposals of integration of WSNs and RFID (Al-Turjman et al., 2013), (Yang et al., 2007; Zhang and Wang, 2006; Jurdak et al., 2008; Al-Turjman et al., 2012) do not consider aspects of collisions when reading tags. However, an inefficient tag reading system can compromise the performance of IoT applications, as the delay in the reading process caused by collisions may increase energy consumption and results in premature death of RS nodes.

Based on this premise, the proposed framework has a component responsible for improving the performance of the tag reading process to be executed by the RS nodes, called the Fuzzy Q-Algorithm (FQA). The FQA is an RFID anti-collision protocol based on the EPC C1-G2 protocol and the Q-Algorithm that uses a computational intelligence technique (Engelbrecht, 2007) known as fuzzy system (Pedrycz and Gomide, 2007) to set the c parameter to the next reading process. The fuzzy system employed is capable of setting the output values for one or more variables using one or more input values and the knowledge of a specialist. Hence, the knowledge of the specialist will be represented as, *if … then*, decision rules. Also, the specialist defines the membership function for each one the input and output variables.

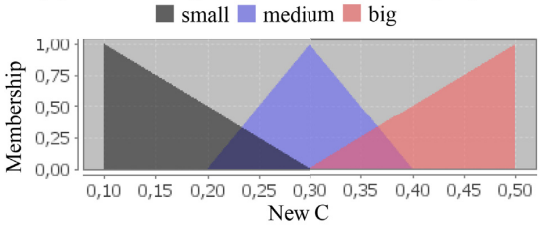
In the FQA, the c value is dynamically reset based on the Q value of the last reading process. Thus, the fuzzy system of FQA uses the last Q value, which varies between 0 and 15, as input to obtain a new c value, which ranges from 0.10 to 0.50, as output. In order not to increase the tag identification process time with the computation of fuzzy system, the calculation of a new c value is performed just at the end of each inventory. The rule base and the membership functions used in the fuzzy system of FQA as present in Table 1 and Fig. 6, respectively. The parameters of the fuzzy system of the FQA are presented in Table 2. Rule base, membership function and the another parameters of the fuzzy system was defined based on empirical studies developed during this work.

Table 1
FQA rule base.

	Input	Output
	Last_Q	New_C
Rule 1	Small	Small
Rule 2	Medium	Medium
Rule 3	Big	Big



(a) Membership function of the last Q input.



(b) Membership function of the new C output.

Fig. 6. Membership functions of the fuzzy system used in the FQA.

Table 2
Fuzzy system parameters.

Parameter	Value
Membership Function	Triangular
Implication Operator (T-Norm)	Minimum
Aggregation Operator (S-Norm)	Maximum
Defuzzifier	Centroid

Using the EPC C1–G2, every time a tag needs to be read, the entire identification process happens again, regardless of whether the tag has already been reported or not. Therefore, the tags send their EPC more than once to the same reader. This process of sending redundant data can increase the tags reading process time and can also increase the chances of collisions. Thus, the FQA presents a new tag-identification method that reduces the amount of redundant EPC codes being sent and the identification process time.

In the FQA, every time a tag is successfully identified, its EPC code is stored along with the RN16 sent by the tag. If a tag sends an RN16 that has been previously received, the reader sends an ACK with an additional checked flag to indicate that the tag has been recognized and does not need to submit its EPC. If the RN16 is new, the reader sends an ACK with an unchecked flag to request the tag's EPC. Notably, in the FQA protocol, the RN16 of tags does not change every time that an identification process starts. If a reader receives the same RN16 more than once during an inventory, the stored value is deleted, and the ACK message is sent with the flag cleared, which makes the tag send its EPC code. In addition, the table that stores the RN16 values and their EPCs can be emptied periodically to prevent the readers' memory from being occupied by tag data that are not in the reading area. Fig. 7 shows the operation of the FQA identification process, which is also described as follows:

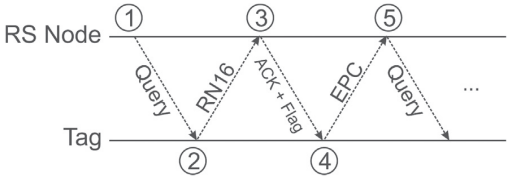


Fig. 7. Illustration of the FQA identification process.

- Step 1 – RS node sends a query command to tags. A query command can be QUERY, QUERY_REP or QUERY_ADJUST;
- Step 2 – similar to the operation of the EPC C1–G2, the tag, upon receiving a query command, decreases its slot count by 1. The tag sends an RN16 message when the slot count reaches 0;
- Step 3 – after receiving the RN16 message of the tag, the RS node checks whether the same RN16 has already been received and that tag has been identified. If so, an ACK message is sent with a checked flag. Otherwise, the ACK message is sent with the flag cleared;
- Step 4 – the tag, after receiving the ACK message, checks the status of the flag. If checked, this means that tag has already been reported and should no longer send its EPC code. If unchecked, the tag must send its EPC code;
- Step 5 – the RS node receives the EPC code of the tag and adds it to a cache near the RN16 previously sent by the tag. This cache is used to verify whether or not a tag has already been identified in Step 3. After that, a new query command is sent ushering in a new slot.

It is expected that the use of the FQA makes the tag reading process more efficient and, at the same time, reduces the time of this process. With the dynamic adjustment of c, it is intended to improve the reading rates of tags resulting from an appropriate Q value. By avoiding redundant EPC code from being sent, the time necessary for the tag identification is also expected to be reduced. Thus, the purpose is to improve the performance of IoT applications with a more efficient identification process.

4.2. Fuzzy System-Based Route Classifier

One of the major challenges related to WSNs refers to the problem of nodes energy consumption (Ramos et al., 2014). One way to minimize the energy consumption and, thus, increasing the network lifetime is by deploying new or adapting available routing protocols (AlShawi et al., 2012). In this context, various studies have proposed new routing solutions in WSN applications (Machado et al., 2013; Tunca et al., 2015; Kumar and Rai, 2017; Abdul-Salaam et al., 2017; Mann and Singh, 2017). In this sense, the FSBRC component uses a fuzzy system to classify routes and assist routing protocols in IoT scenarios that integrate WSNs and RFID. The fuzzy system used by the FSBRC is composed of four inputs and one output. The output, named Route Quality, represents the quality of a path between a sender and a destination based on several parameters (the inputs considered). The four inputs used to obtain the referred Route Quality are presented following:

- Energy level of the route – it is the lowest energy level among the nodes that compose the route;
- Number of hops – it is the counter that defines the number of nodes along the message has already sent. This metric is related to the distance (in hops) between network nodes;
- LQI (Link Quality Indicator) (Diallo et al., 2011; Zhao et al., 2014) – this metric is related to the communication quality between two nodes. For each route, it is considered the worst level of LQI among its nodes. The physical layer calculates the LQI whenever a message is received by a node;
- Tag density – represents the number of RFID tags that were read by the node at the last reading process.

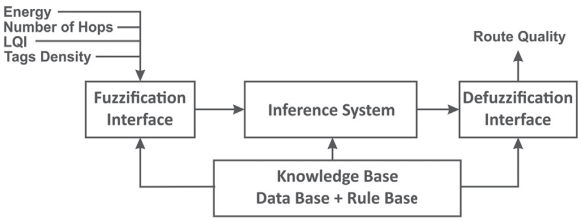


Fig. 8. Structure of the Fuzzy System-Based Route Classifier (FSBRC).

The FSBRC can be integrated into many multipath-based routing protocols (Radi et al., 2012). This kind of protocol has the feature of generating several routes between a sender and a destination. Thus, aiming to reduce the energy consumption and improve the QoS, the FSBRC can be used to select the best route among those available ones. In the same way, the FSBRC can also be adapted to assist others routing protocol that accepts the modification of the routing metrics like RPL (IPv6 Routing Protocol for Low-power and Lossy Networks) (Winter et al., 2012) and LOADng (Lightweight On-demand Ad hoc Distance-vector Routing Protocol - next generation) (Clausen et al., 2016).

In this paper, the proposed framework uses the FSBRC component combined with the Directed Diffusion protocol (DD) (Intanagonwiwat et al., 2003). The DD was chosen because it is one of the most used query-based routing protocol and it is frequently used as a base for emerging new approaches (Chen et al., 2006; Peng and Low, 2015). Therefore, the DD protocol uses the FSBRC mechanism when receiving a message of interest. The sequence of steps performed by the nodes when receiving a message of interest is described as follows:

- Start – as soon as a message is received, the node's physical layer calculates the LQI value that indicates the communication quality between the sender and the recipient. The LQI value is resent to the routing layer (after passing through the MAC layer) together with the received message and the sender node ID;
- Step 1 – after receiving the message of interest in the routing layer, the node checks whether the value of the LQI, calculated when receiving the message, reduces the value of the LQI contained in the message. If positive, the value of the message LQI is then updated with the LQI value calculated when receiving the message. This update is necessary so the message can carry the worst level of LQI of the route;
- Step 2 – the node obtain the LQI data, number of hops, energy, and tag density contained in the received message;
- Step 3 – the node uses the LQI values, number of hops, energy, and tag density to infer the quality of the route by using the FSBRC mechanism (Fig. 8);
- Step 4 – the node checks whether there is an entry in its cache for the received message of interest. If it is true, the cache is then updated and it checks whether to create a new gradient or update an existing one. Subsequently, the message is discarded and no action is taken;
- Step 5 – if the message of interest has not been received yet, the node adds it to the cache along with the route quality information and the sender node ID;
- Step 6 – the node checks whether its residual energy is lesser than that contained in the message. If so, the node changes the message energy value with its residual energy value. Otherwise, the value is unchanged;
- Step 7 – the node checks whether its tag density is greater than that contained in the message. If so, the node changes the tag density value of the message with the number of tags read in the last identification process. Otherwise, the value is not changed. If the node is not a reader-sensor, the tag density value is 0;
- Step 8 – the LQI value, previously recorded, is used to update the LQI value of the message, the number of hops of the message is increased, the value of the message sender ID is changed to the node

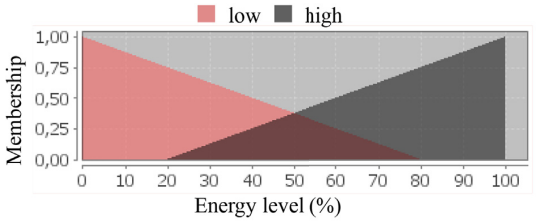
ID, and the message is broadcast.

After the formation of the routes, and before sending a message, the node should select the route with the best quality by observing the information contained in its cache. In a fixed time interval, the network nodes exchange control messages to update the values of the energy level, LQI, and tag density. By using the control messages, it is possible to trace the route qualities, which can be useful for a precise estimation of the route quality.

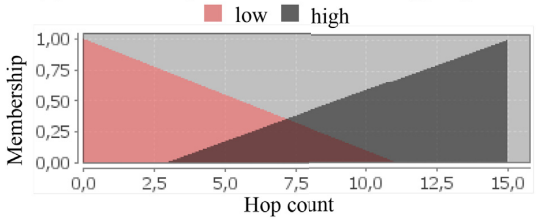
The usage of computational intelligence for improving routing protocols is not novel. In (Zungeru et al., 2012; Guo and Zhang, 2014), the authors present survey studies about using computational intelligence techniques for performing the routing task in a WSN. In the proposed framework, an Ant Colony Optimization (ACO) algorithm (Ant System (Dorigo et al., 1996)) was used to improve the design of the fuzzy system rule base present in the FSBRC. In a fuzzy system, the rule base is responsible for mapping the input and output domains, so it plays an important role in generating the results produced by the inference process (Sun, 1994). ACO is also used in different fields of WSNs. In (Liu and He, 2014), the authors propose an ACO algorithm to decrease the deployment cost of WSNs. In (Liao et al., 2008), an ant colony algorithm is used for data aggregation to reduce the network energy consumption.

The FSBRC system uses four input and one output variables, whose membership functions are presented in Fig. 9. Each input variable has two primary terms (or linguistic terms), creating a rule base with sixteen rules. The output variable has five primary terms. In this way, for each rule, five options of linguistic terms are available. Thus, it is possible to create 5^{16} (152.587.890.625) combinations of rule bases. Thus, due to the very hard issue of defining a correct rule base, the task of the ACO algorithm is to map the search space and to use artificial ants to find the optimal (or near optimal) rule base configuration based on the initial rule base presented in Table 3. The parameters of the fuzzy system used in the FSBRC are the same of FQA presented in Table 2.

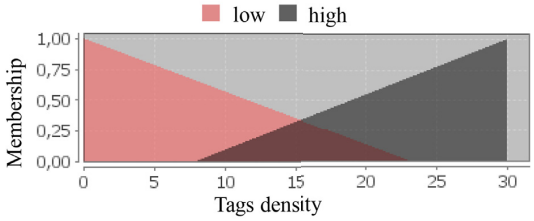
It is worth mentioning that the FSBRC adjustment process is carried out through computer simulations before the network is executed in order to define the best formation of the decision rules' database. The objective to use the FSBRC mechanism for this framework is to improve the performance of the routing protocol, thus providing better network load balancing, reducing energy consumption, and reducing packet loss rate. Considering this, it is expected that the requirements for an improved solution in terms of energy and data exchange between ubiquitous technologies of IoT applications are more efficiently attained.



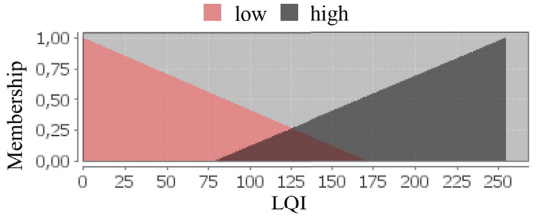
(a) Membership function of the energy input.



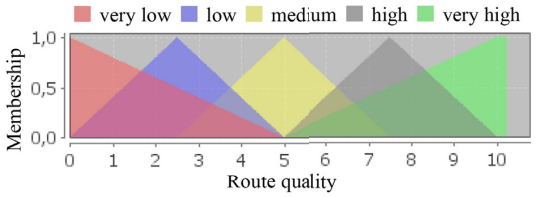
(b) Membership function of the hop count input.



(c) Membership function of the tags density input.



(d) Membership function of the LQI input.



(e) Membership function of the route quality output.

Fig. 9. Membership functions of the fuzzy system used in the FSBRC.

5. Results analysis and discussion

The process of validation and performance evaluation of the proposed framework was done through computational simulations. Hence, the Castalia simulator (Boulis, 2007), a network simulator specific for WSNs and BANs (Body Area Networks) was employed. Although Castalia does not support RFID applications by default, it has an open source code that was extended to make simulating the behavior of tags and RFID readers possible. Thus, a new module that supports simulations of RFID networks was developed, which enabled the simulation of heterogeneous environments for IoT applications. In order to validate the developed RFID module, it was compared to RFID systems analysis available in the literature. The results of the developed framework were similar to analytical and simulated results presented in (Wang et al., 2009; Baloch and Pendse, 2013).

The performed experiments are presented in two steps. The first one exposes the performance evaluation of FQA comparing it with other anti-collision approaches existing in the literature. The second step aims to measure the performance of the proposed framework in a complete fashion. The simulations consider different scenarios, changing the type of RS nodes with DR and SDR architectures, the use of the proposed framework, and the number of RS nodes. The two simulation steps are presented as follows.

5.1. FQA performance evaluation

In the FQA performance evaluation, the simulated application has the task of identifying all the tags in a certain environment. The objec-

Table 3
FSBRC initial rule base.

	Input				Output
	Energy	Hop Count	LQI	Tag Density	Route Quality
Rule 1	Low	Low	Low	Low	Medium
Rule 2	Low	Low	Low	High	Low
Rule 3	Low	Low	High	Low	High
Rule 4	Low	Low	High	High	Medium
Rule 5	Low	High	Low	Low	Low
Rule 6	Low	High	Low	High	Very Low
Rule 7	Low	High	High	Low	Medium
Rule 8	Low	High	High	High	Low
Rule 9	High	Low	Low	Low	High
Rule 10	High	Low	Low	High	Medium
Rule 11	High	Low	High	Low	Very High
Rule 12	High	Low	High	High	High
Rule 13	High	High	Low	Low	Medium
Rule 14	High	High	Low	High	Low
Rule 15	High	High	High	Low	High
Rule 16	High	High	High	High	Medium

Table 4
Parameters for the FQA experiments.

Parameter	Value
Simulation area	10 m × 10 m
Time of simulation	1 min
Reader location	(5,5)
Number of tags	50, 100, 200, 300, 400, 500

tive of this assessment is to measure the performance of FQA compared with other anti-collision protocols in a scenario without wireless sensor nodes. The simulations consider four anti-collision protocols: the proposed FQA, the standard C1G2, the *FastQ* (Teng et al., 2010), and the *Q+* (Lee et al., 2007).

Table 4 presents some parameters of the simulated environment. The reader is placed in the center of the environment while the tags are randomly deployed and have mobility characteristics according to Random Waypoint algorithm (Schindelhauer, 2006). Two metrics are employed to measure the performance of the FQA: Query Success Rate and Tag Identification Speed. Next subsections show how these metrics are computed and shows the results obtained in the simulated scenarios to each one.

5.1.1. Query success rate - QSR

The Query Success Rate (QSR) is defined as the rate of a query command (*QUERY*) can successfully identify a tag (Wang et al., 2009). The QSR is obtained by eq. (1).

$$QSR = \frac{\sum_{i=1}^{K} X_i}{\sum_{i=1}^{K} Y_i}, \tag{1}$$

where K is the number of inventories, Y_i represents the number of query commands used during inventory i, and X_i is the number of tags identified successfully during inventory i. Since the QSR provides information about the quality of the tag reading system, a higher QSR value indicates a more efficient tag identification. Thus, the improvement of the reading process can benefit the IoT applications.

Fig. 10 showcases the results obtained through the experiments for the QSR metrics. Considering the evaluated scenarios, the proposed FQA was able to present a higher success rate in its queries when compared with other approaches. This improvement is justified by using a fuzzy system to provide the dynamic and automated definition of the c parameter that allows the suitable adjustment of the number of rounds. Thus, FQA can increase the efficiency of the tag reading system culminating in an improved data exchange between reader and tags, and a boosted tracking capacity for IoT applications.

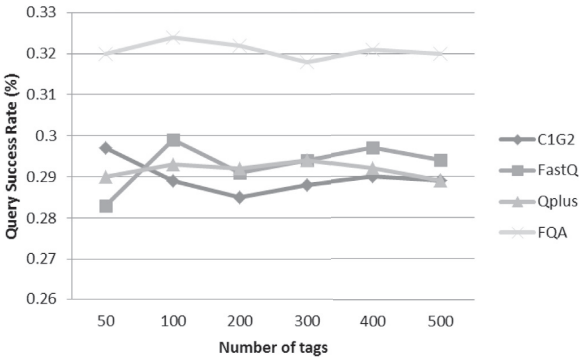


Fig. 10. Query Success Rate between RS nodes and tags.

5.1.2. Tag identification speed - TIS

The Tag Identification Speed (TIS) is a metric that defines the speed that tags are successfully identified (Wang et al., 2009). TIS represents the quantity of tags that a RS node can successfully identify in a time interval, assuming the number of tags to be identified is enough. For example, if the TIS is 100, it means that the RS node can read 100 tags per second. TIS is obtained by eq. (2).

$$TIS = \frac{\sum_{i=1}^{K} X_i}{\sum_{i=1}^{K} T_i}, \tag{2}$$

where K is the number of inventories (or tag reading process), X_i represents the number of tags successfully identified during inventory i, and T_i is period (in seconds) for the i-th inventory. Low TIS indicates that some tags are probably leaving the range area of the RS node without being identified, which is crucial in cases where tags are in the reading area for very short periods. Therefore, low TIS can compromise the performance of IoT applications, turning them less reliable and with long time for tag identification.

Fig. 11 shows the results obtained in the experiments for the TIS metric. The results show that the use of the FQA, when compared with the other approaches, increase the tag identification speed and, as a consequence, reduce the time spent on the identification process. One of the main features of the FQA is decreasing the number of exchanged messages between tags and reader. As it avoids sending redundant messages, the FQA reduces the time spent to identifying a tag successfully. Thus, the tag identification speed is boosted since the time to finish the reading process is reduced. As a benefit, the process of tracking and tags identification is improved making the application more efficient.

The results obtained in the first step of the experiments expose that

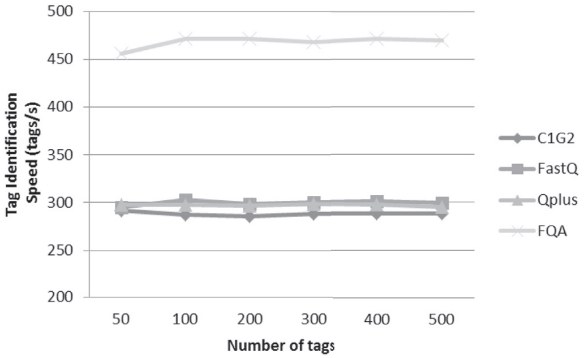


Fig. 11. Tag identification speed.

the proposed FQA has a more consistent performance when compared with other approaches. Thus, the experiments performed at the second step considers just the FQA and the standard C1G2 in the simulated scenarios.

5.2. Performance evaluation of the proposed framework

In the performance evaluation of the proposed framework, the IoT application executed on the network is addressed by queries (query-driven) (Hu and Cao, 2010) and aims to identify the tags that are in the reading area of the RS nodes, as well as the local conditions of the tags. Although the proposed framework can be used in both RS nodes and sensor nodes, only RS nodes were used on these experiments. The RS nodes were set randomly in the simulation area and have no mobility capacity. On the other hand, the tags (representing the "things") are initially placed in random locations of the simulation area, and they move according to the Random Waypoint algorithm (Schindelhauer, 2006). This mobility algorithm was adopted for its simplicity and for being one of the most widely used in simulations of wireless networks (Camp et al., 2002).

All the data collected by the RS nodes are sent to the sink node, which is used as a gateway for communication with the Internet. Therefore, the simulations take into account the heterogeneity of the devices, the data exchange between ubiquitous wireless technologies, traceability, and location, which are important features of IoT applications.

5.2.1. Scenarios description

The following scenarios are considered:

- Scenario 1 (SC 1) – the network uses RS nodes with DR architecture, Directed Diffusion routing protocol and EPC C1–G2 anti-collision protocol;
- Scenario 2 (SC 2) – the network uses RS nodes with SDR architecture, Directed Diffusion routing protocol and EPC C1–G2 anti-collision protocol;
- Scenario 3 (SC 3) – the network uses RS nodes with DR architecture and the proposed framework. Thus, the FSBRC is used with Directed Diffusion protocol and the FQA is the anti-collision protocol;
- Scenario 4 (SC 4) – the network uses RS nodes with SDR architecture and the proposed framework. Hence, once again, the FSBRC is used to help the Directed Diffusion protocol and the FQA is the anti-collision protocol;

Table 5 presents parameters common to all the experiments scenarios. These values are used for all the simulations, regardless the metric or evaluated scenario. The simulation area is about 50 × 50 m, so that each experiment has a duration time about 600 s. The sink node that receives all data collected by the nodes is located at the center of the simulation area (25, 25). The other nodes and tags are randomly distributed. The number of RS nodes composing the network changes between 20, 40, 60, 80, and 100 nodes. The initial energy of the RS nodes is about 100 J. All the scenarios have 50 tags. The sink node is responsible for sending requests to random points in the network, and each request lasts 200 s. Thus, the nodes in the query area must submit the requested data every 5 s. The MAC protocol for communication between RS nodes is the TunableMAC (one of the standard simulator protocols) (Boulis, 2007). It is worthwhile to mention that our purpose is not to compare the architectures of the RS nodes (DR or SDR). Thus, scenario 1 is compared to scenario 3 and scenario 2 is compared to scenario 4.

Table 5
Parameters of the simulations.

Parameter	Value
Simulation Area	50 m × 50 m
Time of Simulation	10 min
Base Station Location	(25,25)
Number of Nodes	20, 40, 60, 80, 100
Number of Tags	50
Initial Energy	100 J
Query Duration	200 s
Frequency of Sending Data	5 s
Mac Protocol	Tunable MAC

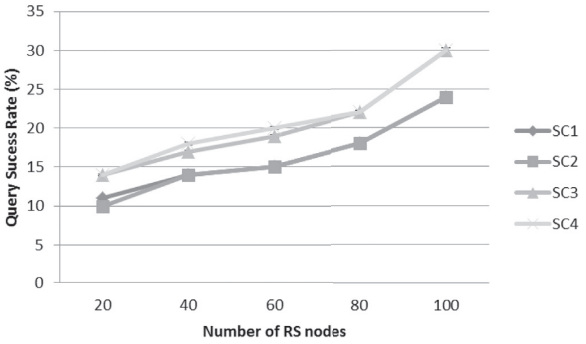


Fig. 12. QSR between RS nodes and tags.

Results are presented according to the evaluation of the different performance metrics as follows.

5.2.2. Query success rate - QSR

The Query Success Rate (QSR) represents the probability that a QUERY command will identify a tag successfully. Thus, Fig. 12 shows the results for the QSR evaluated scenarios. Based on the results, it can be observed that higher values of QSR can be achieved by using the proposed framework. This improvement is explained by the dynamic and automated definition of the *c* parameter that allows the adjustment of the size of the rounds more appropriately, which results in an increased success rate for each query. As a benefit, the IoT application can obtain the requested data in a faster and more reliable way.

5.2.3. Tag identification speed - TIS

The Tag Identification Speed (TIS) is a metric that defines the speed that tags are successfully identified (Wang et al., 2009). The TIS represents the number of tags are identified by a reader in a time interval. Thus, Fig. 13 presents the results obtained in the simulations for the TIS metric. Analyzing the obtained results, it can be inferred that scenarios using the proposed framework have a better tag identification speed than the scenarios that do not use the framework. These results are achieved, as the proposed framework avoids sending redundant identification codes, reducing the time required for the completion of a reading process. Furthermore, the tags identification speed can inter-

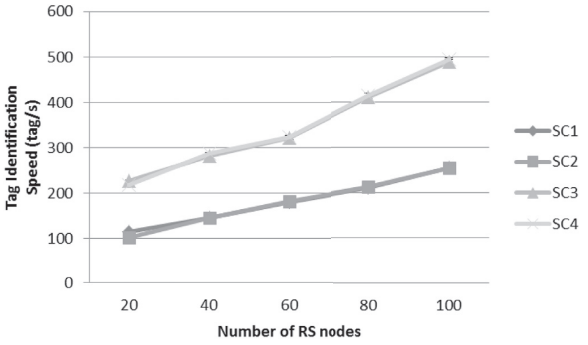


Fig. 13. Tags identification speed.

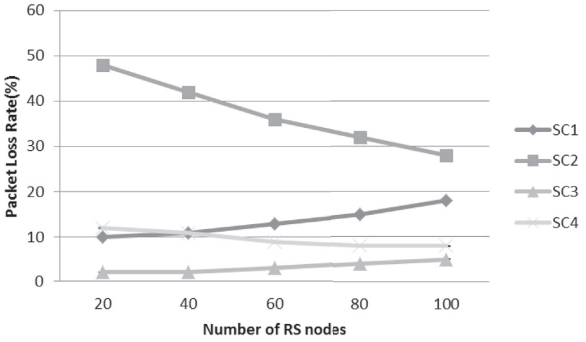


Fig. 14. Packet loss rate between reader and sensor nodes.

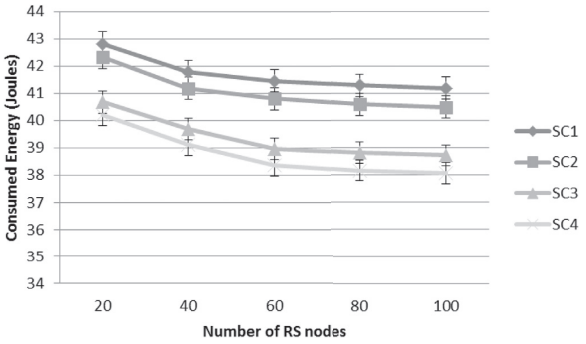


Fig. 15. Average energy consumption of nodes.

fere with the energy consumption and with the packet loss rates. RS-DR nodes with low TIS can increase the identification process time resulting in higher energy consumption, because two communication interfaces are used longer. On the other hand, in RS-SDR nodes, low TIS values can affect the packets routing across the network while identifying the tags. Then, the RS-SDR nodes are unavailable to receive packets from other evaluated nodes. These problems are minimized by using the proposed framework, enabling the IoT applications to meet the goals more efficiently.

5.2.4. Packet loss rate - PLR

The Packet Loss Rate (PLR) represents the number of messages lost in the communication between the RS nodes and the sensor nodes, which can be calculated by eq (3).

$$PLR = \frac{Pr}{\sum_{j=1}^{N} Ps_j}, \tag{3}$$

where Pr is the number of packets received by the sink node, N is the number of nodes on the network, j represents each of the nodes, and Ps_j represents the number of packets sent by each node j.

Fig. 14 presents the results obtained for the packet loss rate. The results show that, for the evaluated scenarios, the proposed framework can reduce the packet loss rate regardless of the RS nodes architecture. For instance, through the proposed framework, the packet loss due to broken routes, with low quality of communication between nodes or with high tag density, can be avoided. Additionally, when considering the current energy levels of each route, the framework is able to avoid routes that have low energy nodes which can cause the route disruption. On the other hand, the framework also considers the number of hops between the destination node and the packet origin. Then, it is possible to measure the distance between two extreme points of the route and select smaller routes. By using the LQI as a parameter to qualify a route, the proposed framework considers the quality of links between nodes that materialize the route. Thus, routes with low quality link can be avoided, which reduces the chances of packet loss. Moreover, as the proposed framework considers the tag density in the reader nodes area, it can prevent packets from being sent through routes whose nodes are overloaded with the tag reading process. The concern with tag density is more important in scenarios where RS nodes use the SDR architecture, as the nodes are unable to forward packets from other nodes when they are in the tag reading process. Thus, avoiding routes with nodes that are in the reading process is extremely important to reduce packet loss rates. Therefore, reducing the packet loss rate of the framework, it can make applications of IoT more reliable since energy consumption may be reduced. Furthermore, the proposed framework is able to provide higher QoS.

5.2.5. Average energy consumption - AEC

The Average Energy Consumption (AEC) metric represents the average amount of energy consumed by each RS node during the experiment time. The AEC is one of the most important metrics for IoT applications due to the fact that, through this, the duration of the application can be estimated (Dietrich and Dressler, 2009; Lin et al., 2015). However, it is important to analyze network efficiency metrics, such as packet loss rate in conjunction with the energy consumption in order to verify if the higher energy consumption benefits the network. The IoT applications can be performed under severe energy restrictions, which makes the energy efficiency a major challenge for the network structure. The AEC is obtained by eq (4).

$$AEC = \frac{\sum_{j=1}^{N} Ec_j}{N}, \tag{4}$$

where N is the number of RS nodes, j represents each of the nodes, and Ec_j represents the energy consumed by each node j.

Fig. 15 illustrates the results for the metrics of nodes average energy consumption. Based on these results, it is observed that scenarios that use the proposed framework present a lower AEC when compared to scenarios that do not use the framework. Regardless the architecture used by the RS nodes, the framework can reduce the average energy consumption by 5%. This is because the time spent on the tag identification process is reduced by using the framework, resulting in lower energy consumption during the identification process. A low packet loss rate provided by the use of the framework may also contribute to reduce energy consumption. By reducing the number of lost packets, the need to resend control and data messages is also reduced, which causes less use of communication interfaces thus providing energy savings. Therefore, the proposed framework can reduce the AEC providing an optimized energy solution for IoT applications.

5.2.6. Network load balance - NLB

The Network Load Balancing (NLB) is the metric that defines the difference between the resource consumption of network nodes. In this

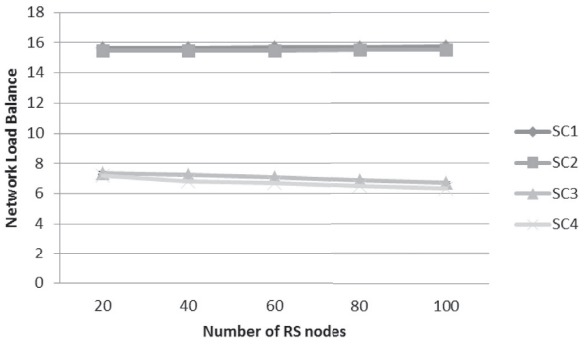


Fig. 16. Network load balancing.

paper, the load balancing is defined by the difference between the highest and the lowest level of energy between network nodes. Thus, the higher NLB value, the lesser network load balancing. Then, the NLB is obtained by eq. (5).

$$NLB = maxEnergy(nodes) - minEnergy(nodes), \quad (5)$$

where *maxEnergy* and *minEnergy* are the functions that get the highest and lowest energy consumption values, respectively, and nodes represent the list of all network nodes with the respective values of consumed energy.

Fig. 16 shows the results obtained for the NLB metric. The proposed framework is able to better allocate network resources. The load balancing is improved by 76%, on average, in scenarios that use RS nodes with DR architecture. On the other hand, the average improvement is 77% in scenarios where RS nodes with SDR architecture were used. Moreover, the proposed framework provides a better distribution of packets on the network, as it selects routes based on current network conditions, which prevents a route from being used until its exhaustion. This feature enables the network resources to be used in a spread and gradual manner. However, the routes should not be selected by observing only the energy levels. Thus, as the proposed framework uses other metrics for packet forwarding, it is able to perform a trade-off between the network load balancing and other important metrics. All in all, a good network load balancing is essential to increase the lifetime of IoT applications, as the packet flow distribution prevents nodes from exhausting its energy source prematurely, which would consequently increase the packet loss rates. Thus, the proposed framework is capable to improve the network load balancing and provide an energy efficient solution for IoT applications.

6. Conclusion

IoT is a paradigm that encompasses various technologies and fields of study. However, in order to achieve satisfactory performances in a wide range of applications, some requirements should be met, such as devices heterogeneity, scalability, data exchange between ubiquitous wireless technologies, optimized energy solutions, traceability and location, self-organizing ability, semantic interoperability and data management, built-in security, and privacy preservation mechanisms. In this sense, it is necessary to integrate existing technologies to provide the resources needed for IoT applications. In this way, the integration of RFID and WSN technologies have a great potential but this integration involves new challenges to be addressed.

This paper aimed to present a framework capable to reduce problems emerged from such integration, aiming to meet some of the aforementioned requirements for IoT applications. For this purpose, the proposed framework considers two components: an RFID anti-collision protocol, responsible for improving the performance of the tag reading system; and a route classifier, responsible for assisting the routing protocols to define the best route to forward a packet.

The presented results demonstrated the capacity of this framework to increase the network reliability for IoT applications, regarding the query success rates. Additionally, the framework increases the tag identification speed by avoiding redundant code information from being sent. The proposed solution was able to improve the query success rate in 25% on average whereas the tag identification speed was increased up to 115% when compared with the standard approaches. Finally, when considering information provided by different technologies at the network, the framework was able to assist the routing protocol in selecting the best route for forwarding a packet. As a result, the packet loss rate is reduced and this indirectly contributes to the reduction of the nodes energy consumption and better network load balancing. The solution provided a reduction in the data packet loss rate of 75% on average, and improved the network load balance in 55% on average, considering the studied scenarios.

In general, it was observed that some of the resources required by IoT applications could be met, such as data exchange between ubiquitous technologies, optimized energy solution, devices heterogeneity, tracking, and location capabilities.

Acknowledgments

This work has been supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Brazil, through the grants 201155/2015-0 and 309335/2017-5, by National Funding from the FCT – Fundação para a Ciência e a Tecnologia through the UID/EEA/500008/2013 Project, by Government of Russian Federation, Grant 074-U01, and by Finep, with resources from FUNTTEL, Grant No. 01.14.0231.00, under the *Centro de Referência em Radiocomunicações* - CRR project of the *Instituto Nacional de Telecomunicações* (Inatel), Brazil.

References

Abahsain, A., Al-Fagih, A.E., Oteafy, S.M., Hassanein, H.S., 2013. Selective context fusion utilizing an integrated rfid-wsn architecture. In: 2013 IEEE Consumer Communications and Networking Conference (CCNC). IEEE, pp. 317–322.
Abdul-Salaam, G., Abdullah, A.H., Anisi, M.H., 2017. Energy-efficient data reporting for navigation in position-free hybrid wireless sensor networks. IEEE Sens. J. 17 (7), 2289–2297.
Agrawal, S., Das, M.L., 2011. Internet of things - a paradigm shift of future internet applications. In: 2011 Nirma University International Conference on Engineering (NUiCONE). IEEE, pp. 1–7.
Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002. Wireless sensor networks: a survey. Comput. Netw. 38 (4), 393–422.
Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., Ayyash, M., 2015. Internet of things: a survey on enabling technologies, protocols, and applications. IEEE Commun. Surv. Tutor. 17 (4), 2347–2376.
Al-Turjman, F.M., Al-Fagih, A.E., Hassanein, H.S., 2012. A novel cost-effective architecture and deployment strategy for integrated rfid and wsn systems. In: 2012 International Conference on Computing, Networking and Communications (ICNC). IEEE, pp. 835–839.
Al-Turjman, F.M., Al-Fagih, A.E., Alsalih, W.M., Hassanein, H.S., 2013. A delay-tolerant framework for integrated rsns in iot. Comput. Commun. 36 (9), 998–1010.
Alfian, G., Rhee, J., Ahn, H., Lee, J., Farooq, U., Ijaz, M.F., Syaekhoni, M.A., 2017. Integration of rfid, wireless sensor networks, and data mining in an e-pedigree food traceability system. J. Food Eng. 212 (Suppl. C), 65–75.
AlShawi, I.S., Yan, L., Pan, W., Luo, B., 2012. Lifetime enhancement in wireless sensor networks using fuzzy approach and a-star algorithm. IEEE Sens. J. 12 (10), 3010–3018.
Andreou, P.G., Zeinalipour-Yazti, D., Samaras, G.S., Chrysanthis, P.K., 2014. A network-aware framework for energy-efficient data acquisition in wireless sensor networks. J. Netw. Comput. Appl. 46, 227–240.
Atzori, L., Iera, A., Morabito, G., 2010. The internet of things: a survey. Comput. Netw. 54 (15), 2787–2805.
Baloch, F., Pendse, R., 2013. Comparison of transmission control protocols based on epc c1g2 standard. Int. J. Comput. Netw. Technol. 1 (1), 83–94.
Bashir, U., Jha, K.R., Mishra, G., Singh, G., Sharma, S.K., 2017. Octahedron-shaped linearly polarized antenna for multistandard services including rfid and iot. IEEE Trans. Antenn. Propag. 65 (7), 3364–3373.
Boulis, A., 2007. Castalia: revealing pitfalls in designing distributed algorithms in wsn. In: Proceedings of the 5th International Conference on Embedded Networked Sensor Systems. ACM, pp. 407–408.
Bueno-Delgado, M.V., Vales-Alonso, J., 2011. On the optimal frame-length configuration on real passive rfid systems. J. Netw. Comput. Appl. 34 (3), 864–876.

Chapter 3. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

Camp, T., Boleng, J., Davies, V., 2002. A survey of mobility models for ad hoc network research. Wireless Commun. Mobile Comput. 2 (5), 483–502.

Chen, M., Kwon, T., Choi, Y., 2006. Energy-efficient differentiated directed diffusion (eddd) in wireless sensor networks. Comput. Commun. 29 (2), 231–245.

Clausen, T., Yi, J., Niktash, A., Igarashi, Y., Satoh, H., Herberg, U., Lavenu, C., Lys, T., Dean, J., 2016. The lightweight on-demand ad hoc distance-vector routing protocol-next generation (loadng), Internet-Draft draft-clausen-lln-loadng-14.txt. IETF Secretariat.

DaCosta, F., 2013. Rethinking the Internet of Things: a Scalable Approach to Connecting Everything. Apress.

Diallo, C., Marot, M., Becker, M., 2011. Efficiency benefits through load-balancing with link reliability based routing in wsns. Int. J. Adv. Netw. Serv. 3 (3 and 4), 430–446.

Díaz, M., Martín, C., Rubio, B., 2016. State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. J. Netw. Comput. Appl. 67, 99–117.

Dietrich, I., Dressler, F., 2009. On the lifetime of wireless sensor networks. ACM Trans. Sens. Netw. TOSN 5 (1) 5:1–5:39.

Dillinger, M., Madani, K., Alonistioti, N., 2005. Software Defined Radio: Architectures, Systems and Functions. John Wiley & Sons.

Dorigo, M., Maniezzo, V., Colorni, A., 1996. Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybernet. Part B Cybernet. 26 (1), 29–41.

Engelbrecht, A.P., 2007. Computational Intelligence: an Introduction. John Wiley & Sons.

EPCglobal, 2008. Epctm Radio-frequency Identity Protocols Class-1 Generation-2 uhf Rfid Protocol for Communications at 860 mhz–960 mhz. version 1.2.0.

Fernández-Caramés, T.M., Fraga-Lamas, P., Suárez-Albela, M., Castedo, L., 2016. Reverse engineering and security evaluation of commercial tags for rfid-based iot applications. Sensors 17 (1), 28.

Gandotra, P., Jha, R.K., Jain, S., 2017. A survey on device-to-device (d2d) communication: architecture and security issues. J. Netw. Comput. Appl. 78 (Suppl. C), 9–29.

García-Hernández, C.F., Ibarguengoytia-Gonzalez, P.H., García-Hernández, J., Pérez-Díaz, J.A., 2007. Wireless sensor networks and applications: a survey. IJCSNS Int. J. Comput. Sci. Netw. Secur. 7 (3), 264–273.

Gluhak, A., Krco, S., Nati, M., Pfisterer, D., Mitton, N., Razafindralambo, T., 2011. A survey on facilities for experimental internet of things research. IEEE Commun. Mag. 49 (11), 58–67.

Guo, W., Zhang, W., 2014. A survey on intelligent routing protocols in wireless sensor networks. J. Netw. Comput. Appl. 38, 185–201.

Hu, F., Cao, X., 2010. Wireless Sensor Networks: Principles and Practice. CRC Press.

Hussain, S., Schaffner, S., Moseychuck, D., 2009. Applications of wireless sensor networks and rfid in a smart home environment. In: 2009 Seventh Annual Communication Networks and Services Research Conference (CNSR). IEEE, pp. 153–157.

Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., Silva, F., 2003. Directed diffusion for wireless sensor networking. IEEE ACM Trans. Netw. 11 (1), 2–16.

Jurdak, R., Ruzzelli, A.G., O'Hare, G.M., 2008. Multi-hop rfid wake-up radio: design, evaluation and energy tradeoffs. In: Proceedings of 17th International Conference on Computer Communications and Networks (ICCCN). IEEE, pp. 1–8.

Kabir, M.A., Han, J., Hong, B., 2015. Reader level filtering for efficient query processing in rfid middleware. J. Netw. Comput. Appl. 48, 58–70.

Kim, J.G., Shin, W.J., Yoo, J.H., 2007. Performance analysis of epc class-1 generation-2 rfid anti-collision protocol. In: Computational Science and its Applications–ICCSA 2007. Springer, pp. 1017–1026.

Klair, D.K., Chin, K.-W., Raad, R., 2010. A survey and tutorial of rfid anti-collision protocols. Commun. Surv. Tutor. IEEE 12 (3), 400–421.

Kulkarni, R.V., Forster, A., Venayagamoorthy, G.K., 2011. Computational intelligence in wireless sensor networks: a survey. IEEE Commun. Surv. Tutor. 13 (1), 68–96.

Kumar, G., Rai, M.K., 2017. An energy efficient and optimized load balanced localization method using cds with one-hop neighbourhood and genetic algorithm in wsns. J. Netw. Comput. Appl. 78, 73–82.

Lee, D., Kim, K., Lee, W., 2007. Q+-algorithm: an enhanced rfid tag collision arbitration algorithm. In: International Conference on Ubiquitous Intelligence and Computing. Springer, pp. 23–32.

Liao, W.-H., Kao, Y., Fan, C.-M., 2008. Data aggregation in wireless sensor networks using ant colony algorithm. J. Netw. Comput. Appl. 31 (4), 387–401.

Lin, Y., Zhang, J., Chung, H.-H., Ip, W.H., Li, Y., Shi, Y.-H., 2012. An ant colony optimization approach for maximizing the lifetime of heterogeneous wireless sensor networks. IEEE Trans. Syst. Man Cybernet. Part C Appl. Rev. 42 (3), 408–420.

Lin, Y.H., Chou, Z.T., Yu, C.W., Jan, R.H., 2015. Optimal and maximized configurable power saving protocols for corona-based wireless sensor networks. IEEE Trans. Mobile Comput. 14 (12), 2544–2559.

Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., Zhao, W., 2017. A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J. 4 (5), 1125–1142.

Liu, X., He, D., 2014. Ant colony optimization with greedy migration mechanism for node deployment in wireless sensor networks. J. Netw. Comput. Appl. 39, 310–318.

Liu, H., Bolic, M., Nayak, A., Stojmenovic, I., 2008. Taxonomy and challenges of the integration of rfid and wireless sensor networks. IEEE Netw. 22 (6), 26–35.

López, T.S., Kim, D., Canepa, G.H., Koumadi, K., 2009. Integrating wireless sensors and rfid tags into energy-efficient and dynamic context networks. Comput. J. 52 (2), 240–267.

Luo, Z., Geng, L., Wang, P., Jiang, J., 2011. An information-upload approach with low power consumption for a rfid-wsn smart node system. In: 2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM). IEEE, pp. 1–4.

Machado, K., Rosário, D., Cerqueira, E., Loureiro, A.A., Neto, A., de Souza, J.N., 2013. A routing protocol based on energy and link quality for internet of things applications. Sensors 13 (2), 1942–1964.

Mainetti, L., Marasovic, I., Patrono, L., Solic, P., Stefanizzi, M.L., Vergallo, R., 2016. A novel iot-aware smart parking system based on the integration of rfid and wsn technologies. Int. J. RF Technol. 7 (4), 175–199.

Mann, P.S., Singh, S., 2017. Energy efficient clustering protocol based on improved metaheuristic in wireless sensor networks. J. Netw. Comput. Appl. 83, 40–52.

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

Namboodiri, V., DeSilva, M., Deegala, K., Ramamoorthy, S., 2012. An extensive study of slotted aloha-based rfid anti-collision protocols. Comput. Commun. 35 (16), 1955–1966.

Pedrycz, W., Gomide, F., 2007. Fuzzy Systems Engineering: toward Human-centric Computing. John Wiley & Sons.

Peng, S., Low, C., 2015. Energy neutral directed diffusion for energy harvesting wireless sensor networks. Comput. Commun. 63, 40–52.

Perera, C., Zaslavsky, A., Christen, P., Georgakopoulos, D., 2014. Context aware computing for the internet of things: a survey. IEEE Commun. Surv. Tutor. 16 (1), 414–454.

Radi, M., Dezfouli, B., Bakar, K.A., Lee, M., 2012. Multipath routing in wireless sensor networks: survey and research challenges. Sensors 12 (1), 650–685.

Rajesh, S., 2013. Integration of active rfid and wsn for real time low-cost data monitoring of patients in hospitals. In: 2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE), pp. 1–6.

Ramos, H.S., Frery, A.C., Boukerche, A., Oliveira, E.M., Loureiro, A.A., 2014. Topology-related metrics and applications for the design and operation of wireless sensor networks. ACM Trans. Sens. Netw. TOSN 10 (3), 53.

Rashid, B., Rehmani, M.H., 2016. Applications of wireless sensor networks for urban areas: a survey. J. Netw. Comput. Appl. 60, 192–219.

Roberts, C.M., 2006. Radio frequency identification (rfid). Comput. Secur. 25 (1), 18–26.

Schindelhauer, C., 2006. Mobility in wireless networks. In: SOFSEM 2006: Theory and Practice of Computer Science. Springer, pp. 100–116.

Sheng, Z., Yang, S., Yu, Y., Vasilakos, A.V., McCann, J.A., Leung, K.K., 2013. A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities. IEEE Wirel. Commun. 20 (6), 91–98.

Stankovic, J.A., 2014. Research directions for the internet of things. IEEE Internet Things J. 1 (1), 3–9.

Sun, C.-T., 1994. Rule-base structure identification in an adaptive-network-based fuzzy inference system. IEEE Trans. Fuzzy Syst. 2 (1), 64–73.

T. Instruments, Cc1000: Single Chip Very Low Power Rf Transceiver, Reference SWRS048. Rev A.

T. Instruments, Cc2420: 2.4 ghz ieee 802.15. 4/zigbee-ready rf transceiver, Available at: http://www.ti.com/lit/gpn/cc2420 (visited on 06/25/2016).

Teng, J., Xuan, X., Bai, Y., 2010. A fast q algorithm based on epc generation-2 rfid protocol. In: 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM). IEEE, pp. 1–4.

Tunca, C., Isik, S., Donmez, M.Y., Ersoy, C., 2015. Ring routing: an energy-efficient routing protocol for wireless sensor networks with a mobile sink. IEEE Trans. Mobile Comput. 14 (9), 1947–1960.

Tuttlebee, W.H., 2003. Software Defined Radio: Enabling Technologies. John Wiley & Sons.

Wang, C., Daneshmand, M., Sohraby, K., Li, B., 2009. Performance analysis of rfid generation-2 protocol. IEEE Trans. Wireless Commun. 8 (5), 2592–2601.

Welbourne, E., Battle, L., Cole, G., Gould, K., Rector, K., Raymer, S., Balazinska, M., Borriello, G., 2009. Building the internet of things using rfid: the rfid ecosystem experience. IEEE Internet Comput. 13 (3), 48–55.

Winter, T., Brandt, A., Hui, J., Kelsy, R., Levis, P., Pister, K., Struik, R., Vasseur, J., Alexander, R., 2012. In: RFC (Ed.), Rpl: Ipv6 Routing Protocol for Low-power and Lossy Networks, RFC 6550.

Yang, G., Xiao, M., Chen, C., 2007. A simple energy-balancing method in rfid sensor networks. In: Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop. IEEE, pp. 306–310.

Yick, J., Mukherjee, B., Ghosal, D., 2008. Wireless sensor network survey. Comput. Netw. 52 (12), 2292–2330.

Zhang, L., Wang, Z., 2006. Integration of rfid into wireless sensor networks: architectures, opportunities and challenging problems. In: 2006 Fifth International Conference on Grid and Cooperative Computing Workshops (GCCW). IEEE, pp. 463–469.

Zhao, H., Ning, X.J., Yang, M.F., Chai, H.F., 2014. A link evaluation method employing statistical means of received signal strength indicator and link quality indicator for wireless sensor networks. In: Mechanical Engineering, Materials Science and Civil Engineering II. Applied Mechanics and Materials, vol. 470. Trans Tech Publications, pp. 722–728.

Zhu, L., Yum, T.-S., 2011. A critical survey and analysis of rfid anti-collision mechanisms. IEEE Commun. Mag. 49 (5), 214–221.

Zhu, Q., Wang, R., Chen, Q., Liu, Y., Qin, W., 2010. Iot gateway: bridging wireless sensor networks into internet of things. In: 2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC). IEEE, pp. 347–352.

Zungeru, A.M., Ang, L.-M., Seng, K.P., 2012. Classical and swarm intelligence based routing protocols for wireless sensor networks: a survey and comparison. J. Netw. Comput. Appl. 35 (5), 1508–1536.

Chapter 3. A framework for enhancing the performance of Internet of Things applications based on RFID and WSNs

José V. V. Sobral (jose.sobral@it.ubi.pt) is currently a Ph.D. student at the University of Beira Interior (UBI), Covilhã, Portugal and Instituto de Telecomunicações, Portugal. He received his M.Sc. degree in Computer Science from the Federal University of Piauí (UFPI), Teresina, Brazil, and B.S. degree in Computer Science from the Centro de Ensino Unificado de Teresina (CEUT), Teresina, Brazil. Jose Sobral is an assistant professor at the Federal Institute of Maranhão (IFMA), São Luís, Brazil, and a member of NetGNA Research Group. His research interests include Internet of Things (IoT), routing protocols for low power and lossy networks, wireless sensors networks, RFID systems, and computational intelligence.

Joel J.P.C. Rodrigues [S′01, M′06, SM′06] is a professor and senior researcher at the National Institute of Telecommunications (Inatel), Brazil and senior researcher at the *Instituto de Telecomunicações*, Portugal. He has been professor at the University of Beira Interior (UBI), Portugal and visiting professor at the University of Fortaleza (UNIFOR), Brazil. He received the *Academic Title of Aggregated Professor* in informatics engineering from UBI, the Habilitation in computer science and engineering from the University of Haute Alsace, France, a PhD degree in informatics engineering and an MSc degree from the UBI, and a five-year BSc degree (licentiate) in informatics engineering from the University of Coimbra, Portugal. His main research interests include e-health, sensor networks and IoT, vehicular communications, and mobile and ubiquitous computing. Prof. Rodrigues is the leader of the Internet of Things research group (CNPq), Member of the IEEE ComSoc Board of Governors as Director for Conference Development, IEEE Distinguished Lecturer, the President of the scientific council at ParkUrbis – Covilhã Science and Technology Park, the Past-Chair of the IEEE ComSoc Technical Committee on eHealth, the Past-chair of the IEEE ComSoc Technical Committee on Communications Software, Steering Committee member of the IEEE Life Sciences Technical Community and Publications co-Chair, and Member Representative of the IEEE Communications Society on the IEEE Biometrics Council. He is the editor-in-chief of the International Journal on E-Health and Medical Communications, the editor-in-chief of the Recent Advances on Communications and Networking Technology, the editor-in-chief of the Journal of Multimedia Information Systems, and editorial board member of several high-reputed journals. He has been general chair and TPC Chair of many international conferences, including IEEE ICC, IEEE GLOBECOM, and IEEE HEALTHCOM. He is a member of many international TPCs and participated in several international conferences organization. He has authored or coauthored over 550 papers in refereed international journals and conferences, 3 books, and 2 patents. He had been awarded several Outstanding Leadership and Outstanding Service Awards by IEEE Communications Society and several best papers awards. Prof. Rodrigues is a licensed professional engineer (as senior member), member of the Internet Society, and a senior member ACM and IEEE.

Ricardo de Andrade Lira Rabêlo is a professor in the Intelligent Systems and Optimization (Linear Programming and Nonlinear Programming) area in the Computer Science Department at Federal University of Piaui (UFPI) since 2013. He leads research activities and projects, teaches undergraduate and post-graduate courses, and is an advisor for M.Sc. and B.Sc. students. Ricardo Rabêlo is the leader of OASIS (Optimization, Autonomous Solutions, and Intelligent Systems) Laboratory at UFPI. He holds a Ph.D. degree in Computer Science from the University of São Paulo (USP). He has publications in international conferences and journals. He is a member of the IEEE and the Brazilian Computer Society (SBC). He is interested in research involving topics of Intelligent Systems, Internet of Things, Optimization, Electric Power Systems and Search-Based Software Engineering.

José Carlos Lima Filho received the Bachelor degree in Information System from the Faculty of Business Activities of Teresina (FAETE), Piaui, Brazil, in 2013. He is Specialist in Technologies for WEB Applications from the Universidade Norte do Paraná (UNOPAR), Brazil, since 2014. In 2015 he served as Technical Teaching Instructor in Database at the Fundação Bradesco, Teresina unit. He works as System Analyst at the Status Soluções Tecnológicas and is an M.S. student at the Federal University of Piaui. His current research interest is Internet of Things. He is a coauthor of more than 15 papers in peer reviewed journals and conferences over the last ten years.

Natanael Sousa received B.S. degree in Computer Science from the Federal University of Piauí (UFPI) in 2017. He has experience in Computer Science, with emphasis on Computer Networks and Computational Intelligence. He works mainly on the following topics: Wireless Sensor Networks, RFID Systems, Internet of Things, Wireless Network Routing, Fuzzy Systems, Artificial Neural Networks.

Harilton S. Araújo is graduated in Data Processing and Specialist in Information Technology from the Federal University of Ceará (UFC). He received its M.S. degree in Applied Computing from the University of Fortaleza (UNIFOR). He is a Ph.D. student at the University of Fortaleza (UNIFOR). He is a member the Internet of Things Research Group at the Faculty Estácio of Teresina and the Wireless Sensor Networks Laboratory (LARES) at the Technological Sciences Center of UNIFOR. He is professor and coordinator of Computer Science course at Faculty Estácio of Teresina. He has publications in international conferences and is an official of the Brazilian Army - R2 (Informatics) acting in the consulting area. He has experience in computer science, with knowledge in communication networks, wireless sensor networks, Internet of Things (IoT), telematics and information security.

Raimir Holanda Filho is Ph.D. in Computer Science from the Universitat Politecnica de Catalunya (2005). Currently, he is a full professor at the University of Fortaleza (UNIFOR). He has experience in Computer Science, with a focus in Teleinformatics, acting mainly in the following topics: wireless sensor networks, ubiquitous computing, network security, Internet of things and smart cities.

Chapter 4

LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks

This chapter consists in the following paper:

LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks

José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Kashif Saleem, and Vasco Furtado

Sensors, MDPI, ISSN: 1424-8220, 2019.

DOI: doi.org/10.3390/s19010150

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 3.275

ISI Article Influence Score (2019): 0.530

Journal Ranking (2019): Q1 - 15/64 (Instruments & Instrumentation)

Journal Ranking (2019): Q2 - 77/266 (Engineering, Electrical & Electronic)

sensors

MDPI

Article

LOADng-IoT: An Enhanced Routing Protocol for Internet of Things Applications over Low Power Networks

José V. V. Sobral [1,2], Joel J. P. C. Rodrigues [1,3,4,*], Ricardo A. L. Rabêlo [5] and Kashif Saleem [4] and Vasco Furtado [6,†]

1 Instituto de Telecomunicações, Universidade da Beira Interior, 6201-001 Covilhã, Portugal; jose.sobral@it.ubi.pt
2 Department of Education, Federal Institute of Maranhão (IFMA), R. Afonso Pena, 174, São Luís-MA 65010-030, Brazil
3 National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí-MG 37540-000, Brazil
4 Center of Excellence in Information Assurance (CoEIA), King Saud University, Riyadh 11653, Saudi Arabia; ksaleem@ksu.edu.sa
5 Department of Computing, Federal University of Piauí (UFPI), Teresina-PI 64049-550, Brazil; ricardoalr@ufpi.edu.br
6 Programa de Pos-Graduação em Informática Aplicada (PPGIA), University of Fortaleza (UNIFOR), Av. Washington Soares, 1321, Fortaleza-CE 60811-905, Brazil; vasco@unifor.br
* Correspondence: joeljr@ieee.org; Tel.: +55-35-3471-9200
† Current address: Avenue João de Camargo, 510-Centro, Santa Rita do Sapucaí 37540-000, Brazil.

Received: 12 December 2018; Accepted: 31 December 2018; Published: 3 January 2019

Abstract: The Internet of Things (IoT) is an emerging paradigm that proposes the connection of objects to exchange information in order to reach a common objective. In IoT networks, it is expected that the nodes will exchange data between each other and with external Internet services. However, due to deployment costs, not all the network devices are able to communicate with the Internet directly. Thus, other network nodes should use Internet-connected nodes as a gateway to forward messages to Internet services. Considering the fact that main routing protocols for low-power networks are not able to reach suitable performance in the displayed IoT environment, this work presents an enhancement to the Lightweight On-demand Ad hoc Distance-vector Routing Protocol—Next Generation (LOADng) for IoT scenarios. The proposal, named LOADng-IoT, is based on three improvements that will allow the nodes to find Internet-connected nodes autonomously and dynamically, decreasing the control message overhead required for the route construction, and reducing the loss of data messages directed to the Internet. Based on the performed assessment study, which considered several number of nodes in dense, sparse, and mobility scenarios, the proposed approach is able to present significant results in metrics related to quality-of-service, reliability, and energy efficiency.

Keywords: internet of things; LOADng; LOADng-IoT; low power and lossy networks; routing protocol

1. Introduction

The Internet of Things (IoT) is a wide concept that has attracted attention from the research community in recent years [1]. The term IoT can be used to describe a pervasive and ubiquitous network in which devices exchange information between each other without the requirement for human intervention [2]. This network can be used by applications of a wide variety and with varying objectives, such as smart homes and cities [3], industrial automation [4], smart markets [5], and healthcare systems [6,7]. However, at the same time, the IoT has captured the attention of

the business world and society. IoT concepts give rise to several technical challenges that limit its broad adoption.

IoT devices form, in general, a low power and lossy network (LLN), composed of many nodes with strong restrictions on memory, processing capacity, and, in some cases, energy. Depending on the application, the nodes in an IoT network can have different hardware capacities and application objectives. In the IoT scenario presented in Figure 1, some network nodes can have a direct Internet connection to send and receive messages from the Internet. In contrast, other nodes from the same network, due to hardware limitations, cannot have a direct Internet connection and require the use of the Internet-connected nodes to access external services. All of these nodes can also exchange information in a local context without the necessity of transmitting data to the Internet. The task of discovering the routes and allowing data messages to be exchanged among nodes is performed by the routing protocol. Thus, in an LLN, the network performance is strongly related to how the routing protocols use the limited hardware resources of the network devices.

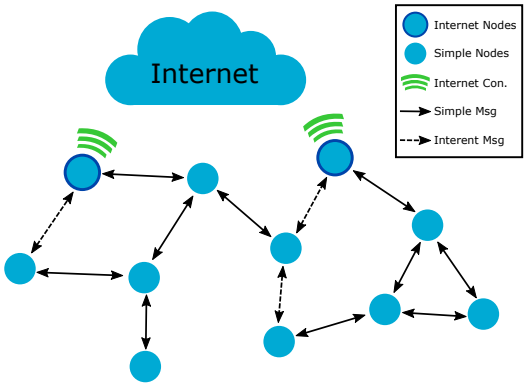


Figure 1. Illustration of an IoT network model with different devices and data message types.

In the context of low-power IoT networks, the routing protocols can be grouped into two types based on the route creation principles: proactive and reactive [8]. Proactive protocols begin the creation of routes among the nodes without the prior necessity of data message transmission. The route creation process is, in general, triggered by gateway nodes that have the function of collecting information from other network devices (multipoint-to-point (MP2P) traffic). For this reason, this type of protocol is widely adopted by periodical data collection applications. The main example of a proactive routing protocol for LLNs is the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [9]. In contrast, reactive protocols create routes only when a node intends to send a message to a destination. Thus, the route discovery process is triggered by the node that wants the message. The created routes are maintained in the routing table of the nodes, remaining while in use and removed afterwards. Hence, this type of protocol is indicated for non-periodical traffic application, where any node can send messages to any other (point-to-point (P2P) traffic). The current and most relevant reactive protocol for LLN is LOADng [10]. This work is focused on IoT applications where the traffic pattern is variable, the network devices have different capacities, and the communication among the devices is, mainly, P2P (as exemplified in Figure 1). Thus, considering that reactive protocols are the most appropriate for this type of scenario, LOADng will be used as a basis to the design the proposed solution. The protocol is currently under consideration by the Internet Engineering Task Force (IETF) to become a defined standard. LOADng is in its fifteenth draft and has undergone several modifications since its initial proposal in October 2011 [11]. Thus, the most recent version will be used in this work. A short but comprehensive description of LOADng is presented in Section 3.

Although it uses the most indicated route creation principles in the IoT scenario considered in this study, LOADng can present several problems such as the necessity of previous and static definitions of the nodes responsible for providing the Internet connection to other network devices.

Also, the obligation of several routing creation processes to construct on-demand routes to P2P traffic can provoke a high control message overhead. Thus, the main objective of this work is to create an enhancement for LOADng to allow the protocol to better discover and maintain routes for traffic directed to the Internet in IoT networks formed by devices with different capacities. The proposed approach, LOADng-IoT, is composed of three improvements that are able to boost the process of route discovery, reduce the overhead of control messages, and improve the network's quality-of-service (QoS). In summary, the proposal allows the nodes to find Internet-connected nodes without the prior definition of a gateway. This behavior allows nodes without an Internet connection to forward their data packets to external Internet services with much greater reliability and lower latency. Also, the proposed solution presents a cache system for routes to Internet nodes to reduce the control message overhead required in the process of route discovery. Finally, the solution introduces a new error code that should be used to avoid the loss of data packets to the Internet. Thus, the main contributions of the proposal presented in this work (LOADng-IoT) are as follows:

- LOADng-IoT improves network QoS and reliability by increasing the packet delivery ratio and the reduction of end-to-end latency for the different message types exchanged by nodes in both dense and sparse IoT scenarios.
- It reduces the number of control messages required to construct routes among nodes, contributing to a more efficient network with lower overhead.
- LOADng-IoT reduces the amount of energy required to both build paths and route data messages, making the network more power efficient.
- It dispenses with the use of predefined Internet gateways since the Internet-connected nodes are sought on demand and can change according to their connection availability. This feature also removes the existence of a single point of failure (SPOF) for the connection of IoT devices with external Internet services.
- It presents a flexible solution, whereby parts of the proposal can be adopted according to the hardware capacities of the nodes.

The remainder of this document is organized as follows. Section 2 presents the most important related works on the topic and Section 3 presents an overview of LOADng to provide a clear understanding of the proposed solution. Section 4 gives a detailed description of the proposed LOADng-IoT, while Section 5 describes the used scenario for the performance evaluation, the obtained results, and corresponding discussion concerning the performance of the studied protocols. Finally, Section 6 concludes the paper and offers suggestions for future work.

2. Related Work

This work proposes a new enhancement to aid the LOADng route discovery process in IoT scenarios composed of nodes with different capacities and variable message traffic. In the current literature, several studies focus on performance and propose improvements for LOADng. However, to the best of the authors' knowledge, the current related literature does not propose improvements to the route discovery process of Internet nodes for LOADng in IoT scenarios.

In [12], the authors compared LOADng with RPL in scenarios with different traffic patterns. According to the results obtained, LOADng surpassed RPL in point-to-multipoint (P2MP) and P2P traffic scenarios. However, in the MP2P scenarios, the RPL presented better results. As expected, the proactive features of RPL make it the most appropriate for data collection application scenarios. In contrast, LOADng was able to work better with more generalized traffic but presented a higher delay due to the necessity of realizing the process of route discovery on demand. The performance of LOADng in IoT applications with P2P and MP2P traffic pattern is also studied in [13].

Based on the limited performance of LOADng in MP2P scenarios, Yi and Clausen [14] proposed the LOADng Collection Tree Extension (LOADng-CTP). The proposed enhancement allows the construction of collection trees on-demand to better attend the traffic that flows from the leaf nodes to

the root. The proposal was compared with RPL through computational simulations. According to the obtained results, LOADng-CTP obtained delivery ratios, delay, and number of collisions that were very similar to RPL. However, the proposed approach required less control message overhead in all studied scenarios.

A new composite routing metric for LOADng is proposed in [15]. The termed LRRE metric presents an additive approach that merges residual energy (RE), hop count, and number of live routes (LR) in a node. According to the obtained results, the proposed LRRE was able to demonstrate better results when compared to each merged metric used individually. The authors also studied the behavior of the proposed metric in LOADng with a multipath routing adaption. Again, the proposal was able to deliver better results in terms of energy consumption, packet delivery ratio, and load balancing. The impact of the route selection mechanism in the performance of LOADng is widely studied in [16,17].

A multipath improvement for LOADng is also proposed in [18]. The Neighbour-Disjoint Multipath for LOADng (LOADng+NDM) presents a new multipath mechanism adapted for LOADng aiming to improve the network reliability and QoS. The proposed approach creates an initial shortest primary path between a sender and a receiver node. In sequence, it tries to construct a set of backup routes avoiding the nodes that compose the primary path. This behavior seeks to create disjoint routes where backup routes cannot be affected by the failures in the primary path. Simulation and testbed results have exposed the proposed LOADng+NDM overcomes the default LOADng and RPL in simple P2P scenarios. However, in a scenario with multiple P2P messages exchange, the proposed approach have presented higher control overhead and end-to-end latency.

Araújo et al. [19] propose an enhancement routing metric for RPL in IoT environments composed of heterogeneous devices. In the studied scenario, the network was composed of sensor devices, radio frequency tags, and reader-sensor devices. The proposal, based on fuzzy systems, dynamically adapts the routing metrics during the network functioning according to the application requirements. The experimental simulation results showed that the proposed solution was able to overcome the default RPL metrics in terms of energy consumption and packet delivery ratio.

A routing protocol for IoT networks based on the composition of routing metrics is presented in [20]. The Routing by Energy and Link quality (REL) proposes an adaption of Ad-hoc On-demand Distance Vector (AODV) aiming to increase the network reliability and power efficiency. The proposed protocol uses a *weakLinks* mechanism to identify links with low quality across a path. In the route creation process, paths with a high number of *weakLinks* are avoided. The choice of the best route also considers the residual energy of the nodes and hop count. In all the studied scenarios, REL was able to outperform the AODV regarding latency, packet delivery ratio, and network lifetime.

Araújo et al. [21] present a solution based on smartphones to allow the interoperability of IoT devices. The authors propose an architecture in which a smartphone aggregates several communications interfaces to work as a gateway for different technologies. The proposal is software-based and uses only the default implementation of the considered standards. Thus, no improvements were made at the routing layer. A specific testbed was deployed to evaluate the solution in terms of memory, CPU, and energy consumption.

In [22,23], the authors studied solutions for gateway discovery in mobile ad hoc networks (MANETs). The proposals are based on the usage of periodical control messages and consider networks that have mobile nodes or are equipped with an IEEE 802.11 communication interface, which is not the most appropriate for IoT low-power devices. The performance of the solutions was evaluated in terms of packet delivery ratio and end-to-end delay.

Considering the limitations of the current literature and the requirements of IoT low-power networks, this work proposes a new mechanism for LOADng that will allow it to search Internet-connected nodes in a dynamic and on-demand manner. Moreover, the proposed approach can improve normal data traffic among the nodes, enabling the network to become more energy efficient and reliable.

3. LOADng Protocol Overview

The LOADng routing protocol proposes a simplification of the AODV [24], a well-known reactive routing protocol based on route discovery using messages of route request and route reply. In LOADng, some aspects are simplified with the aim of reducing the protocol complexity and the amount of computational resources required to execute it. Among the simplifications, it is possible to detach the restriction to the sending of intermediate route reply messages and the avoidance of the use of periodical control messages [16]. Thus, LOADng is specifically designed for networks composed of devices with strong hardware restrictions. Also, it allows the use of different addressing schemes such as IPv6, IPv4, and Rime [25].

The following subsections present an overview of the LOADng protocol. These explanations are necessary to understand the approach proposed in this work.

3.1. LOADng Functioning in Brief

LOADng is a reactive routing protocol based on route discovery using route request and route reply messages. Thus, when a node wants to send a data message and the route to the destination is unknown, it should begin a new route discovery process. To this end, the node broadcasts a route request (RREQ) message to search for a route to the desired destination. Each node that receives an RREQ should perform message processing and consider the message to be forwarded. This process continues until the RREQ reaches the sought destination. The destination should then generate a route reply (RREP) message to answer the received RREQ. The RREP is forwarded in unicast to the RREQ originator, constructing a route between the two nodes interested in the message exchange. Finally, the RREP is received by the RREQ originator, which should begin to send data messages using the path created by the route discovery process.

3.2. LOAng Control Messages and Information Base

The process of route discovery is performed with the use of control messages inspired by AODV. RREQ messages are always used to request the creation of a route to a destination when a node needs to send a data message and the path to the destination is unknown. RREP messages are used by the destination that receives the RREQ as an answer to the request for route creation. An RREP message may, optionally, require an acknowledgment. In this case, the route reply acknowledgment (RREP_ACK) message is used to answer a received RREP. When a node fails at the moment of data message forwarding, a route error (RERR) message can be used to inform the data message originator of the problem detected. The RERR can also be used when the data message destination is unknown by the intermediate node. Table 1 summarizes the control messages of LOADng and presents its fields with a brief description.

In the process of route discovery, control messages are used in conjunction with an Information Base maintained by each network node. According to the content of the control messages, the Information Set of nodes are fed and updated. The main elements of the Information Base are the following: Routing Set, Blacklisted Neighbor Set, and Pending Acknowledgment Set. The Routing Set is composed by route tuples entries that store data about the neighbor nodes. Based on the Routing Set, a node can verify the existence of a path to a destination or the necessity of starting a new route discovery process. The Blacklisted Neighbor Set is responsible for storing the addresses of nodes with possible communication inconsistencies that make the bidirectional linkage unavailable. The Pending Acknowledgment Set records information about the RREP messages sent with the field `ackrequired` defined as true. Table 2 presents the fields and the descriptions of the main components of the Information Set.

Table 1. LOADng Control Messages.

RREQ Message	
Field	**Description**
addr-length	Defines the length of the addresses used by originator and destination nodes
seq-num	Indicates the sequence number that uniquely identifies each message generated by the originator node
metric-type	Determines the type of metric used by the message originator node
route-metric	Defines the value of the route metric of the path
hop-count	Indicates the number of hops that the message has traversed
hop-limit	Indicates the maximum number of times that a message can be forwarded
originator	Specifies the address of the message originator
destination	Specifies the address of the message destination
RREP Message (contains the same fields as RREQ plus the above)	
Field	**Description**
ackrequired	Indicates the necessity of generating an RREP_ACK when be received
RREP_ACK Message	
Field	**Description**
addr-length	Defines the length of the addresses used by originator and destination nodes
seq-num	Indicates the sequence number of the RREP messages that was triggered the generation of the RREP_ACK
destination	Specifies the address of the message destination
RERR Message	
Field	**Description**
addr-length	Defines the length of the addresses used by originator and destination nodes
errorcode	Indicates the error code of the message
unreachableAddress	Specifies the address of the node unable to be reached
originator	Specifies the address of the message originator
destination	Specifies the address of the message destination
hop-limit	Indicates the maximum number of times that a message can be forwarded

Table 2. LOADng Information Base.

Routing Set	
Field	**Description**
R_dest_addr	Indicates the address of the route destination
R_next_addr	Indicates the address of the next hop in the path to the route destination
R_metric	Specifies the value of the metric computed for the path to the destination nodes
R_metric_type	Determines the route metric used to compute the metric value
R_hop_count	Specifies the number of hops to the route destination
R_seq_num	Indicates the sequence number of the control message used to generate the entry in the set
R_bidirectional	Indicates if the message is bidirectional
R_local_iface_addr	Specifies the communication interface used to reach the route destination (used only when a node has more than one interface)
R_valid_time	Specifies the length of time the entry is considered valid in the set
Blacklisted Neighbor Set	
Field	**Description**
B_neighbor_address	Indicates the address of the blacklisted neighbor
B_valid_time	Specifies the length of time the entry is considered valid in the set
Pending Acknowledgement Set	
Field	**Description**
P_next_hop	Indicates the address of the node that the RREP was sent
P_originator	Indicates the address of the RREP originator
P_seq_num	Defines the sequence number of the sent RREP
P_ack_received	Determines whether the pending RREP_ACK was received
P_ack_timeout	Specifies the length of time the entry is considered valid in the set

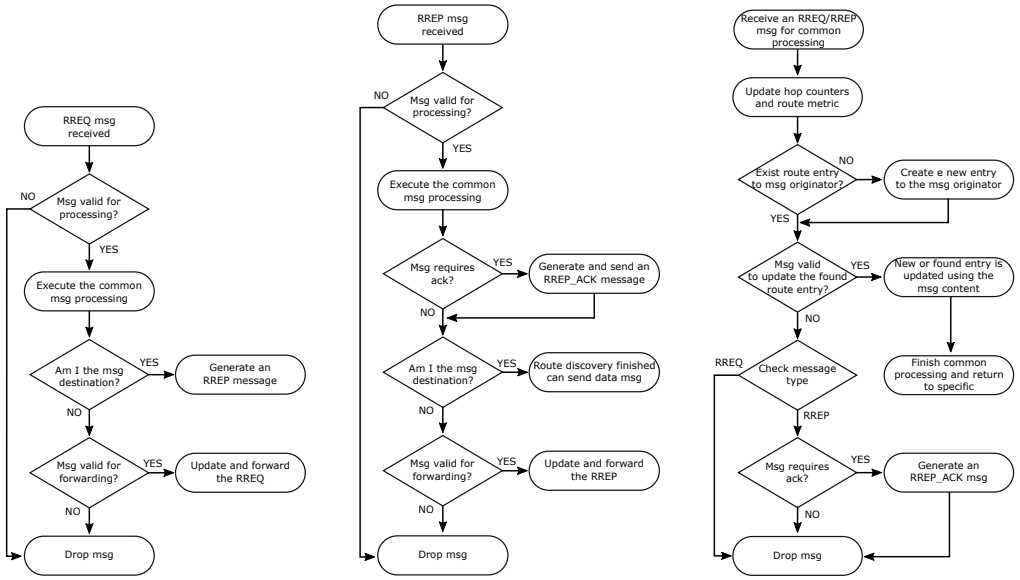
3.3. LOADng Route Discovery

When a node wants to send a data message, it should look for a route to the message destination on its Routing Set. If the path is found, the node should forward the message to its destination through the next hop node. The message forwarding process is described in detail in Section 3.4. However, if the wanted destination is not found, the node should start a new route discovery process.

In the routing discovery process, the node generates a new RREQ message, defining itself as `originator` and the address of the desired destination in the `destination` field. It should also set a unique `seq-num` to the RREQ and define the other message fields. Then, the node should broadcast the generated RREQ to its neighbors.

Each receiver of the RREQ message should execute its processing according to the flowchart presented in Figure 2a. In the first verification that checks the length of addresses and other details, the message, if considered valid, is subjected to the common processing used both for RREQ and RREP messages. In the common processing (presented in the flowchart in Figure 2c), the node should update the fields of `hop-count`, `hop-limit`, and `route-metric` from the message. In sequence, the node should search for a route entry for the message originator on its Routing Set. If the route is not found, a new route entry for the message originator is created. Then, the created or found route entry is compared with the fields of the received message to verify whether or not the message can be used to update the route entry. If the message is valid, the route entry is updated, the common processing is finished, and the message returns to its specific processing. In contrast, if the message is not used to update the route entry, the node should verify the message type, send an RREP_ACK if required, and drop the message. When the message returns to the specific RREQ processing, the node should check whether it is a message destination. If negative, the node should verify whether the message is valid to be forwarded (checking the `hop-count` and `hop-limit`), updating its fields, and forward it using broadcast. Otherwise, if the node is the RREQ destination, it should generate an RREP message to answer the request from the RREQ originator.

The generated RREP message should have the address of the RREQ originator as `destination`, the address of its originator in the `originator` field, and a unique `seq-num`. After being generated, the RREP should be sent in unicast to its destination. Thus, the RREP originator should look for a route entry to the destination on its Route Set and forward the message to the `R_next_hop` node. Note that the route entry should be found after it has been created by the RREQ message being received. A node that receives the RREP message should perform its processing as described in the flowchart in Figure 2b. After the first validation, the message is submitted to the common message processing (similar to an RREQ message and following the flowchart in Figure 2c). After this processing, the RREP receiver should verify the necessity of generating and sending an RREP_ACK message. In sequence, the node should check whether it is the message destination. If not, the node should consult whether the message is valid to forwarding, verify its Routing Set looking for an entry to the RREP destination and send the message to the `R_next_hop` node using unicast. Otherwise, if the node is the RREP destination, the route discovery process is completed, and the data message can be sent using the constructed path.



(**a**) RREQ message processing (**b**) RREP message processing (**c**) Common message processing

Figure 2. Flowcharts of LOADng RREQ and RREP control messages processing.

3.4. LOADng Data Message Forwarding

In the data message sending process, the node should use the path created in the route discovery process to deliver the data message to its correct destination. Thus, the node consults its Routing Set looking for an entry that matches the message destination. In sequence, the node should forward the message to the next hop of the found route entry. According to the latest LOADng specification, a node should always refresh the valid time of a route entry that it uses. The intermediate node that receives a data message should forward to the next hop of the path based on the information in its Routing Set. This process occurs until the message reaches its final destination. If an intermediate node does not find a route entry that matches the message destination, it should perform a new route discovery process to recover the broken path. If the path recovery does not succeed, the node should generate an RERR message to inform the data message originator of the impossibility of delivering the message successfully.

3.5. SmartRREQ Enhancement for LOADng

To reduce the number of control messages exchanged during the route discovery process, the SmartRREQ enhancement was proposed for LOADng [26]. With the use of SmartRREQ, the node should start the route discovery process with an RREQ containing a new `smart-rreq` flag set as true. Every node that receives a SmartRREQ (RREQ message with `smart-rreq` true) should perform additional processing in the RREQ message handling. After executing all the initial processing, and after verifying whether the message is valid for forwarding, the node should perform the specific processing of SmartRREQ. Thus, the node checks whether it owns a route entry on its Routing Set to the message destination with `R_next_hop` that is different from the previous hop of the received SmartRREQ. If this condition is satisfied, the node should transmit the SmartRREQ message in unicast to the next address found. The next hop that receives the SmartRREQ message should perform the same processing until the message reaches the final destination. If a node does not find a route entry to the SmartRREQ destination, the message should be forwarded using broadcast. The destination of a SmartRREQ should answer the request by generating a normal RREP. Hence, the SmartRREQ enhancement can reduce the number of broadcast transmissions, thereby contributing to

reducing the control message overhead required to discover a new route and decreasing the network energy consumption.

4. Proposed LOADng Enhancement for IoT Networks

This work proposes an enhancement for the LOADng protocol in IoT networks composed of devices with different capabilities. The proposed LOADng-IoT introduces a new route discovery process dedicated to finding devices with the capacity to forward special messages from other nodes. The following subsection explains the IoT applications scenario in which the proposed approach can be applied. In the sequence, LOADng-IoT is fully described, including its features, requirements, and operation.

4.1. Considered IoT Applications and Network Model

This work considers an IoT network as presented in Figure 1. The network is composed of simple nodes and Internet-connected nodes, hereafter referred to as INs. The simple nodes represent devices with low capacity, equipped with IEEE 802.15.4, which are unable to realize a direct Internet connection. On the other hand, INs represent devices with high potential, equipped with IEEE 802.15.4 and another communication interface, which are able to provide direct Internet connectivity (such as 4G, Ethernet, or Wi-Fi). All of the network nodes can generate simple messages, which are sent to any network node, and Internet messages, which are directed to external Internet services. The simple messages are locally generated and processed by the nodes. In contrast, Internet messages are created locally and need to reach external services using the Internet. Thus, the INs can directly send Internet messages once they have Internet connectivity. However, simple nodes that generate Internet messages need to find an IN to work as a gateway so that they can then forward their packets. Thus, an IN that receives an Internet message from a simple node should handle and forward the message to the final IP destination using other communication interfaces.

A smart home (SH) IoT application can be used to exemplify the use of the presented network model. In a SH, the smart objects with a low necessity of consulting external Internet services (such as lights, windows, doors, showers, and air-conditioners) can be represented by simple nodes. In contrast, smart objects that require continuous Internet-connection (such as smart meters, smart TVs, smartphones, tablets, and routers) can be represented by INs linked to the Internet using cellular network or optical fiber. Thus, as an example, a smart window can occasionally consult an external Internet service to verify the weather forecast. The smart window, to access the service, should use an IN as its gateway to the Internet. Besides, in a context of a local message exchange, an air-conditioner, when activated, can send a local message to close the smart windows without the necessity of using the Internet to perform this communication.

Current routing solutions can address the presented network model when the simple nodes have been previously configured with a default gateway to forward their Internet messages. Thus, prior knowledge of the nodes with Internet connection capacity is required to then define a gateway for each simple node. This approach, although functional, can give rise to several issues. The obstacles that can occur using this simplistic approach are identified as: (i) high deployment time is required to define the gateway of each simple node; (ii) Internet-connected nodes can be overloaded with Internet messages from simple nodes; (iii) bad deployment can make simple nodes create long paths to their gateways; (iv) simple nodes can become unable to send Internet messages if their gateways lose Internet connectivity.

Based on the exposed constraints, and seeking to better address the requirements of the described IoT network scenario, this work proposes a new enhancement for the LOADng protocol that is able to optimize the route discovery process for INs and improve the network performance. The proposed LOADng-IoT can simplify the discovery of Internet routes, avoiding the necessity of the prior definition of gateways for Internet messages. In addition, the proposal reduces the number of control messages

required to construct paths to INs and makes the data message forwarding process more reliable. The following subsections present a detailed description of the proposed LOADng-IoT.

4.2. Proposal Overview

The proposed LOADng-IoT is composed of three components: the Internet route discovery process, the Internet Route Cache (IRC) system, and a new error code for RERR messages. The first component is responsible for finding IN nodes without the requirement of previous knowledge of its address in the local network. Thus, a node that wants to send a message to the Internet should start an Internet route discovery process by broadcasting a special RREQ, named RREQ-IoT. The message has the objective of seeking an IN that can be used as a gateway for the RREQ-IoT originator. To reduce the number of broadcast transmissions, an intermediate node that knows a route for an IN can forward the RREQ-IoT message to it using unicast transmission (in the same way as SmartRREQ). When an IN receives an RREQ-IoT message, it should generate a special RREP to answer the request. This message, named the RREP-IoT, is forwarded in unicast via the opposite route created by the RREQ-IoT. Each node that receives an RREP-IoT should create an entry on its Routing Set with the information that the message originator has an Internet connection. When the RREP-IoT reaches its destination, the node should immediately start to send the Internet data messages. The proposed Internet route discovery process of LOADng-IoT is fully described in Section 4.4.

The second component is responsible for storing the Internet routes (routes to Internet-connected nodes) removed from the Routing Set. During the Internet route discovery process, the nodes create entries on the Routing Set to the INs. These entries, which have a valid time, can expire and be removed from the Routing Set when not used. Thus, to reduce the number of transmissions in a new Internet route discovery and to allow the nodes to follow a previously known Internet route, these entries, when removed, have some of its information inserted in a new data structure, the IRC. The IRC should always be consulted when a new Internet route discovery process is started and should, when possible, indicate a previously known Internet route to direct the discovery process. The IRC is optional and should be adopted according to the hardware capacity of the network devices. A complete description of the proposed IRC is presented in Section 4.5.

The third component is a new error code for RERR messages. The `INTERNET_CONN_LOST` error code should be used to indicate to an Internet data message originator that it was not possible to forward the message to the Internet. Thus, the receiver of the RERR should start a new Internet route discovery process to find a new IN to transmit its Internet messages. The generation, processing, and functioning of the proposed new error code for RERR messages are detailed in Section 4.6.

4.3. LOADng-IoT Required Increments and New Features

The proposed LOADng-IoT requires several increments in the existing default LOADng structure. Table 3 shows the fields added to the LOADng to make possible the use of IoT enhancement. Basically, a new field was inserted in both the RREQ and RREP to allow the nodes to identify the messages as IoT and realize special handling. In the Routing Set, a new field was included to enable a route to be identified as an Internet route. Hereafter, in this paper, the term Internet route will be used to refer to a route whose destination is a node with an Internet connection.

In addition to the increments previously presented, the LOADng-IoT also presents several new features required to improve the performance of the studied IoT networks. Table 4 presents these features. The data structure of the proposed IRC is composed of two fields to store the address related to the route entry removed from the Routing Set. The memory required by the IRC structure varies according to the adopted addressing scheme (IPv6, IPv4, or Rime). However, it is important to note that the IRC is optional and could be used just in nodes with slightly greater memory capacity. The option to use or not use the IRC should be configured in each node at the deployment moment using the configuration parameters dedicated to it. In addition, it is possible to define the number of entries in the IRC set according to the memory restriction of the nodes. Considering that, in the Routing

Set, the Internet routes may require a valid time that is different from that of common routes, a new parameter was created to allow this feature. Finally, a new error code was created to be used with the LOADng-IoT. The new specific error code allows an Internet node to inform that a received Internet message was not forwarded due to loss of Internet connection. Thus, the Internet data message originator can find a new node to forward its messages.

Table 3. LOADng-IoT required increments in to LOADng structure.

RREQ and RREP	
Field	**Description**
iot	Indicates that the RREQ or RREP message is *-IoT; when used in RREQ, indicates that the originator is searching for an Internet node; when used in RREP, indicates that the originator has an Internet connection.
Routing Set	
Field	**Description**
R_Internet_conn	Indicates the address of the blacklisted neighbor

Table 4. LOADng-IoT new features.

Internet Route Cache	
Field	**Description**
RC_dest_addr	Specifies the address of the destination removed from the Routing Set
RC_next_hop_addr	Specifies the address of the next hop to reach the destination removed from the Routing Set
Configuration Parameters	
Field	**Description**
R_INTERNET_HOLD_TIME	Defines a valid time to an Internet route entry in the Routing Set
USE_INTERNET_ROUTE_CACHE	Specifies if the node uses the IRC mechanism
NUM_ROUTE_CACHE_ENTRIES	Indicates the number of entries supported in the IRC
Error Code	
Field	**Description**
INTERNET_CONN_LOST	Indicates the impossibility of forwarding a data message to the Internet due to lost; the error code is 253.

4.4. Internet Route Discovery Process

The Internet route discovery should always be initiated when a node does not find a route entry to an IN on its Routing Set. Thus, the node should create an RREQ with the flag `iot` set as `TRUE` (hereafter, this message is named RREQ-IoT). Then, the node defines its own address as the RREQ-IoT `originator` and `destination`, and a set new unique `seq-num`. In the following, the node should consult its IRC (if in use) to verify the existence of a previously known Internet route. At this point, the IRC is considered to be empty, and will be explored in the next subsection. Thus, the node should transmit the generated RREQ-IoT in a broadcast to its neighbors.

Each node that receives an RREQ-IoT should perform its processing according to the flowchart in Figure 3. Thus, the node first checks whether the message is valid for handling and then conducts the common message processing. The common message processing is done in a similar way as in the standard RREQ message and following the flowchart in Figure 2c. Additionally, during this processing, at the moment a new entry is created in the Routing Set, the node should include the information about the Internet connection status of the message originator. As the RREQ-IoT message originator does not have an Internet connection, the route entry that was created based on the received RREQ-IoT should have the `R_Internet_conn` set as `FALSE`. In addition, if the received RREQ-IoT is used to update the

information on an existing route entry, the `R_Internet_conn` should never be changed to `false`. If the common processing is concluded without the message being dropped, the message handling proceeds to the next step. Thus, in the specific RREQ-IoT processing, the node should check whether it has an active Internet connection (the information on whether or not the node has an Internet connection can be obtained in several ways; this work does not mandate a specific way of doing this. However, as a suggestion, the routing layer can be informed about the Internet connection status through the application layer.). If the node identifies an active Internet connection, it should reply to the RREQ-IoT by generating and transmitting an RREP-IoT message to the request originator. However, if the node does not have an Internet connection, it should check whether the message is valid for forwarding. If true, the node searches for an Internet route on its Routing Set. Thus, if it is found, the node should use a mechanism similar to SmartRREQ to assist the path creation. Hence, the node changes the RREQ-IoT destination to the found `R_destination`, updates the other message field and sends the message in unicast to the `R_next_hop`. If more than one Internet route exists in the Routing Set, the node should select the one that best address the used routing metric. However, if an Internet route entry is not found in the Routing Set, the node should use the IRC mechanism, if it is activated. Considering that the node does not use the IRC, it should update the RREQ-IoT message fields and transmit the message in broadcast. Independent of the transmission mode of the RREQ-IoT (unicast or broadcast), the next receiver of the message should perform the process previously described.

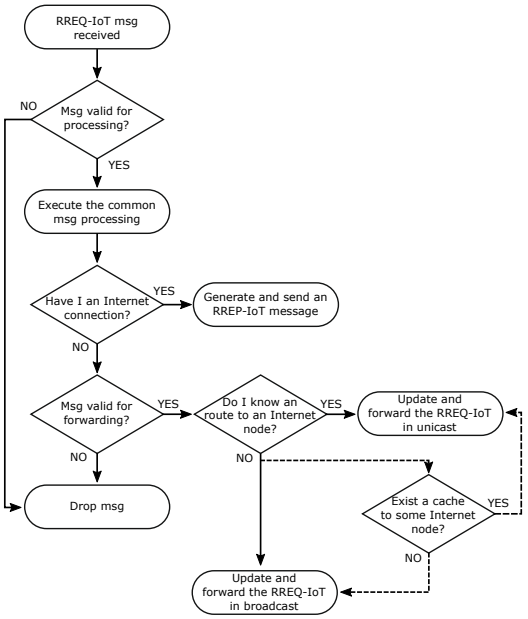


Figure 3. Flowcharts of RREQ-IoT control messages processing. Dotted lines represent the optional flow used when IRC is adopted.

Please note that unlike the normal route discovery process, the Internet route discovery process searches for any IN rather than a defined destination address. During the RREQ-IoT message processing, the node does not verify whether it is the message destination, but whether it has an active Internet connection. In addition, it is possible that an intermediate node that knows an Internet route will change the RREQ-IoT destination and redirect the Internet route discovery. Figure 4 explains how this may occur. Consider that node *A* needs to send an Internet message and has begun an Internet route discovery process. Node *B* has received the RREQ-IoT from *A* and begun its processing. *B* does not have an Internet connection but finds an Internet route to node *C* on its Routing Set. Thus, *B* changes the RREQ-IoT destination to node *C* and forwards the message in unicast to *C*. Thus,

inspired by the SmartRREQ, the Internet route discovery process is optimized to reduce the number of broadcast transmissions, thereby contributing to the reduction of energy consumption.

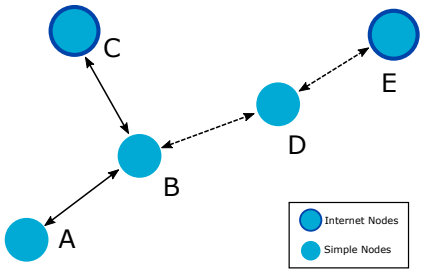


Figure 4. LOADng-IoT Internet route discovery process. Dotted lines represent the linkages allowed but not used.

As previously explained, a node with an Internet connection that receives an RREQ-IoT should generate a reply message to answer the request. Thus, the node generates an RREP-IoT, defines its own address as `originator`, and the address of RREQ-IoT originator as `destination`. The message should also have a new unique `seq-num`. Then, the node transmits the RREP-IoT message in unicast to the next address in the path to the destination.

The node that receives an RREP-IoT should perform its processing, which is very similar to the normal RREP processing, following the flowchart in Figure 2b. Thus, the node verifies whether the message is valid for handling and then executes the common message processing. In the common message processing of an RREP-IoT, all created or updated route entries to the message originator should include the information that `R_Internet_conn` is `TRUE`. In addition, the `R_valid_time` of the Internet route should be, by default, two times greater than the valid time of regular routes. Since only Internet-connected nodes can generate RREP-IoT messages, the intermediate nodes that forward RREP-IoT messages should not be included as `R_Internet_conn` set as `TRUE`. Thus, the RREP-IoT message performs the maintenance of routes and creates the path to the Internet nodes. At the end of the common processing, the node can generate an RREP_ACK message, if necessary. In the LOADng-IoT, the RREP_ACK generation, transmission, and processing are equal to the default LOADng. In sequence, the RREP-IoT receiver verifies whether it is the message destination. If false, the node should check whether the message is valid to forward, perform the message update, and transmit it to its destination. Otherwise, if the node is the message destination, the Internet route discovery process is completed, and the node can begin the sending of Internet messages.

In the described Internet route discovery process, all nodes that receive an RREQ-IoT should reply using an RREP-IoT. Thus, it is possible for more than one RREP-IoT to be received by the request originator. This behavior makes the construction of several Internet routes possible, one for each different IN. Hence, a node that intends to send an Internet message should look up its Routing Set and find the best path among those available. The selection of the best Internet route should be made based on the used route metric or the lowest number of hops. The process of Internet message sending and forwarding is described in detail in Section 4.7.

4.5. Internet Route Cache for LOADng-IoT

In the course of the network functioning, all route entries of the Routing Set can be removed when the valid time expires. This process occurs to allow the nodes to reduce the memory usage and make the creation of paths to other nodes possible. In the LOADng-IoT, the valid time of the regular routes and the Internet routes can be different and should be adjusted according to the expected traffic in the network. However, considering a scenario where both message types are equality generated, an Internet route tends to be used more by representing a path to all Internet messages. Thus, by

default, the authors suggest that the Internet routes have a valid time that is two times greater than regular routes.

Even with a higher valid time, Internet routes that are not used can expire and be removed from the Routing Set of the nodes. Thus, when a node needs to send a new Internet message, the whole Internet route discovery process should be completed again, transmitting several control messages and expending more network resources. To reduce this problem, LOADng-IoT offers an optional improvement that is able to minimize the control message overhead during the construction of Internet routes. This mechanism, the IRC, stores the most relevant information about the last Internet route entries removed from the Routing Set. Then, when it is necessary to perform an Internet route discovery, the node should check its IRC to verify the existence of a previously known Internet route. If positive, the node can direct the Internet route discovery to the destination of the found entry in the IRC.

An entry can be removed from the Routing Set due to valid time expiration or to lack of memory for the insertion of a new entry. Thus, with the use of the IRC, if the removed entry is an Internet route (i.e., `R_Internet_conn` is `TRUE`), its `R_next_addr` and `R_dest_addr` are used to create a new route cache entry that is inserted in the IRC set. As presented in Section 4.3, the number of route cache entries should be previously defined and should consider the memory limitation of the nodes. The authors suggest that this number be incremented by one for every four possible entries in the routing table. Thus, if the size of the routing table is four, the number of IRC entries should be one; if the size of the routing table is eight, the number of IRC entries should be two. It is also possible to consider the reduction of the number of Routing Set entries to allow the use of the IRC in devices with severe memory restrictions.

The IRC entries do not have valid time and only can be removed by the reception of an RERR message (discussed in Section 4.6). According to the number of route entries defined in the IRC, the oldest entries are removed when a new entry needs to be inserted. In addition, the search for an IRC entry should always get the most recent entry in the set. Thus, the IRC set should work like a stack, where a new entry must be inserted at the start of the set, and the find for an entry should always return the head of the list. When the set is full, the node should to remove the last element in the list and insert the new entry at the beginning.

With the use of the IRC, the nodes should verify its route cache set before starting a new Internet route discovery process. If an entry is found, the node should direct the Internet route discovery process to the destination of the found entry. Thus, the node creates an RREQ-IoT with a destination equal to the found `RC_dest_add` and sends the message to the `RC_next_hop` in unicast. The node that receives the RREQ-IoT should realize the message handling normally, as described in Section 4.4. As shown in the flowchart in Figure 3, during the RREQ-IoT processing, a node that does not have an Internet connection should consult its Routing Set to verify the existence of any Internet route. Please note that at this point, as occurs in the normal processing of RREQ-IoT, the destination of the unicasted RREQ-IoT can be changed and the message can be redirected to another node with an Internet connection.

This situation is exemplified in Figure 4. Consider that node *A* needs to send an Internet message and does not find an Internet route on its Routing Set; *A* checks its IRC and finds an entry to the IN *E* with next hop *B*. Thus, *A* generates an RREQ-IoT with a `destination` equal to the *E* address and unicasts it to node *B* (the found next hop to *E*). *B* receives the RREQ-IoT from *A* and realizes its processing normally. However, *B*, at the moment of checking its Routing Set, finds an Internet route to node *C*. Thus, *B* changes the received RREQ-IoT destination to *C* and forwards the message in unicast. Node *C* then receives the request from *A* and sends a reply offering the required Internet route. This behavior is accepted because the intention of an RREQ-IoT is to reach a route to the Internet, independent of the IN providing the connection. In addition, this redirection of the RREQ-IoT ensures that the Internet route discovery process follows with most recent information (considering that the information provided by the Routing Set is frequently newer than the information provided by the IRC). However, if an Internet route is not found in the Routing Set of the intermediate node,

the intermediate node should verify its IRC, change the message destination (if necessary), and then forward the message in unicast to an IN that is able to provide an Internet connection to the RREQ-IoT originator. Finally, whether the IRC set is empty, the node should change the destination address of the RREQ-IoT to the same address as its originator and send the message in broadcast. This process "converts" the RREQ-IoT received in unicast in a normal RREQ-IoT to be broadcasted and the Internet route discovery process can continue until an IN is reached.

The use of the IRC mechanism allows the nodes to reduce the number of control messages required to construct an Internet route. The cache mechanism is used to direct the RREQ-IoTs to a previously known IN. Thus, when adopted, the IRC contributes to the reduction of the number of packet collisions, minimizes energy consumption, and improves network efficiency. As explained, the entries in the IRC set are only removed by lack of memory or the reception of an RERR message. This process is discussed in the description of the new error code proposed in this work in Section 4.6.

4.6. Internet Lost Error Code for LOADng-IoT

Due to unexpected situations, the Internet nodes can sometimes lose Internet connection. To avoid these nodes receiving Internet messages when they have no connection, this paper proposes a new error code with the function of advising the neighbor nodes that the Internet connection has been lost. The error code, described as "Internet connection lost" (or *INTERNET_CONN_LOST*), is defined by code 253. The used code number is included in the range of experimental use codes according to the most recent LOADng [10] specifications and can be altered in the future.

During network functioning, when an IN receives an Internet message from another node, it should forward the message to the Internet address destination. If the node detects that the connection has been lost or that it is not possible to realize the forwarding, it should generate an RERR message. The generated RRER message, which has the same structure as the normal RERR presented in Table 1, receives the `errorcode` 253, the `originator` as its own node address, and the `destination` as the originator of the not forwarded Internet message. Notice that, in this case, the field `unreachableAddress` can be ignored when this error type has not used it. Then, the message is transmitted in unicast to the previous hop of the received Internet message.

A node that receives an RRER with code 253 should perform its processing according to the flowchart presented in Figure 5. Thus, the receiver node should initially decrement the `hop-limit` field. Then, the node should check its Routing Set and, if an entry is found, change the `R_Internet_conn` to `FALSE`. If the RERR receiver uses the IRC, it should also verify whether an entry for the RERR originator exists on its IRC set. If an entry is found, the node should remove it. Notice that the route updated to `R_Internet_conn` false is not removed from the table and is not included in the IRC set. This procedure avoids the node trying to start a new Internet route discovery from using information about the node with the connection lost. In sequence, the node verifies if itself is the message destination. If true, the RERR process is completed and the message is not forwarded. Otherwise, the node should ascertain whether the message is valid for forwarding, update the message fields, and send the message to its destination.

An RERR message with code 253 can also be generated by an intermediate node in the path to the destination of an Internet message. An intermediate node that receives an Internet message and detects that the message destination does not have an active Internet connection should start a new Internet route discovery process to find a new Internet path. If the path is not successfully created, the RERR message is generated to the Internet message originator.

The use of the proposed error code allows the network to reduce the number of Internet messages sent to a node without an Internet connection. Thus, the nodes are able to find alternative Internet routes when they receive an RERR message with code 253. As a benefit, the number of packets lost can be reduced, which contributes to improving the network efficiency and reliability.

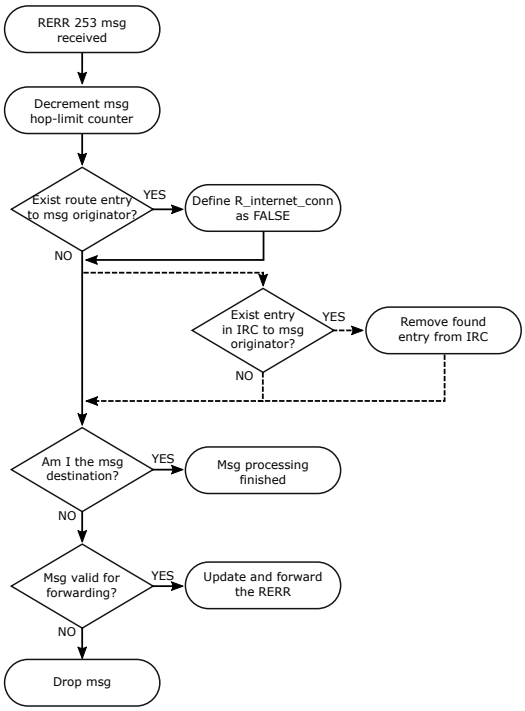


Figure 5. Flowcharts of RERR control messages with code 253 processing. Dotted lines represent the flow used only when IRC is in use.

4.7. LOADng-IoT Data Message Forwarding

In the network operation with the use of the proposed LOADng-IoT, the nodes that intend to send data messages should verify its Routing Set and execute the route discovery process, if necessary. After constructing the routes, simple messages and Internet messages can be sent to its destinations through the next hop node. In the case of simples messages, as the LOADng constructs only one path to each destination, it is not possible to compare the best route for sending (the best path selection is made during the process of route discovery). However, to send an Internet message, the node should to select the best path based on the selected routing metric since the Internet route discovery process can create several routes to different INs. Thus, it is possible to choose the best path among those available. The nodes that receive a data message (both simple and Internet) should consult its Routing Set to find the next hop to the destination node. This process should continue until the data message is delivered.

Independent of the message type, it is possible for a broken route to occur during the message forwarding process. In this case, the intermediate node that was not able to forward the message detects the broken path, queues the data message, and starts a new route discovery process according to the message type. If the message is simple, the node should begin a normal route discovery using the standard LOADng procedure. However, if the message is directed to the Internet, the node should start an Internet route discovery following the process described in Section 4.4. Thus, the node should consult its IRC set (if enabled) and send an RREQ-IoT unicast or broadcast following whole the process. In both cases, if the route is constructed successfully, the queued data message is forwarded normally. Otherwise, if it is not possible to reconstruct the path, the node should generate an RERR according to the data message type and send it to the data message originator.

5. Performance Evaluation and Results Analysis

This section presents the performance assessment realized to evaluate the behavior of the proposed solution. The Cooja simulator/emulator, which is a part of the Contiki O.S. [27], was used for this purpose. Although the use computational simulations cannot precisely represent the behavior of a network in the real world, it can allow a fair comparison among routing protocols since it makes the reproduction of an identical environment possible for all studied proposals [25]. In addition, because it works as an emulator, Cooja permits the replication of the hardware conditions of real nodes such as Tmote Sky, Zolertia Z1, and others. Thus, each emulated node represents the identical hardware conditions of real devices in terms of memory usage and processing capacity.

The proposed LOADng-IoT was compared with the most recent version of LOADng and LOADng-SmartRREQ. The objective was to analyze the behavior of the proposed solution with the other proposals in different scenarios. Thus, situations were created with three topology organizations: grid sparse, random dense, and mobility dense. For all the topologies, the number of nodes in the network changed from 16 to 64. This quantity was chosen because it can represent the majority of the existing small-scale IoT application scenarios, mainly in smart homes. The following itemization presents more details on the used grid, random, and mobility topologies.

- Grid Sparse Scenarios: the network nodes were organized in linear grids of nxn nodes. The simulated area grew together with the number of nodes. Thus, a fixed network density was maintained where the nodes had between two and four neighbors.
- Random Dense Scenarios: the different quantity of nodes was randomly deployed in an area of 200 square meters just once. Thus, the random deployments were the same for all compared proposals. The simulated area was the same for the different quantities of nodes. Hence, the network density grew with the increase in the number of nodes.
- Mobility Dense Scenarios: the nodes were deployed with the same positions of random dense scenarios in an area of 200 square meters. However, the nodes with an Internet connection was able to move in the whole area of the studied environment.

For all scenarios and the different quantities of nodes, the simulation time was 600 s. In the application, all network nodes generated and sent data messages in variable intervals of between 10 and 15 s. The minimum data message interval was defined as 10 to avoid the nodes being overloaded with several data messages while they were still realizing a route discovery process. This measure was required because the nodes do not implement a significant buffer for data messages in the routing layer. Thus, all data messages generated inside a route discovery process are lost. The data message generation should be able to address the requirement of Equation (1):

$$2 * (1 + RREQ\_RETRIES) * NET\_TRANSVERSAL\_TIME \quad (1)$$

where $RREQ\_RETRIES$ is the maximum number of route discovery retries and $NET\_TRANSVERSAL\_TIME$ is the expected time to a control message traversing the whole network. The generated data messages were simple messages or Internet messages, as explained in Section 4.1. Both messages were generated randomly with a chance of 50%. Thus, the simulated scenarios created a network traffic pattern that merged P2P (sending simple messages from a local node to another local node) and MP2P (sending Internet message from local nodes without an Internet connection to a local node with an Internet connection).

In the simulated scenarios, the number of INs grew according to the number of nodes in the network, representing around 10% of network devices. To simulate an environment where the IN could, periodically, lose its Internet connection, these nodes had a random Internet connection time of between 60 and 90 s. After this time, the connection remained lost for a random time of between 0 and 60 s. In sequence, the connections were once again reestablished with a new random time of between 60 and 90 s. The most critical parameters of the simulation are presented in Table 5.

The parameters of the nodes used in the simulation environment are presented in Table 6. Notice that INs simulate an additional communication interface for the Internet connection. In the mobility scenario, the mobile nodes, which are the INs, move according to the random waypoint mobility model [28]. The BonnMotion 3.0.1 [29] tool was used to generate the movement of the mobile nodes. Table 7 presents the parameters of mobility used in this study. All studied approaches were based on LOADng and used the same settings presented in Table 8. However, the proposed approach (LOADng-IoT) used two more parameters, presented in Table 9.

The following subsections show the obtained results for the metrics of the packet delivery ratio (PDR), the energy spent per delivered data bit (AES), the control message overhead per delivered data message (CMO), and the percentage of packets with low latency (PLL). For all studied metrics, the simulations were executed 30 times, and the results presented a confidence interval of 95%.

Table 5. Parameters of Simulation.

Parameter	Value
Network Area	150∼280 m^2
Number of Nodes	16∼64
Num. of Internet-connected Nodes	2∼6
Simulation Time	600 s
Radio Environment	Unit Disk Graph Model (UDGM)-Distance Loss
Transmission Range	50 m
Interference Range	50 m
TX and RX Chance	90%
Data Message Frequency	10 s∼15 s
Data Message Length	512 bits
Traffic Pattern	P2P and MP2P
Medium Access Control (MAC) Protocol	Carrier Sense Multiple Access (CSMA)
Radio Duty Cycle (RDC) Protocol	ContikiMAC
Check Channel Rate (CCR)	16 Hz

Table 6. Nodes Parameters.

Parameter	Value
Mote Type	Tmote Sky
Radio	CC2420
Max. Transmission Power	31 dBm
Supply Voltage	3.6 v
TX Current Consumption	21.0 mW
RX Current Consumption	23.0 mW
Low Power Mode (LPM) Current Consumption	1.2 mW
CPU Current Consumption	2.4 mW

Table 7. Mobility Parameters.

Parameter	Value
Mobility Model	Random Waypoint
Mobility Area	200 m^2
Max. Speed	3 m/s
Min. Speed	1 m/s
Max. Pause Time	60 s
Min. Pause Time	0 s

Table 8. Parameters of LOADng.

Parameter	Value
NET_TRANSVERAL_TIME	2
RREQ_RETRIES	1
RREQ_MIN_INTERVAL	2
R_HOLD_TIME	60
MAX_DIST	65,535
B_HOLD_TIME	4
MAX_HOP_LIMIT	255
RREQ_MAX_JITTER	1
RREP_ACK_REQUIRED	FALSE
USE_BIDIRECTIONAL_LINK_ONLY	FALSE
RREP_ACK_TIMEOUT	2
MAX_HOP_COUNT	255
NUM_RS_ENTRIES	8
NUM_BLACKLIST_ENTRIES	16
METRIC_TYPE	HOP_COUNT

Table 9. Parameters of LOADng-IoT.

Parameter	Value
R_INTERNET_HOLD_TIME	120
USE_INTERNET_ROUTE_CAHCE	TRUE
NUM_ROUTE_CACHE_ENTRIES	2

5.1. Packet Delivery Ratio

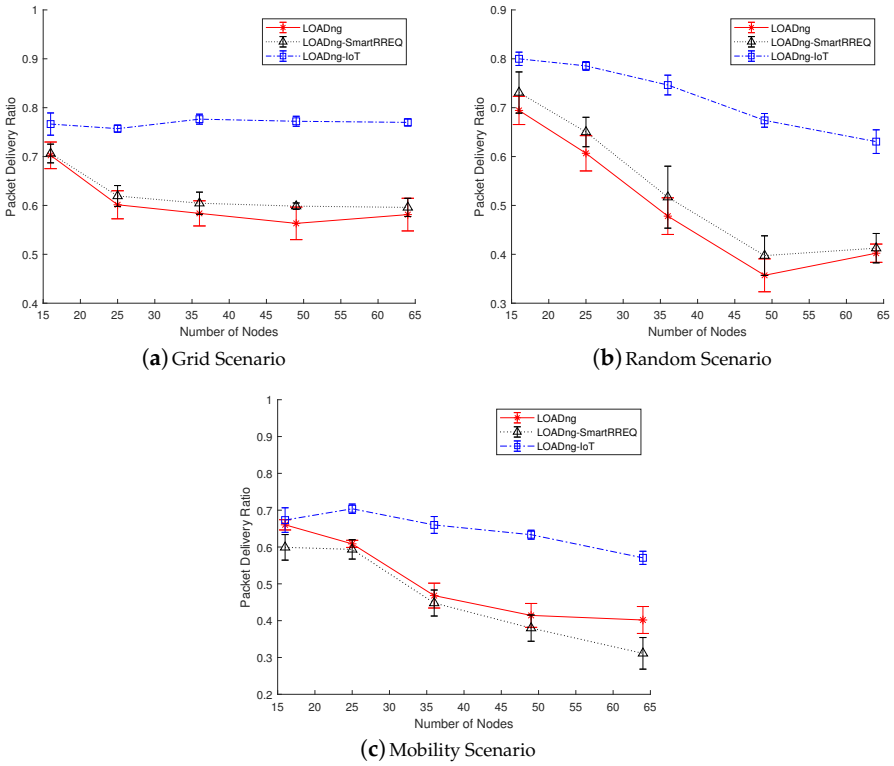
The PDR metric represents the number of data messages that were successfully delivered to the destination node. Thus, a high PDR represents an efficient network that is able to deliver the generated data messages with high reliability. This metric is constantly affected by the quality of the links among the nodes, the radio interference provoked by neighbor devices, and collisions with other data and control messages. In this paper, the Internet messages delivered for a node without an Internet connection were considered lost. The PDR value of the network was obtained according to Equation (2):

$$PDR = \frac{\sum_{i=1}^{N} Pr_i}{\sum_{i=1}^{N} Ps_i} \quad (2)$$

where N is the number of nodes in the network, Pr_i is the number of data packets received for each node i, and Ps_i is the number of data packets sent by each node i.

Figure 6 presents the results obtained for the PDR metric. In all studied scenarios, the proposed LOADng-IoT obtained better results when compared with the other approaches. In the grid scenario, where the network was sparse and the quantity of neighbor nodes was constant, the proposed solution presented satisfactory results, demonstrating its scalability and reaching a PDR of between 75% and 80%. In contrast, the other approaches decreased its performance with the increase in the number of nodes. The reason for this behavior was the dependency on a fixed Internet-connected node to send the Internet messages. In addition, networks using LOADng and LOADng-SmartRREQ were unable to detect an IN's loss of Internet connection. Thus, the Internet messages were continuously forwarded and lost due to the incapacity of the IN to route them to the Internet. With the use of the proposed approach, the INs were able to use the new error code to inform the Internet message originator when its Internet connection was lost. Thus, the message originator was able to start a new process of Internet route discovery to find a new gateway to forward its messages to the Internet. In the random and mobility scenarios, where the network was dense, the same behavior was perceived. However, the obtained PDR values were lower for all the studied networks, mainly in the mobility scenario. The reduction of the network performance is already expected when the network density grows. As the

number of nodes is increased in a fixed area, the probability of packet collisions grows, provoking a high packet loss. However, the proposed solution was able to obtain better results (in some cases, 40% better than default LOADng). The movement of INs also contributes to the packet loss since paths previously constructed can rapidly become unavailable due to the mobility of the nodes. In this case, the nodes that detect a route broken should use RERR messages to inform the data originator about the forwarding incapacity. This behavior provokes the necessity of a new route discovery implying in the transmission of control messages and, consequently, increasing the energy consumption. However, as the mechanism used by LOADng-IoT to find routes requires fewer control messages when compared with the other approaches, the losses provoked by Internet-connected nodes movement are reduced. Hence, with the lower necessity of radio transmissions, the probabilities of packet collisions and message loss are decreased.



(**a**) Grid Scenario (**b**) Random Scenario

(**c**) Mobility Scenario

Figure 6. Packet delivery ratio in function of number of nodes.

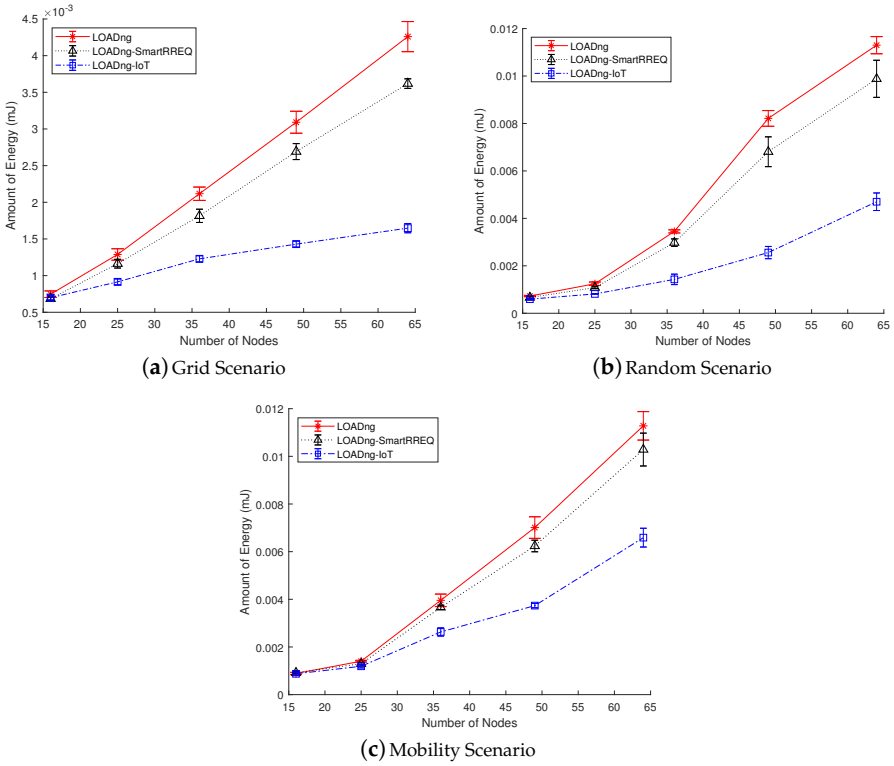
5.2. Average Energy Spent per Delivered Data Bit

The average energy spent per delivered data bit metric represents the amount of energy spent by the network to successfully delivery each data bit to its destination. Thus, the less energy spent to deliver the data successfully, the higher the power efficiency of the network. The results obtained by this metric are affected by the energy consumption of the nodes and the packet delivery ratio. The metric is computed using Equation (3):

$$AES = \frac{\sum_{i=1}^{N} Ec_i}{\sum_{i=1}^{N} Pr_i * M_{length}} \quad (3)$$

where N is the number of nodes in the network, Ec_i is the total energy consumed (in millijoules) by each node i, M_{length} is the length of the data message in bits, and Pr_i is the number data packets received by each node i.

Figure 7 shows the results obtained for the AES metric. In all studied network scenarios with more than 25 nodes, the proposed LOADng-IoT was able to obtain better results when compared with the other studied approaches. Considering the grid sparse scenario with 64 nodes, LOADng-IoT outperformed LOADng by ∼168%, and LOADng-SmartRREQ by ∼125%. In the mobility scenario, also with 64 nodes, the proposed approach is able to give a performance ∼73% better when compared with LOADng. In general, the reduction in the number of control messages used in the route discovery process allowed the proposed protocol to use fewer radio transmissions, provoking the energy consumption decrement. The improved process of route discovery for Internet nodes also enabled the protocol to find closer destinations to forward the Internet messages. Thus, the data messages delivered to the Internet-connected nodes required fewer transmissions, contributing to the reduction of energy consumption. Less transmission also implies a lower probability of packet loss caused by radio interference or packet collisions. Hence, by reducing the energy consumption and reaching a high packet delivery ratio, LOADng-IoT was able to spend less energy to deliver each data bit, exposing a high power efficiency in comparison with LOADng and LOADng-SmartRREQ.



(**a**) Grid Scenario (**b**) Random Scenario

(**c**) Mobility Scenario

Figure 7. Average energy spent per delivered data bit in function of number of nodes.

5.3. Control Message Overhead per Delivered Data Message

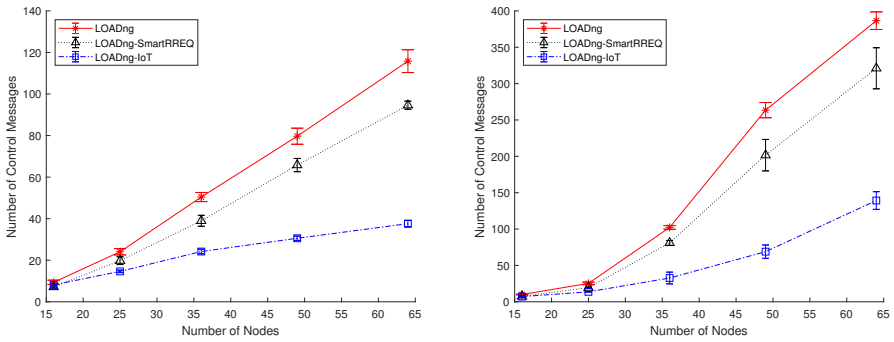
The control message overhead per delivered data message (CMO) metric shows the number of control message transmissions required to deliver each data message successfully. As in the previously presented metric, the results of the CMO are directly related to the packet delivery efficiency. Although control messages are not used during the forwarding of data packets, it is possible to count the overhead generated by the nodes to construct the path used to deliver the data messages. Thus,

calculating the ratio between the quantity of control message transmission used to discover the routes and the number of data packets delivered, it is possible to obtain the mean of control message transmissions required to deliver each data message. Thus, this metric is calculated according to Equation (4):

$$CMO = \frac{\sum_{i=1}^{N} CMt_i}{\sum_{i=1}^{N} Pr_i} \quad (4)$$

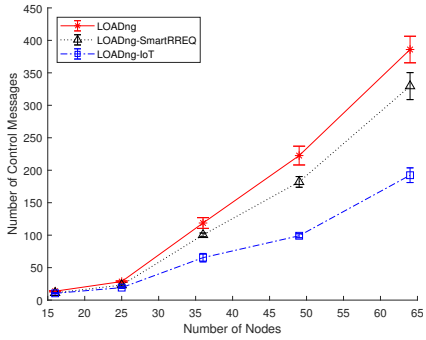
where N is the number of nodes in the network, CMt_i is the total number of control message transmissions performed by each node i, and Pr_i is the number of data packets received by each node i.

The results obtained for the CMO metric are presented in Figure 8. In all studied scenarios, the proposed solution was able to obtain better results concerning the compared approaches. The main advantage of using LOADng-IoT is the reduction in the number of control message transmissions during the route discovery process. The mechanism of Internet route discovery, which was inspired by LOADng-SmartRREQ, has reduced the number of control message broadcasts avoiding the nodes outside the already known route received and processed control messages irrelevant to them. In addition, the facility provided by the IRC has allowed the nodes to improve the IN discovery, since the route discovery is directed to a previously known IN using unicast transmissions. Notice that the mobility of the INs was not able to affect the benefits of IRC. The capacity of LOADng-IoT converting a unicasted RREQ into a normal RREQ allows the node to discover a path to an Internet-connected node even if IRC has directed the RREQ to an IN no more available. In the results from all studied scenarios, LOADng-IoT outperforms LOADng and LOADng-SmartRREQ being able to, in some cases, reduce by three times the number of control message transmissions required to deliver one data message. It is also important to note that, in the random and mobility scenarios, the control overhead is about two times greater than in the grid sparse scenario. In addition, in general, the number of control message transmissions is elevated in relation to the number of delivered messages. This behavior is justified by the configurations adopted in the network and protocol parameters. The data throughput is low and the valid times of the routes are short, meaning that a high number of route creation processes is required by the nodes. These parameters were purposely adjusted in this way to permit a better analysis of the behavior of the studied protocols.



(**a**) Grid Scenario (**b**) Random Scenario

Figure 8. *Cont.*



(**c**) Mobility Scenario

Figure 8. Control message overhead per delivered data message in function of number of nodes.

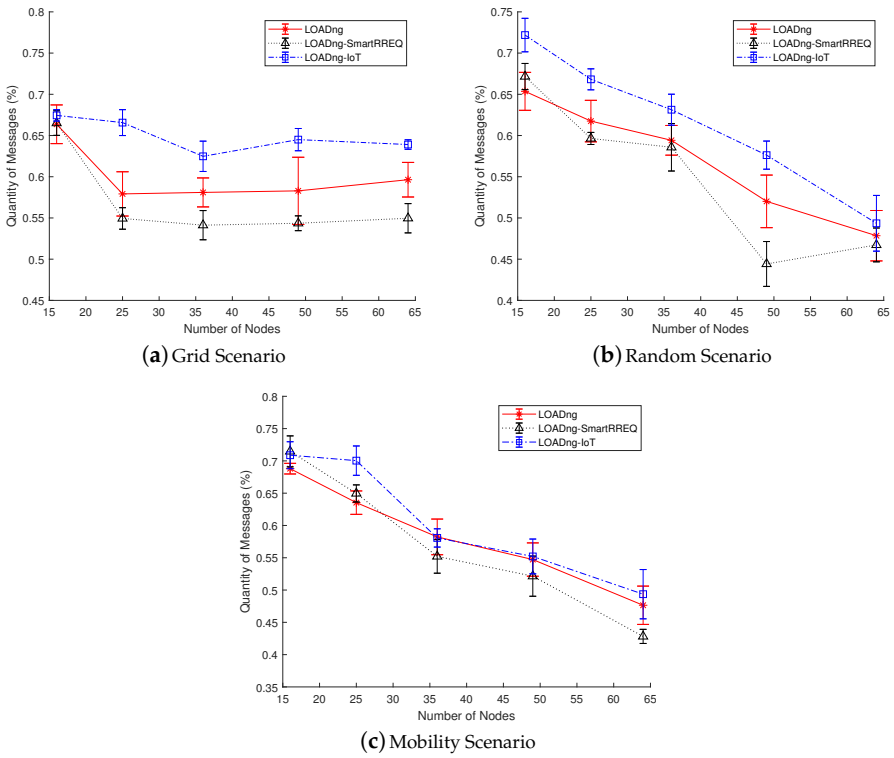
5.4. Percentage of Packets with Low Latency

The metric of the percentage of packets with low latency exposes the percentage of data packets delivered with a latency that is considered low and acceptable for low-power devices in IoT applications. Several aspects can affect the metric results; among these, it is possible to cite: the length of the path constructed during the route discovery process; the quality of links among the nodes; the radio duty cycle frequency; the medium access control protocol; the density of nodes, etc. The PLL metric is computed according to Equation (5):

$$PLL = \frac{\sum_{i=1}^{N} lat(Pr_i) < L_{th}}{\sum_{i=1}^{N} Pr_i} \quad (5)$$

where N is the number of nodes in the network, Pr_i is the number of packets received by each node i, $lat()$ is a function that returns the latency of each packet received Pr, and L_{th} is the latency threshold of a data packet be considered delivered with low latency. In this work, the L_{th} was defined in 500 milliseconds.

Figure 9 shows the results obtained for the metric of PLL. According to the results, the proposed LOADng-IoT was able to overcome the other compared solutions in the majority of studied network configurations. In the grid sparse network, except for the network with 16 nodes, the studied approaches were able to maintain almost constant results. Thus, LOADng-IoT delivered approximately 65% of the data packets with a latency lower than 0.5 s. This result outperformed the default LOADng by ∼8% and LOADng-SmartRREQ by ∼ 14%. With the use of LOADng-IoT, the nodes do not need to use a predefined gateway to forward Internet messages. Thus, the process of Internet route discovery was able to find closer nodes, constructing shorter paths and reducing the number of hops required to send an Internet message. In the dense random scenarios, the proposed LOADng-IoT was also superior to the other studied proposals. In the mobility scenario, the performance of all studied approaches was almost the same, except for the network with 25 nodes, where LOADng-IoT presented slight better results. However, it was perceived that the performance of all approaches decreased with the increase in network density. This behavior is already expected, considering that the latency tends to be increased when several nodes try to use the same frequency spectrum in a common region. The MAC protocols tend to spend more time sending the messages due to the high number of devices accessing the wireless medium [30].



(**a**) Grid Scenario (**b**) Random Scenario

(**c**) Mobility Scenario

Figure 9. Percentage of packets with low latency in function of number of nodes.

6. Conclusions and Future Works

This work presents a new improvement to the LOADng routing protocol in IoT scenarios when the network devices have different capacities and use different message types. The proposal was compared with the default implementation of LOADng and LOADng-SmartRREQ through simulations using COOJA. Four different metrics were studied to expose the network performance in terms of reliability, QoS, and power efficiency. For all the considered metrics, LOADng-IoT demonstrated better performance for sparse, dense, and mobile networks. These significant results were obtained due to the set of improvements provided by the proposed approach.

Unlike other approaches, the proposed LOADng-IoT does not require a previous definition of gateways and, hence, it can find the most appropriated Internet node to forward messages. This feature provides a self-adaption capacity to the network nodes and reduces the necessity of human intervention in both network deployment and execution. Optionally, LOADng-IoT allows the use of a cache system dedicated to storing Internet routes, enabling the nodes to direct the route discovery to INs that anteriorly had been used as gateways. These approaches together permits LOADng-IoT to reduce the control message overhead required to find the Internet routes, contributing to the reduction of power consumption and improving the packet delivery ratio. Finally, this work proposed a new error code that makes it possible for the Internet nodes to advise the other network nodes about its temporary Internet connection loss. Thus, devices that intend to send Internet messages can find new gateways to forward messages, increasing the chances of successful delivery. It was also noted that LOADng-IoT performance was, in general, less variable when compared with the other studied proposals. This behavior is justified because LOADng-IoT requires fewer radio transmissions to perform the data packet delivery and route discovery. Thus, LOADng-IoT is less affected by the interferences and collision that commonly makes the network performance unstable. In conclusion, the authors deduce that, together, all of the proposed mechanisms that comprise LOADng-IoT provide

a new significant enhancement for IoT networks, allowing them to attain better QoS, efficiency, and reliability.

For future work, the authors suggest that experiments in real IoT environments should be conducted to test the results obtained using computational simulation. Moreover, the source code from the proposed solution should be improved, documented, and disseminated to the scientific community.

Author Contributions: J.V.V.S. collected and performed the in-depth analysis and reviewed the related literature on the topic, wrote the first draft of the document, proposed the new approach in cooperation with the second author, performed the comparison study, and identified several open research issues. J.J.P.C.R. supervised the study, consolidated the comparison analysis and open issues, and reviewed the structure and the first draft. All other authors reviewed the text carefully, verified the comparison study, and reviewed the identified open issues. All authors contributed equally to the scope definition, motivation, and focus of the paper.

Acknowledgments: This work has been supported by the Brazilian National Council for Research and Development (CNPq) via Grant Nos. 201155/2015-0 and 309335/2017-5; by National Funding from the FCT—Fundação para a Ciência e a Tecnologia through the UID/EEA/500008/2013 Project; and by Finep, with resources from Funttel, grant no. 01.14.0231.00, under the Radiocommunication Reference Center (Centro de Referência em Radiocomunicações — CRR) project of the National Institute of Telecommunications (Instituto Nacional de Telecomunicações — Inatel), Brazil. The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group no. RG-1439–022.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Musaddiq, A.; Zikria, Y.B.; Hahm, O.; Yu, H.; Bashir, A.K.; Kim, S.W. A Survey on Resource Management in IoT Operating Systems. *IEEE Access* **2018**, *6*, 8459–8482. [CrossRef]
2. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
3. Hui, T.K.; Sherratt, R.S.; Sánchez, D.D. Major requirements for building Smart Homes in Smart Cities based on Internet of Things technologies. *Future Gener. Comput. Syst.* **2017**, *76*, 358–369. [CrossRef]
4. Wollschlaeger, M.; Sauter, T.; Jasperneite, J. The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0. *IEEE Ind. Electron. Mag.* **2017**, *11*, 17–27. [CrossRef]
5. Kim, H.; Ko, J.; Bahk, S. Smarter Markets for Smarter Life: Applications, Challenges, and Deployment Experiences. *IEEE Commun. Mag.* **2017**, *55*, 34–41. [CrossRef]
6. Catarinucci, L.; de Donno, D.; Mainetti, L.; Palano, L.; Patrono, L.; Stefanizzi, M.L.; Tarricone, L. An IoT-Aware Architecture for Smart Healthcare Systems. *IEEE Internet Things J.* **2015**, *2*, 515–526. [CrossRef]
7. Díaz, M.; Martín, C.; Rubio, B. State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing. *J. Netw. Comput. Appl.* **2016**, *67*, 99–117. [CrossRef]
8. Tripathi, J.; De Oliveira, J.C.; Vasseur, J.P. Proactive versus reactive routing in low power and lossy networks: Performance analysis and scalability improvements. *Ad Hoc Netw.* **2014**, *23*, 121–144. [CrossRef]
9. Alexander, R.; Brandt, A.; Vasseur, J.; Hui, J.; Pister, K.; Thubert, P.; Levis, P.; Struik, R.; Kelsey, R.; Winter, T. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*; Internet Engineering Task Force (IETF): Fremont, CA, USA; 2012.
10. Clausen, T.; Yi, J.; Niktash, A.; Igarashi, Y.; Satoh, H.; Herberg, U.; Lavenu, C.; Lys, T.; Dean, J. *The Lightweight on-Demand ad hoc Distance-Vector Routing pRotocol-Next Generation (LOADng)*; Internet-Draft draft-clausen-lln-loadng-15.txt; IETF Secretariat: Fremont, CA, USA; 2016.
11. Clausen, T.; Yi, J.; Lavenu, C.; Lys, A.; Niktash, A.; Igarashi, Y.; Satoh, H. *The LLN On-Demand Ad Hoc Distance-Vector Routing Protocol-Next Generation (LOADng)*; Internet-Draft draft-clausen-lln-loadng-00.txt; IETF Secretariat: Fremont, CA, USA; 2011.
12. Yi, J.; Clausen, T.; Igarashi, Y. Evaluation of routing protocol for low power and Lossy Networks: LOADng and RPL. In Proceedings of the 2013 IEEE Conference on Wireless Sensor (ICWISE), Kuching, Malaysia, 2–4 December 2013; pp. 19–24.
13. Sobral, J.V.V.; Rodrigues, J.J.P.C.; Saleem, K.; Al-Muhtadi, J. Performance evaluation of LOADng routing protocol in IoT P2P and MP2P applications. In Proceedings of the 2016 International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, Croatia, 13–15 July 2016; pp. 1–6.

14. Yi, J.; Clausen, T. Collection Tree Extension of Reactive Routing Protocol for Low-Power and Lossy Networks. *Int. J. Distrib. Sens. Netw.* **2014**, *10*, 352421. [CrossRef]
15. Sasidharan, D.; Jacob, L. Improving network lifetime and reliability for machine type communications based on LOADng routing protocol. *Ad Hoc Netw.* **2018**, *73*, 27–39. [CrossRef]
16. Sobral, J.V.; Rodrigues, J.J.; Kumar, N.; Zhu, C.; Ahmad, R.W. Performance Evaluation of Routing Metrics in the LOADng Routing Protocol. *J. Commun. Softw. Syst.* **2017**, *13*, 87–95. [CrossRef]
17. Sobral, J.V.V.; Rodrigues, J.J.P.C.; Neto, A. Performance Assessment of the LOADng Routing Protocol in Smart City Scenarios. In Proceedings of the 2017 IEEE First Summer School on Smart Cities (S3C), Natal, Brazil, 6–11 August 2017; pp. 49–54.
18. Hossain, A.K.M.M.; Sreenan, C.J.; Alberola, R.D.P. Neighbour-Disjoint Multipath for Low-Power and Lossy Networks. *ACM Trans. Sen. Netw.* **2016**, *12*, 1–25. [CrossRef]
19. Araújo, H.d.S.; Filho, R.H.; Rodrigues, J.J.P.C.; Rabelo, R.d.A.L.; Sousa, N.d.C.; Filho, J.C.C.L.S.; Sobral, J.V.V. A Proposal for IoT Dynamic Routes Selection Based on Contextual Information. *Sensors* **2018**, *18*, 353. [CrossRef] [PubMed]
20. Machado, K.; Rosário, D.; Cerqueira, E.; Loureiro, A.A.; Neto, A.; de Souza, J.N. A routing protocol based on energy and link quality for internet of things applications. *Sensors* **2013**, *13*, 1942–1964. [CrossRef] [PubMed]
21. Aloi, G.; Caliciuri, G.; Fortino, G.; Gravina, R.; Pace, P.; Russo, W.; Savaglio, C. Enabling IoT interoperability through opportunistic smartphone-based mobile gateways. *J. Netw. Comput. Appl.* **2017**, *81*, 74 – 84. [CrossRef]
22. Javaid, U.; Rasheed, T.; Meddour, D.; Ahmed, T. Adaptive Distributed Gateway Discovery in Hybrid Wireless Networks. In Proceedings of the 2008 IEEE Wireless Communications and Networking Conference, Las Vegas, NV, USA, 31 March–3 April 2008; pp. 2735–2740.
23. Singh, D.; Kim, D. Performance Analysis of Gateway Discovery Techniques: IPv6-Based Wireless Sensor Networks. In Proceedings of the 2010 2nd International Conference on Evolving Internet, Valencia, Spain, 20–25 September 2010; pp. 142–146.
24. Das, S.R.; Perkins, C.E.; Belding-Royer, E.M. *Ad hoc On-Demand Distance Vector (AODV) Routing*; Internet Engineering Task Force (IETF): Fremont, CA, USA, 2003.
25. Clausen, T.; Yi, J.; Herberg, U. Lightweight On-demand Ad hoc Distance-vector Routing - Next Generation (LOADng): Protocol, extension, and applicability. *Comput. Netw.* **2017**, *126*, 125–140. [CrossRef]
26. Yi, J.; Clausen, T.; Bas, A. Smart Route Request for on-demand route discovery in constrained environments. In Proceedings of the 2012 IEEE International Conference on Wireless Information Technology and Systems (ICWITS), Maui, HI, USA, 11–16 November 2012; pp. 1–4.
27. Dunkels, A.; Gronvall, B.; Voigt, T. Contiki-a lightweight and flexible operating system for tiny networked sensors. In Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, Tampa, FL, USA, 16–18 November 2004; pp. 455–462.
28. Bettstetter, C.; Santi, P.; Resta, G. The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks. *IEEE Trans. Mob. Comput.* **2003**, *2*, 257–269. [CrossRef]
29. Aschenbruck, N.; Ernst, R.; Gerhards-Padilla, E.; Schwamborn, M. BonnMotion: A Mobility Scenario Generation and Analysis Tool. In Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, Malaga, Spain, 16–18 March 2010; ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering): Brussels, Belgium, 2010; pp. 1–10.
30. Kim, Y.; Shin, H.; Cha, H. Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks. In Proceedings of the 2008 International Conference on Information Processing in Sensor Networks (IPSN 2008), St. Louis, MO, USA, 22–24 April 2008; pp. 53–63.

Chapter 5

Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios

This chapter consists in the following paper:

Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios

José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Kashif Saleem, and Sergei Kozlov

IEEE Access, IEEE, ISSN: 2169-3536, 2019.

DOI: doi.org/10.1109/ACCESS.2019.2932718

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 3.745

ISI Article Influence Score (2019): 0.642

Journal Ranking (2019): 35/156 (Computer Science, Information Systems)

Journal Ranking (2019): 26/90 (Telecommunications)

Journal Ranking (2019): 61/266 (Engineering, Electrical & Electronic)

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON MOBILE EDGE COMPUTING AND MOBILE CLOUD COMPUTING: ADDRESSING HETEROGENEITY AND ENERGY ISSUES OF COMPUTE AND NETWORK RESOURCES

Received July 15, 2019, accepted July 28, 2019, date of publication August 2, 2019, date of current version August 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2932718

Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios

JOSÉ V. V. SOBRAL[1,2], JOEL JOSÉ P. C. RODRIGUES[1,3,5,6], (Senior Member, IEEE), RICARDO A. L. RABÊLO[4], KASHIF SALEEM[5], AND SERGEI A. KOZLOV[6]

[1]Instituto de Telecomunicações, Universidade da Beira Interior, 6201-001 Covilhã, Portugal
[2]Department of Education, Federal Institute of Maranhão (IFMA), São Luís 65010-030, Brazil
[3]PPGEE, Federal University of Piauí (UFPI), Teresina 64049-550, Brazil
[4]PPGCC, Federal University of Piauí (UFPI), Teresina 64049-550, Brazil
[5]Center of Excellence in Information Assurance (CoEIA), King Saud University, Riyadh 11653, Saudi Arabia
[6]International Institute of Photonics and Optoinformatics, ITMO University, 197101 St. Petersburg, Russia

Corresponding authors: Joel José P. C. Rodrigues (joeljr@ieee.org) and Kashif Saleem (ksaleem@ksu.edu.sa)

This work was supported in part by the Brazilian National Council for Research and Development (CNPq) under Grant 201155/2015-0 and Grant 309335/2017-5, in part by the National Funding from the FCT—*Fundação para a Ciência e a Tecnologia* under Project UID/EEA/50008/2019, in part by the Government of the Russian Federation under Grant 08-08, and in part by the Deanship of Scientific Research at King Saud University through the Research Group under Grant RG-1439-022.

ABSTRACT Routing protocols represent an important issue on the Internet of Things (IoT) scenarios since they are responsible for creating paths and forwarding data packets among the network nodes. In mobile IoT scenarios, the topology changes caused by the movement of nodes makes the work of routing protocols more difficult. Thus, the current IoT routing solutions tend to present strong limitations and a poor performance in these scenarios, generally requiring complex additional improvement to better support the mobility of the devices. In this context, the Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation (LOADng), is an emerging solution for IoT networks that despite being adequate for a mobile environment due to its reactive functioning still lacks in performance. Thus, this paper proposes a novel solution to enhance the performance of LOADng in mobile IoT networks. The improved version, LOADng-IoT-Mob, introduces a mechanism that permits nodes to be aware of the availability of their neighbors through the harnessing of control messages. As a result, these nodes can shorten paths and avoid sending data packets through broken routes due to the movement of the nodes. Additionally, a short periodical control message is introduced, allowing the nodes to update their routing table, even with a low control message frequency. Furthermore, a new routing metric is proposed for creating routes based on the reliability of the link and proximity of the neighboring nodes. Finally, through computational simulations, the performance of the LOADng-IoT-Mob is studied under multiple scenarios varying the network size, the number of mobile devices, and maximum nodes' speed. The results obtained demonstrate the efficiency of the proposed solution in terms of packet delivery ratio, latency, and power and overhead efficiency, with a slight increase in memory consumption.

INDEX TERMS Internet of Things, LOADng, low power and lossy networks, routing protocol, wireless network.

I. INTRODUCTION

In recent years, Internet of Things (IoT) has grabbed the attention of both the scientific community and business sector for its great potential and research and business opportunities [1], [2]. From a simple smart temperature control system [3] all the way to a city entirely operating on smart technology [4], [5], IoT covers a vast range of applications. However, along with the opportunities of IoT emerge the technical challenges of the existing technological limitations. From these, it is possible to detach the issues related to the networks that interconnect IoT devices.

In general, IoT devices are miniaturized and suffer from severe limitations of hardware (processing and memory) and energy. In a broad set of IoT environments, network devices can have mobility capacity, which makes hardest the network functioning [6]. Thus, due to these characteristics, IoT networks are categorized as Low power and Lossy

The associate editor coordinating the review of this manuscript and approving it for publication was Raja Wasim Ahmad.

Networks (LLNs). In an IoT context, routing protocols are responsible for providing routes and forwarding data packet among the connected devices [7]. Thus, regarding the characteristics of LLNs, routing protocols need to be simple to fit and perform efficiently in any IoT device with the lowest computational cost possible. In addition to device level performance, routing protocols should be able to self-adapt to the typological changes caused by the mobility of IoT devices.

The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [8], although considered as the standard routing protocol for IoT networks [9], does not efficiently support mobility because it has been particularly designed for static networks [10]. In addition, it has limitations regarding multicast data forwarding [11], [12], memory usage [13], [14], and point-to-point traffic pattern [9], [15]. Thus, as an alternative to RPL, the Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation (LOADng) [16], provides a lightweight, modular, and simple routing solution for LLNs.

LOADng is a reactive routing protocol that was initially designed for Mobile Ad-hoc Networks (MANET) and efficiently manages high point-to-point (P2P) data traffic. However, its main limitations, as mentioned in [17], are delay in constructing routes and problems provoked with inefficient flooding consuming high energy, with massive overhead and packet collision.

Although designed for networks composed by mobile nodes, to the best of authors' knowledge, the current literature does not address any comprehensive study to assess the performance of LOADng in IoT networks with mobile devices. The low performance of existing IoT routing solutions in mobile scenarios makes emerging the need for less complex improvements to fulfill the requirements of these networks. Therefore, this work aims to study the performance of LOADng in mobile IoT scenarios and additionally propose a novel mechanism to improve protocol performance. The proposed LOADng-IoT-Mob introduces a new mechanism that allows the nodes to harness control messages to help the routing table management and adapt to the dynamic changes in the topology. This mechanism also allows the nodes to shorten paths to destinations that have come closer. Further, the LOADng-IoT-Mob uses a new short periodical control message that can be used when the regular control message exchange is not enough to maintain the routing table of the nodes updated. These periodic messages are controlled and can be suppressed to avoid an increase in the network overhead. Moreover, the proposed solution includes an additional routing metric that seeks to build reliable routes composed of the nearest neighboring nodes. Thus, the main contributions of this work are as follows:

- LOADng routing protocol performance analysis and highlight key issues in supporting mobile nodes in IoT networks;
- Propose a novel solution known as LOADng-IoT-Mob based on the harnessing of control messages to improve the performance of LOADng. This enhanced control messaging supports mobility in IoT scenarios and keeps track of the changes in the location of its mobile neighbors.
- Creating a new periodic control message to trigger the control message harnessing mechanism and permit the routing table management without significantly increasing the message overhead.
- Introducing a new routing metric in the LOADng-IoT-Mob based on the received signal strength, which increases network reliability and the Quality of Service (QoS).
- Last but not least, the proposed solution's network performance is studied for memory consumption, packet delivery ratio, latency, and power and overhead efficiency.

TABLE 1. List of acronyms and abbreviations.

Index	Meaning
AMI	Advanced metering infrastructure
AODV	Ad-hoc On-Demand Distance Vector
CCR	Check channel rate
COB	Control bit overhead per delivered data bit
CSMA	Carrier Sense Multiple Access
CTP	Collection Tree Protocol
DEST	Destination
ESB	Energy spent per delivered data bit
ExpRing	Expanding Ring
IC	Internet-connected
IoT	Internet of Things
kB	Kilobytes
LLN	Low power and Lossy Networks
LOADng	Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation
LR	Live routes
LTE	Long Term Evolution
MAC	Medium access control
MANET	Mobile Ad-hoc Networks
mJ	Millijoules
MNB	Maximum number of broadcasts
Mob	Mobile
MP2P	Multipoint-to-point
OS	Operating system
P2P	Point-to-point
PDR	Packet delivery ratio
PLL	Packets delivered with low latency
Pos	Position
QoS	Quality of Service
RAM	Random access memory
RDC	Radio duty cycle
RE	Residual energy
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks
RREP	Route Reply
RREQ	Route Request
RSSI	Received signal strength indicator
SRC	Source
UDGM	Unit Disk Graph Model

Table 1 presents the list of acronyms and abbreviations adopted in this work. The rest of the paper is organized as follows. Section 2 discusses the LOADng routing protocol and details the mobile IoT network model while

Section 3 reviews the literature and relevant improvements for LOADng. Section 4 presents the proposed LOADng-IoT-Mob with a novel route discovery mechanism, control message harnessing function, and routing metric and the performance assessment of the proposed solution is analyzed in Section 5. Finally, Section 6 concludes the paper and suggests some future recommendations.

II. BACKGROUND AND SYSTEM MODEL

This Section discusses the LOADng routing protocol and the model of mobile IoT network considered on this work.

A. LOADng ROUTING PROTOCOL

LOADng is a routing protocol that simplifies the AODV (Ad-hoc On-Demand Distance Vector) [18] to attain and efficiently handle the requirements of LLNs. Its reactive feature constructs a route among the nodes only when data packets need to be transferred. Thus, a source (SRC) node that needs to transmit a data packet needs to commence a route discovery process to find and construct a path till the destination (DEST) node. This process is performed using a core structure comprising control messages and an Information Base. The main elements for route discovery are the Route Request (RREQ) and Route Reply (RREP) message and the Routing Set (also termed as routing table), which can comprise several route entries [19]. While control messages are exchanged among the nodes to discover and construct paths, the Routing Set is used to store the information about the created routes, handle control message flooding, and provide information to the forwarding of data packets.

Simply stating, in the LOADng protocol, when a node wants to send a data packet to a destination and the path is unknown, it should store the data packet in a queue and start a route discovery process. Thus, the packet originator should create and broadcast an RREQ addressed to the packet destination indicating the necessity of route creation. Every node that receives the RREQ should process the message, create or update an entry in the Routing Set to the message originator, and verify if it is the message destination before deciding to drop or rebroadcast the route request. An RREQ is dropped mainly when its hop limit expires, or when the node has already received the same message. The route request is rebroadcasted when not dropped.

When RREQ reaches its destination, the receiver node generates an RREP message as a feedback to the request source address. The RREP is then unicasted through the same path traveled by the RREQ. On the way, every intermediate node that receives the message verifies and update its Routing Set before forwarding the reply message to the destination. When the RREP reaches its RREQ source, the route discovery process is completed, and the data packets are then unicast over the recorded path. Figure 1 exemplifies the route discovery process of LOADng, where node SRC needs to send a data packet to the DEST node. LOADng presents considerable simplifications concerning AODV, one of the most relevant being the non-use of periodical HELLO messages.

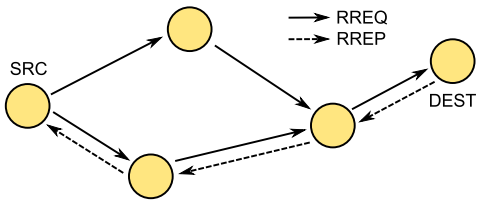


FIGURE 1. LOADng route discovery process. SRC creates and broadcasts an RREQ that floods the network to the point of reaching the destination node DEST. DEST replies the request of SRC using an RREP that is unicasted through the path constructed by RREQ.

This simplification helps reduce the network traffic overhead and, consequently, reduces the energy consumed. However, it also implies a lesser exchange of control information, which is crucial in mobile networks.

B. MOBILE IoT NETWORK AND APPLICATION MODEL

This work seeks to study and improve LOADng specifically for IoT networks with mobile devices. Therefore, in this paper, the mobile IoT network model considered comprises devices with movement and variation in the availability of Internet connection. The network consists of Internet-connected and simple nodes. The Internet-connected (IC) nodes are equipped with IEEE 802.15.4 radio and an additional communication interface (*e.g.* LTE and Wi-Fi) that allows the nodes to communicate directly with the Internet. In contrast, simple nodes are equipped with an IEEE 802.15.4 radio but do not have direct Internet-connection, making it necessary to use an IC node as a gateway to forward the messages addressed to the Internet. Both types of nodes can be mobile.

In the considered IoT scenario, the nodes can send two types of data packets: Internet packets and simple packets. The Internet packets are addressed to an external Internet destination or service located outside the local network. Consequently, these Internet packets can generate multipoint-to-point (MP2P) traffic once sent from simple nodes to IC nodes that should forward them to the Internet. Contrarily, the simple packets are destined for any local network nodes in a P2P traffic pattern. All network nodes, Internet-connected or not, can create, send and forward both packet types.

Figure 2 presents an IoT smart home scenario with mobile nodes that can encompass the previously described network and application. In the scenario, devices can communicate amongst themselves to reach a common objective. For example, an air conditioner can use a laptop as a gateway for sending an Internet packet to consult the weather condition during daytime. A smart TV, an IC device, can directly access the streaming services and at the same time provide a connection to a smart refrigerator to consult the prices of and order groceries that are out of stock. Smartphones can also be used to activate/deactivate a robotic vacuum cleaner both locally or through the Internet.

The IoT network and application model presented here have numerous applications. Also, the given model can

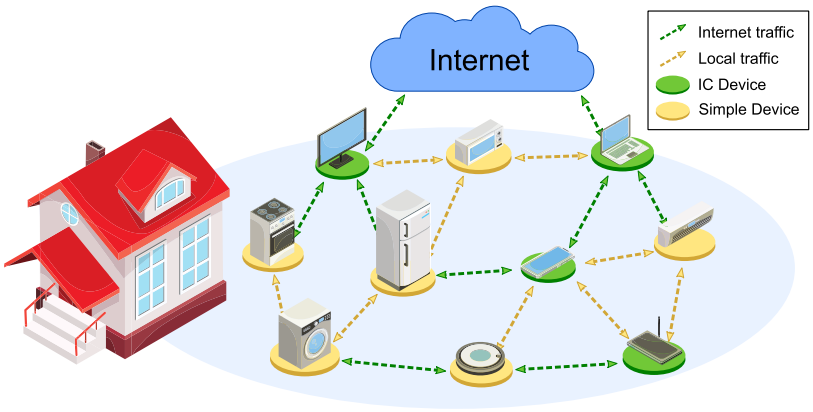


FIGURE 2. Smart home IoT scenario composed by devices with different capacities. Note: This image has been designed using resources from Freepik.com.

encompass the concepts of IoT edge and fog computing, where devices exchange data in a closer way to reduce the latency and the required bandwidth to communicate with data centers [20]. Thus, it is "common sense" that due to the deployment cost, not all IoT devices will be directly connected to the Internet. Hence, an efficient solution is required to help these resource-limited devices connect and communicate with the Internet. Further, considering the mobility of devices, the adopted routing solution should self-adapt to autonomously maintain its functioning with minimum human intervention and fulfill the application's requirements in an optimal manner.

III. RELATED WORKS

LOADng presents essential changes in AODV to create a more lightweight and reliable protocol to meet the requirements of LLNs. However, this protocol still has drawbacks that sharply limit its performance. Although LOADng has been most studied in recent years and regarded a promising protocol for IoT networks, mainly with P2P traffic, the proposed improvements for the protocol are still few in number.

A performance comparison between LOADng and RPL is presented in [21]. In their work, the authors have assessed the performance of both the protocols regarding packet delivery ratio, control overhead, average path length, and end-to-end delay. The obtained results showed that LOADng overcomes RPL mainly in P2P scenarios, where the proactive solution has recognized lower performance. The performance of LOADng was also studied in IoT scenarios with P2P and MP2P traffic in [22]. Based on the limitations of LOADng in MP2P scenarios, Yi and Clausen [23] have proposed a collection tree extension for the protocol. This proposal, named LOADng-CTP, introduces proactive features to LOADng and helps the nodes create a routing tree to forward data packets from the leaves (nodes) to a root (sink) node. The proposed approach has performed similar to RPL regarding packet delivery ratio and delay, but with a lower control overhead. The performances of RPL and LOADng-CTP were also compared in an advanced metering infrastructure (AMI) scenario in [24].

The default LOADng proposal is modular and permits the use of different routing metrics to construct paths among the nodes. The native routing metric of LOADng is the hop count, which is not reliable and takes into account the number of hops between the nodes only once to select the best route. Thus, Sasidharan and Jacob [25] proposed LRRE, a new routing metric based on the number of live routes (LR), residual energy (RE), and hop count. Here, the referred metrics are merged in addition to hold the monotonicity and isotonicity properties of routing algebra [26]. The evaluation results exhibited that the proposed LRRE obtained better results when compared with each one of the metrics used separately. The performance of LOADng with different routing metrics was also studied in [27]. LOADng, due to its reactive features, produces a considerable control message overhead during its functioning. To mitigate this problem, Yi *et al.* [28] have proposed the SmartRREQ mechanism to limit the broadcasting of messages during the route discovery process. This solution uses the already existent information about the routes in the intermediate nodes to control the number of RREQ broadcasts. Thus, when an intermediate node, which is neither the RREQ originator nor the destination, receives an RREQ, it should perform additional processing to verify if there is any entry in its Routing Set to the RREQ destination. If an entry is found, then the RREQ message should be forwarded in unicast to the destination using the next hop node stored in the route entry or else, the message will be normally forwarded in broadcast as the default implementation of LOADng. The obtained results compared with default LOADng and AODV show this simple mechanism can reduce the number of broadcast transmissions, contributing to the decrement of packet collisions and traffic overhead.

Still seeking to reduce the control traffic overhead of LOADng, Bas *et al.* [29] introduced the Expanding Ring flooding extension for LOADng to perform the discovery process using flooding of control message based on rings that

can grow progressively. Thus, the Expanding Ring creates a new field on the RREQ messages known as the maximum number of broadcasts (MNB) (or `mnb-value`) to control the range of RREQs in the number of hops. Also, the authors introduced three new parameters to adjust the mechanism's functioning: MNB_START is used to define the initial `mnb-value`; MNB_INCREMENT is used to define the increase in range after a route discovery fault; and MNB_THRESHOLD is used to define the maximum value of MNB. When the process of route discovery is initiated, the `mnb-value` field of the RREQ is initialized with the MNB_START value and, then, the message is sent in a broadcast. The RREQ originator should also define a timer to wait for a reply to the generated request. Each intermediate node that receives the RREQ should verify the existence of some entry for the message destination into its Routing Set. If found to be true, the node should use the SmartRREQ mechanism to transmit the message to the destination, otherwise the node will verify if the `mnb-value` field is higher than 0. If found positive, the `mnb-value` is decreased by one, and the message is broadcasted. In contrast, the received RREQ is dropped. When the timer defined by the RREQ originator expires, the originator starts a new route discovery using the `mnb-value` field added to the MNB_INCREMENT value. This process occurs until the MNB reaching the MNB_THRESHOLD value or the desired route is constructed. The Expanding Ring mechanism can regulate the area of the RREQ messages and reduce the control overhead. However, if the message destination is not found in the initial discovery tentative, the mechanism can produce a reverse effect, increasing the network overhead and energy consumption. Compared with SmartRREQ, Expanding Ring can reduce the number of collisions, but this leads to an increase in the end-to-end delay.

Considering IoT scenarios with different types of nodes and traffic patterns similar to that described in the subsection II-B, Sobral *et al.* [30] proposed the LOADng-IoT. The proposed improvement for LOADng in IoT scenarios comprises three different mechanisms that can enhance the network performance regarding QoS and energy efficiency. The introduced Internet route discovery mechanism allows the network nodes to find a route to IC devices without a previous definition of a gateway. For this purpose, both RREQ and RREP messages received an additional flag known as `iot` to indicate that messages are used to discover an IC node. RREQ messages with `iot` flag true does not have a defined destination, and any device Internet-connected can reply to it using an RREP message with IoT flag true. Further, the authors have introduced a new Internet Route Cache mechanism that allows the nodes to store the main information about a previously known Internet route removed from the Routing Set. This feature enables the nodes to reduce the number of broadcasts required to find an Internet path and contributes to a lower energy consumption and control overhead, although there is a slight increase in the memory usage. Finally, the authors also proposed a new error code to advise the network nodes about the status of the Internet connection of the devices working as gateways.

Although LOADng-IoT has presented significant improvements to the use of LOADng in IoT scenarios, it still has some limitations in mobility scenarios since it has been projected for static networks. The LOADng-IoT performance evaluation study presented by the authors showed that the proposal overcame the compared solutions in several situations, including one with mobile nodes. However, in the mobility scenario, only a small number of Internet-connected nodes managed to move across the studied area. Furthermore, the authors have not addressed any solution to mitigate the decrease in performance due to the dynamicity in the network topology, which occurs with the movement of the nodes.

The limitation presented by the current literature solutions, mainly related to the lack of mobility support, have motivated this work. The proposed solution to improve the performance of LOADng in mobile IoT networks has been thoroughly described in the following section.

IV. PROPOSED LOADng-IoT-Mob

This work proposes an improvement for LOADng protocol in mobility IoT scenarios. The proposed approach, LOADng-IoT-Mob, introduces a new mechanism that allows nodes harnessing the control messages, exchanged during the route discovery process, to manage the information of its Routing Set to better support the topology changes due to the mobility of devices. Further, it allows the nodes to shorten paths to a known destination that can grow closer because of its movement. Moreover, LOADng-IoT-Mob also includes a new routing metric based on the received signal strength indicator (RSSI) to help the nodes construct paths with more close and reliable nodes.

In addition to mobility support, LOADng-IoT-Mob includes features of SmartRREQ, Expanding Ring, and LOADng-IoT to provide an improved and efficient network performance, specifically for IoT applications. Thus, some fields are added into default LOADng control messages and Routing Set. In both RREQ and RREP, the fields `smart-rreq`, `mnb-value`, and `iot` are inserted to support SmartRREQ, Expanding Ring, and LOADng-IoT proposals. Further, in the Routing Set, the field `R_internet_conn` is inserted to support the LOADng-IoT. Finally, the novel mechanism proposed by LOADng-IoT-Mob requires the addition of a new fields, namely `R_next_hop_valid_time`, in the Routing Set to record the valid time of each next hop node.

The following subsections describe the proposed approach functioning and indicate how both control message and Routing Set additional fields are used to improve the network performance.

A. ROUTE DISCOVERY IN LOADng-IoT-MOB

LOADng-IoT-Mob, as previously explained, encompass other improvements to offer an improved performance in IoT networks similar to the shown the subsection II-B. Thus, the

proposed solution allows the use of two different types of route discovery processes.

The first, known as simple route discovery process, is dedicated to create paths to forward the local data packets. Thus, when a node intends to send a simple packet to a destination not found on its Routing Set, it should kick off with a simple route discovery process using the Expanding Ring flooding mechanism, as shown in Figure 3. In this process of route discovery, the SRC node generates an RREQ with `mnb-value` field equal to one and broadcasts it. If the node that receives the RREQ contains a route to the desired DEST, it unicasts the RREQ by means of the SmartRREQ mechanism to the DEST. The DEST node receives the RREQ and replies it with an RREP message. After receiving the RREP, the SRC node completes the discovery process by updating the Routing Set with new information and can start sending the data packets to the DEST node. Note that other neighbor nodes that receive the RREQ and do not have a route to DEST drop the RREQ packet due to the MNB limitation.

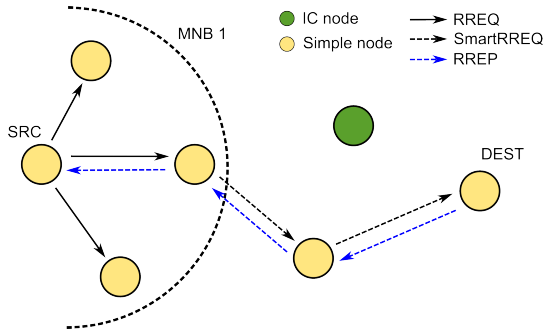


FIGURE 3. LOADng-IoT-Mob simple route discovery process using expanding ring flooding for constructing a path for a simple node.

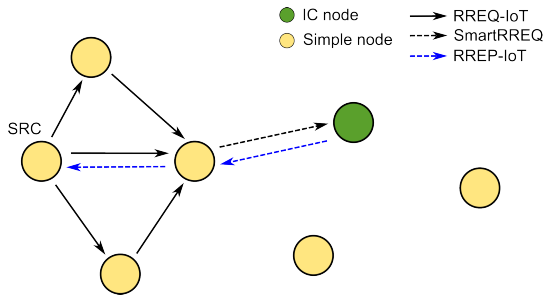


FIGURE 4. LOADng-IoT-Mob Internet route discovery process for constructing a path for an Internet-connected node.

The second process, termed Internet route discovery, is used when a node needs to send a data packet to an Internet address. Hence, the node performs an Internet route discovery process, as proposed by the LOADng-IoT and exemplified in Figure 4. At the time, when the SRC node needs to send a data packet to an Internet address, the node generates and broadcasts an RREQ-IoT (RREQ with `iot` flag = 1) in the network. Note that as the interest of SRC is to find an IC node to forward the message to the Internet, RREQ-IoT does not indicate any specific destination inside the local network. The neighbor node receiving the RREQ-IoT and knowing a path to an IC node transmits the request in unicast similar to the SmartRREQ mechanism. The IC node receiving the request generates an RREP-IoT (RREP with `iot` flag = 1) and sends it to the SRC node. Finally, the SRC, after receiving the RREP-IoT, sends the Internet data packets through the found IC node, using it as a gateway. It is important to highlight that in the Internet route discovery, flooding is not limited by the MNB value because the Expanding Ring mechanism is not used.

Note that the LOADng-IoT-Mob does not create new route discovery mechanisms but has already merged existing efficient solutions. However, in order to fulfill its aim and allow support mobility, the proposed solution introduces an additional field for each entry in the Routing Set to manage the availability of the next hop node in the path to each destination. Thus, during both the route discovery processes, whether simple or Internet, each new entry added in the Routing Set should include the `R_next_hop_valid_time` field. The value of the new field value is set according to the value of NEXT_HOP_VALID_TIME parameter, which is represented in seconds, plus one hysteresis value.

The NEXT_HOP_VALID_TIME parameter, which is better explained in the next subsection, should initially be defined in the network deployment phase and accessible to the whole network's nodes. The following subsection also describes the use of control messages to manage the availability of routes and the refreshing procedure of `R_next_hop_valid_time` values.

B. CONTROL MESSAGES HARNESSING FOR ROUTES MANAGEMENT

LOADng in scenarios with mobile nodes suffers from topology changes during data forwarding. A route discovered in an instant can become unavailable a few seconds after its creation. The simplification that led the AODV toward LOADng has removed the use of periodical HELLO messages to reduce the control overhead and the protocol complexity. Contrarily, this change makes the protocol lose its capacity of self-adaption to topology changes, suffering decrease in notable performance in high mobility scenarios.

To confront the presented challenge and provide adequate mobility support for mobile IoT networks, LOADng-IoT-Mob proposes a mechanism that can harness all control messages used during the network functioning for managing the Routing Set. Thus, during the route discovery processes (both simple and Internet), after receiving a control message, the nodes should perform an initial procedure, before handling the message content, to refresh the `R_next_hop_valid_time` field of the Routing Set related to the message sender. The `R_next_hop_valid_time` is used to indicate the valid time of the next hop of a route. Thus, while this valid time does not expire, the node considers the next hop to be on its communication range. In other words,

Algorithm 1 Algorithm for Next Hop Valid Time Refresh and Path Shorten Executed for Each Received Control Message

```
Input: Received control message
msg ← received control message;
previous_hop ← address of node from the message was received;
for each entry of Routing Set do
    if entry.R_next_addr = previous_hop then
        entry.R_next_hop_valid_time ← NEXT_HOP_VALID_TIME + 1;
    end
    if entry.R_dest_addr = previous_hop and entry.R_next_addr ≠ entry.R_dest_addr then
        entry.R_next_addr ← previous_hop;
        entry.R_next_hop_valid_time ← NEXT_HOP_VALID_TIME + 1;
        entry.R_hop_count ← 1;
        entry.R_metric_type ← HOP_COUNT;
        entry.R_metric ← MAX_DIST;
    end
end
```

R_next_hop_valid_time can indicate the presence of the next hop node in the area of the route entry owner.

The R_next_hop_valid_time value is updated every second, decreasing by one. When a node receives a control message from its neighbor, the R_next_hop_valid_time value of the route entries related to the message sender are refreshed. This procedure is exposed in Algorithm 1. For each route entry existing on its Routing Set, the node should verify if the previous hop node (i.e., the message sender) is the next hop for reaching the destination of the entry. If yes, then that node should refresh the R_next_hop_valid_time according to the NEXT_HOP_VALID_TIME parameter plus one hysteresis value which, in the algorithm, is 1.

When a R_next_hop_valid_time is decreased to zero, the node should assume that the neighbor node defined as the next hop of the route entry lies outside the communication range. Thus, the route entry of the expired R_next_hop_valid_time is considered invalid and should be used neither for data forwarding or path construction. Hence, if the node to need to send a message, the invalid route should not be considered, and a new route discovery process should be performed. This mechanism, although forces the execution of new route discoveries, ensures that the data packet is not being sent through an unreachable next hop, provoking data packet loss. However, note that this invalid route should not be removed from the Routing Set but only maintained as inexistent. In this way, if the node receives a control message from the next hop of the invalidated route entry, the route can be reactivated by refreshing the entry's R_next_hop_valid_time. This behavior avoids the data packets from being forwarded for destinations with an inaccessible next hop. At the same time, it permits the node to recover route to nodes that have regressed to the communication range without the need of performing a new route discovery process.

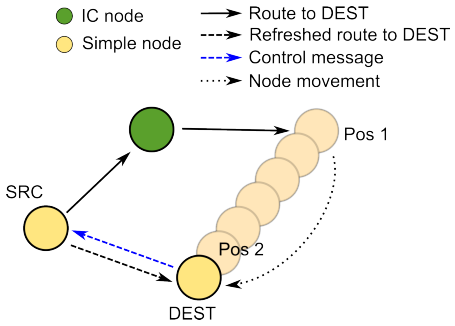


FIGURE 5. Path shorten mechanism proposed by LOADng-IoT-Mob. The SRC shortens the path constructed to the DEST node when it moves from Pos 1 to Pos 2.

Algorithm 1 also shows an important feature provided by the LOADng-IoT-Mob to shorten previously constructed paths. When a control message is received, in addition to the procedure previously described, the node should check if the previous hop node is the destination of some entry in the Routing Set. If yes, then the node should update the route entry information, shortening the path to just one hop, and updating some other data of the entry to maintain it coherence. Figure 5 exemplifies this procedure. Initially, when the route is created, the DEST node is in Pos 1, and the path from SRC to DEST have two hops. Seconds after, DEST moves to Pos 2, which is the nearest to the SRC node. At this point, the DEST node broadcasts a control message (*e.g.*, discovering a route for another node outside the figure). The SRC node receives the control message from DEST and, then, executes the procedure described in Algorithm 1. The SRC, verifying the existence of a route entry to the DEST on its Routing Set, refreshes and shortens the path. Thus, a future message from the SRC to the DEST can be sent using the new and most suitable one hop path.

The use and frequency of control messages during network functioning can vary according to several parameters, such as the network application, data traffic, and quantity of nodes. Thus, to allow the nodes to maintain its Routing Set refreshed even in a network with little control messages usage, LOADng-IoT-Mob has adopted periodical control messaging, where the introduced HELLO_MOB messages are simpler and shorter than the previously known HELLO messages from AODV. The proposed new control message is formed by only two fields, message type and originator address. Thus, the unique function of HELLO_MOB is to make the message receiver execute the procedure that refreshes the R_next_hop_valid_time value and shorten existent paths when possible.

HELLO_MOB messages are generated and scheduled to be sent in a fixed interval based on the HELLO_MOB_INTERVAL parameter, whose value represents seconds. To avoid the generation of excessive control message overhead, the HELLO_MOB messages sent can be suppressed when another control message is sent in broadcast within the HELLO_MOB sending interval. Thus, when a node sends a control message in broadcast, the scheduled HELLO_MOB message is delayed with an additional HELLO_MOB_INTERVAL. This practice presumes that as the neighbors have received a control message and executed the Algorithm 1 procedure a few moments ago, the reception of the HELLO_MOB only represents an unnecessary overhead. Contrarily, when the use frequency of the control message is low, the HELLO_MOB generated in a fixed interval can help the nodes refresh and maintain its Routing Sets. It should be highlighted that the HELLO_MOB suppression is only performed by the transmission of control messages in broadcast. It is because unicast transmissions, different from broadcasts, cannot be received by all nodes in the area and, thus, it is not possible to assume that all the neighbors have received the communication.

The proposed solution helps LOADng better self-adapt to the topology changes that occur with the movement of nodes. The `R_next_hop_valid_time` field introduced in the Routing Set and refreshed by harnessing control messages allows the nodes to be aware of the availability of the next hop and avoid sending data packets through broken paths. At the same time, the mechanism to shorten paths can help the nodes reduce the transmissions required for reaching a destination. It is important to highlight that the path shortening forces the route entry to use the hop count routing metric, which is the basic and default metric of LOADng. Also, it is recommended to adjust the NEXT_HOP_VALID_TIME and HELLO_MOB_INTERVAL parameters with equal values to ensure synchronicity between the valid time of the next hop and HELLO_MOB sending. Finally, the experiments performed during the proposed solution design exposed that HELLO_MOB message usage strongly decreases in dense networks with reasonable use of control message, becoming almost unused in some cases. Thus, the authors indicate that the overhead generated by the proposed periodic message is insignificant regarding the benefits of its usage.

Taking into account that default route metric of LOADng cannot be reliable for mobile IoT networks, the next subsection presents the proposed routing metric to be used with the LOADng-IoT-Mob.

C. WEAKRSSI ROUTING METRIC

The default LOADng uses the hop count as the routing metric for selecting the best route between two nodes. This metric, although simple and easily implementable, can neither represent the real distance or the quality of the links that compose a path. However, the core structure of LOADng permits the creation and use of different routing metric without significant modifications in the protocol. Thus, LOADng-IoT-Mob introduces the new weakRSSI routing metric that allows the nodes to create routes based on the distance among the nodes and the strength of the received transmissions.

The weakRSSI is a minimizable routing metric (lower values is better) based on the RSSI, which is a value computed in the reception of a transmission indicates the strength of the received signal. RSSI values vary according to the used radio interface; *e.g.*, in the CC2420, the RSSI ranges from 0 to 100 (nearest to the value zero is better) [31]. Frequently, RSSI values are used both for distance [32] and link quality [33] estimation. However, the measured RSSI values represent only the communication between two nodes. Thus, to measure the quality of a whole path, weakRSSI uses a simple aggregation mechanism that counts the "weak" RSSI values along the path. Under this mechanism, the RSSI of a link is counted as "weak" if its value is above a threshold value defined through the WEAKRSSI_THRESHOLD parameter.

In the course of the network functioning with the use of weakRSSI, the nodes should define the metric in the generation of each RREQ and RREP using the `metric-type` field, which already exists in the message. Thus, during the route discovery processes, each node that receives an RREQ or RREP should calculate the RSSI value for the received message and verify if it is lower than the defined threshold. If true, the RSSI value is considered "weak" and, the `route-metric` field, which already exists in both the messages, should be incremented by one. In contrast, the field is not modified. In the following, the message is normally handled, the route is created or updated, in the case of an already existent path having a higher weakRSSI value. Figure 6 helps exemplify the use of weakRSSI routing metric. Considering the weakRSSI threshold as −30, all RREQ or RREP received within an RSSI lower are computed as "weak" and incremented in the `route-metric` message field. Thus, the route constructed by DEST to reach SRC is the DEST-D-C-SRC once the weakRSSI value is 1.

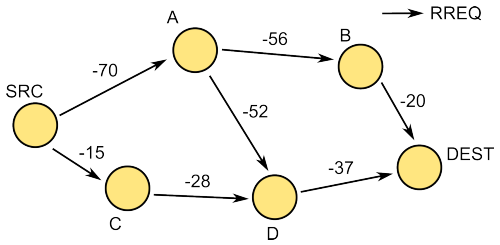


FIGURE 6. Best path selection using weakRSSI routing metric. If the WEAKRSSI_THRESHOLD is defined as −30, then the selected path from DEST to SRC should be DEST-D-C-SRC.

Note that neither LOADng or LOADng-IoT-Mob are multipath and, thus, the Routing Set only stores one entry for each destination. Hence, when forwarding data packets, the nodes do not need to choose the best path because this task is already done in the moment of route discovery, and only one path should exist.

The simple but efficient proposal of weakRSSI allows the nodes to choose the best route to a destination based on a

metric that collects aspects both distance and link quality. High RSSI values indicate transmissions received from the nearest nodes and also with lesser probability of the loss provoked by signal attenuation. Thus, constructing paths avoiding the use of ''weak'' RSSI links consequently implies more reliable and adequate routes for data transmissions.

The next Section is dedicated to present the study conducted to assess the performance of the proposed LOADng-IoT-Mob.

V. PERFORMANCE EVALUATION

To assess the performance of the proposed LOADng-IoT-Mob in mobile IoT scenarios, the authors have conducted several simulation studies using Cooja Simulator [34], which is a part of Contiki OS [35]. Cooja is a well known and widely used tool that allows the emulation of the real behavior motes, (*e.g.* TMote Sky and Zolertia Z1), in terms of processing power, memory, and energy consumption. Although simulation tools may not constitute a real wireless environment, they represent a critical tool that can reproduce equivalent conditions to compare different protocols, making the study more reliable, fair, and free from the interference of external factors.

TABLE 2. Considered network scenarios.

Scenario Name	Network Area	Num. Nodes	Speed Min-Max	Mobility Config.	% of Mobile Nodes
Scenario 1	200 m	10, 20, 30, 40, 50	1∼3 m/s	All moves	100%
Scenario 2				Only simple moves	80%
Scenario 3				Only IC moves	20%
Scenario 4		30	1∼1, 1∼3, 3∼5, 5∼7, 7∼9 m/s	All moves	100%
Scenario 5				Only simple moves	80%
Scenario 6				Only IC moves	20%

In the performed study, the proposed solution was compared with the default LOADng, LOADng with SmartRREQ, LOADng with Expanding Ring, and LOADng-IoT. The application and network model considered was the same presented in subsection II-B. Although in the scenarios described in the referred subsection did not regard the mobility of all network devices [36], this study considers a ''worst case'' scenario, where all nodes can move in the network area. Thus, six different scenarios that varied in the number of nodes and the maximum speed of the mobile devices were created. In all scenarios, the Internet-connected nodes were 20% of the network devices. Table 2 presents all considered scenarios. The mobility pattern of the nodes was generated based on the Random Waypoint mobility model [37] using the BonnMotion 3.0.1 [38] tool. Table 3 exposes the most relevant parameters adopted in the study common to all scenarios. Table 4 shows the configuration parameters

TABLE 3. Network parameters common to all scenarios.

Parameter	Value
Network Execution Time	600 s
Traffic Pattern	P2P and MP2P
Data Packet Frequency	10 s ∼ 15 s
Data Packet Length	512 bits
Medium Access Control (MAC) Protocol	Carrier Sense Multiple Access (CSMA)
Radio Duty Cycle (RDC) Protocol	ContikiMAC
Check Channel Rate (CCR)	16 Hz
Radio Environment	Unit Disk Graph Model (UDGM) - Distance Loss
Transmission Range	50 m
Interference Range	50 m
TX and RX Chance	90%
Mote Type	Tmote Sky

TABLE 4. Configuration parameters of compared approaches.

LOADng parameters	
Parameter	**Value**
NET_TRAVERSAL_TIME	2 sec
RREQ_RETRIES	1
RREQ_MIN_INTERVAL	2 sec
R_HOLD_TIME	60 sec
MAX_DIST	65535
B_HOLD_TIME	4 sec
MAX_HOP_LIMIT	255
RREQ_MAX_JITTER	1 sec
RREP_ACK_REQUIRED	FALSE
USE_BIDIRECTIONAL_LINK_ONLY	FALSE
RREP_ACK_TIMEOUT	2 sec
LOADng-ExpRing parameters	
Parameter	**Value**
MNB_START	1
MNB_INCREMENT	2
MNB_THRESHOLD	7
LOADng-IoT parameters	
Parameter	**Value**
USE_INTERNET_ROUTE_CACHE	TRUE
R_INTERNET_HOLD_TIME	120 sec
NUM_ROUTE_CACHE_ENTRIES	2
LOADng-IoT-Mob parameters	
Parameter	**Value**
NEXT_HOP_VALID_TIME	60 sec
HELLO_MOB_INTERVAL	60 sec
WEAKRSSI_THRESHOLD	-50

defined for each studied approach. The values were defined in an empirical way to allow all the approaches reaching a consistent performance. Note that all the proposals of improvement adopt parameters from the default LOADng. Moreover, LOADng-IoT-Mob uses the parameters of all the available approaches (as common) and only the last three parameters are introduced by the new proposed solution. Furthermore, LOADng-SmartRREQ approach does not need any configuration parameter using only the one provided by the default LOADng. The studied routing solutions were compared regarding five metrics: packet delivery ratio, packet delivered with low latency, control bit overhead per delivered

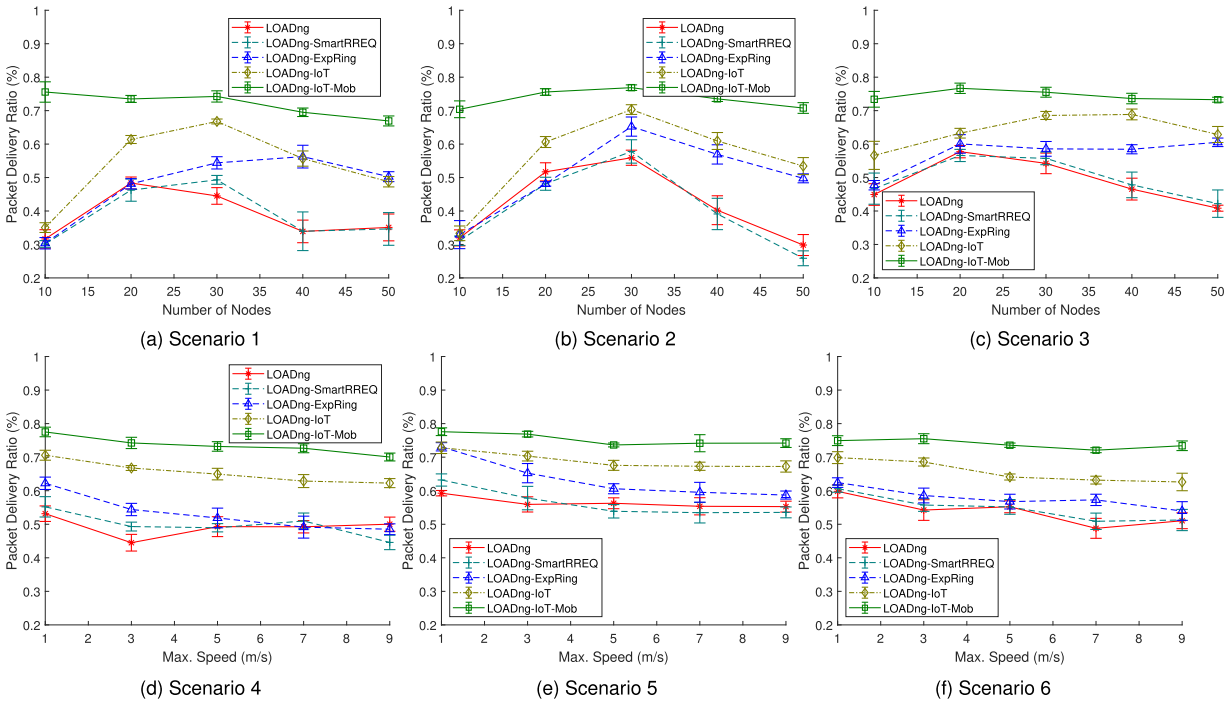


FIGURE 7. Packet delivery ratio in function of the number of nodes (a, b, c) and maximum nodes' speed (d, e, f) for LOADng, LOADng-SmartRREQ, LOADng-ExpRing, LOADng-IoT, and LOADng-IoT-Mob.

data bit, energy spent per delivered data bit, and memory usage. The obtained results for each metric have been presented and analyzed separately. For all studied scenarios, the simulations were executed 30 times, and the results show a confidence interval of 95%.

Figure 7 exposes the results for the metric of packet delivery ratio (PDR), which presents the ratio between the data packets that were successfully delivered to its final destination and all the data packets sent. The obtained results show that the proposed LOADng-IoT-Mob can overcome all other studied proposals in all studied scenarios. The mechanism proposed to harness the control messages were seen to be efficient to the Routing Set management, avoiding the sending of data packets through broken paths. With the use of the proposed solution, the message sender can know about the availability of the next hop and, thus, start a new discovery process when detects that the previously existent path was broken. The proposed solution also exposed to be consistent in scenarios with mobility of the whole network devices, different from the other compared solutions. It is also important to note that in the scenarios with 10 nodes, the other solutions exposed a very low PDR, mainly in Scenarios 1 and 2, when the percentage of the mobile nodes was over 80%. This has been justified by the low density of the nodes in the area, where the number of available routes was limited and the inability of self-adaption to the topology changes provoked by the mobility of the nodes led the solutions without mobility support to a poor performance. In the networks with 20 and 30 nodes, the majority of the studied solutions presented the best results. It was observed that this reasonable quantity of nodes exposed a proper balance between density and the interference produced by the devices. In the network with more than 30 nodes, the proposals that did not implement an efficient flooding mechanism presented a considerable performance decrement. The high number of broadcasts required by the route discovery processes were executed almost at the same time by several nodes generating a high probability of packet collisions, contributing to the PDR decrease. The proposed LOADng-IoT-Mob, in contrast, has not suffered from this problem since it has adopted features both from the Expanding Ring and LOADng-IoT for performing efficient route discoveries. These features, with the addition of weakRSSI and the control message harnessing mechanism, have permitted the LOADng-IoT-Mob to have a reliable and adequate performance regarding the PDR in the studied scenarios. In Scenarios 4, 5, and 6, where the number of nodes is fixed in 30 and the speed of the nodes is variable, all studied approaches have exposed a few variations in the results, proposed solution reaching the best ones. Thus, it was noted that the PDR performance of the studied solutions are few affected by the speed of the nodes when varied below 9 m/s, according to the obtained results.

Figure 8 presents the results obtained to the metric of packets delivered with low latency (PLL). This metric exposes the percentage of data packets successfully delivered with latency lower than a predefined threshold. The threshold value should

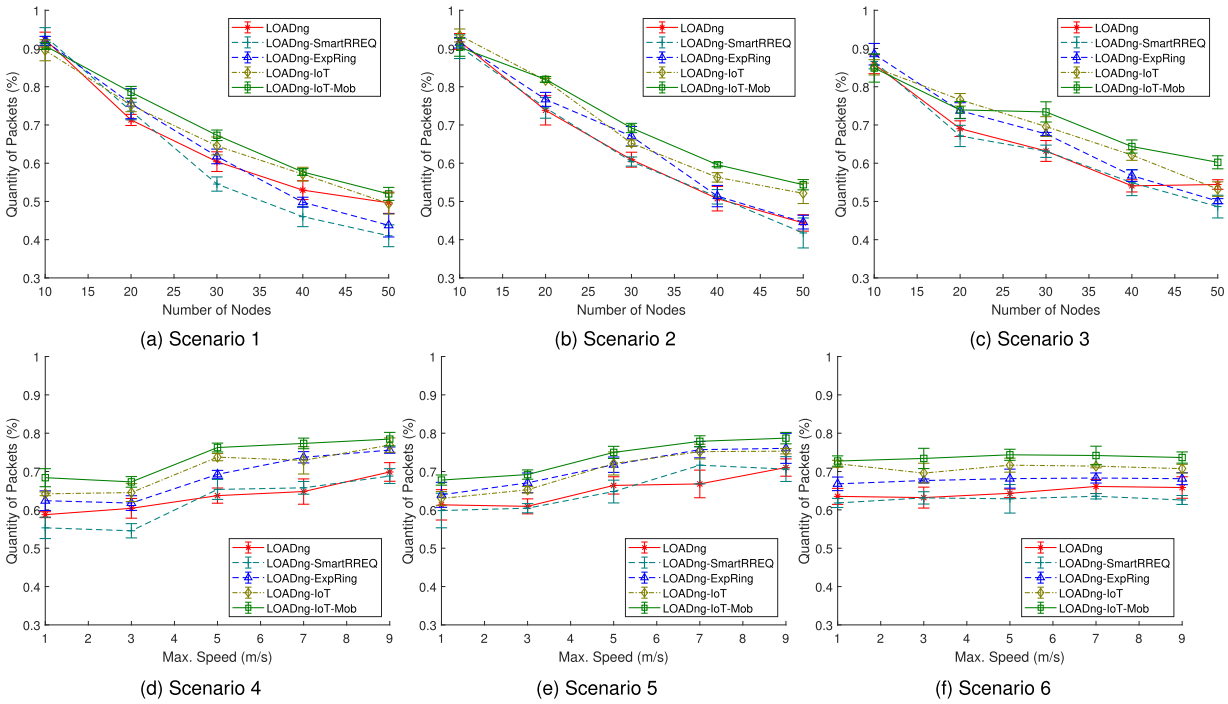


FIGURE 8. Quantity of data packets delivered with latency lower than 0.5 seconds in function of the number of nodes (a, b, c) and maximum nodes' speed (d, e, f) for LOADng, LOADng-SmartRREQ, LOADng-ExpRing, LOADng-IoT, and LOADng-IoT-Mob.

be defined according to the goals of the application to measure how much the proposal can fulfill its requirements. This work set the threshold value for low latency as 0.5 seconds once the considered network was used by a smart home IoT application. According to the obtained results, the LOADng-IoT-Mob has accomplished, in majority, the best results compared to the other studied approaches. However, a decrement in performance was observed with the increasing number of network nodes. This behavior is already expected since the increasing of the network density in a wireless environment can provoke the increasing of the end-to-end latency as a whole [39]. Contrarily, in the scenario with variable speed and mobility over 80%, all studied approaches presented slightly better PLL values with increasing device speed. It was noted that in the higher speed scenarios, the number of messages delivered in one-hop transmission increased in comparison with messages delivered through multi-hop. Thus, the percentage of packets delivered with lower latency increased. In general, the capacity of path shortening of the proposed approach contributed to the faster data packet delivery. Also, the use of combined flooding control mechanism has reduced the overhead and, consequently, the size of the contention window for performing the transmissions. As a result, the time for message forwarding and the latency was also reduced.

Figure 9 shows the results obtained for the metric of control bit overhead per delivered data bit (COB). COB metric represents the average control overhead needed to provide each data bit successfully. The metric value is computed based on the ratio between the sum of the control bit transmitted by all the network and the amount of data bit successfully delivered. Thus, this metric exposes the network efficiency concerning overhead. The results obtained for the metric of COB exposed that for all the studied scenarios, the proposed solution required less control overhead to deliver each data bit. The mechanism proposed for managing the Routing Set and topology changes helps the node detect broken path and create new routes to send data packets. Thus, although the nodes are forced to perform new discovery processes and use more control messages, the proposal allows the nodes to find new paths and successfully deliver the data packets. Further, the proposed LOADng-IoT-Mob introduced a new HELLO_MOB control message to boost the Routing Set refreshing mechanism. In the first impression, one can assume that the use of a periodical control message can increase the overhead. However, the obtained results showed the opposite. In the proposed solution, the HELLO_MOB messages were only used when the control overhead was not sufficient enough to maintain the information about the availability of the next hop nodes updated. Thus, even when used, HELLO MOB messages have shown more benefits than harm, allowing the nodes to know about the movement of their neighbors and avoiding sending of data packets through broken paths.

The results obtained to the metric of energy spent per delivered data bit (ESB) are shown in Figure 10. The ESB

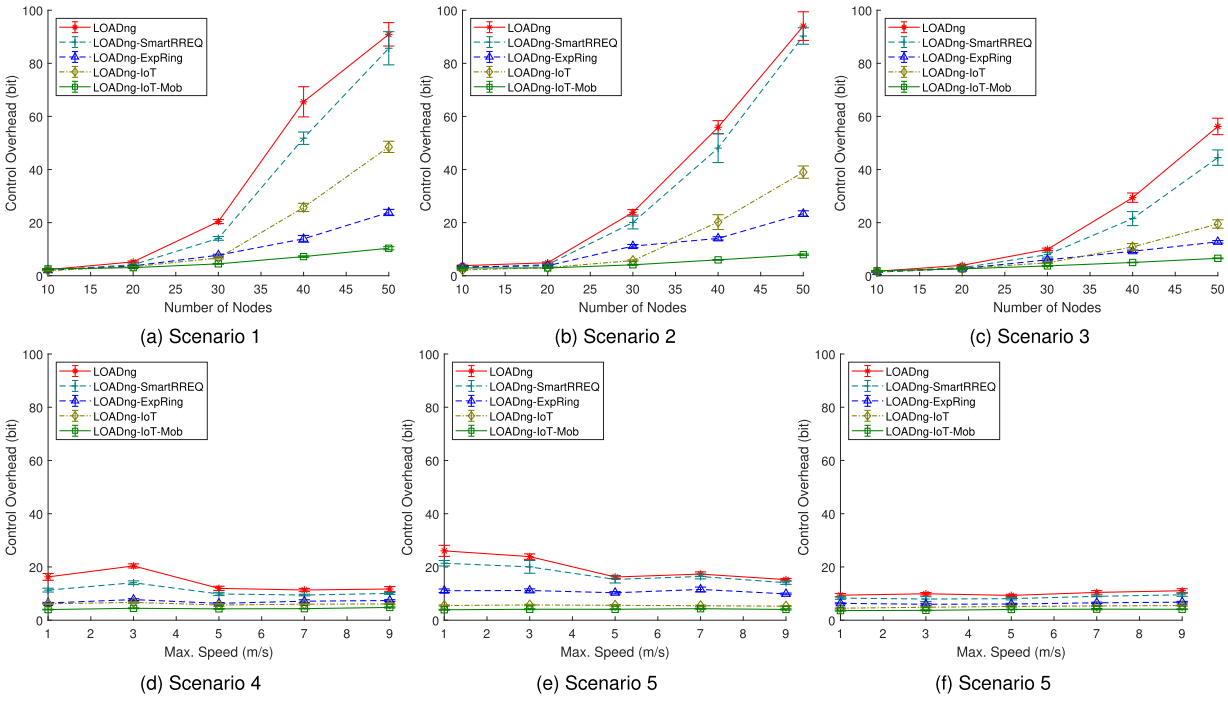


FIGURE 9. Control bit overhead to delivery each data bit in function of the number of nodes (a, b, c) and maximum nodes' speed (d, e, f) for LOADng, LOADng-SmartRREQ, LOADng-ExpRing, LOADng-IoT, and LOADng-IoT-Mob.

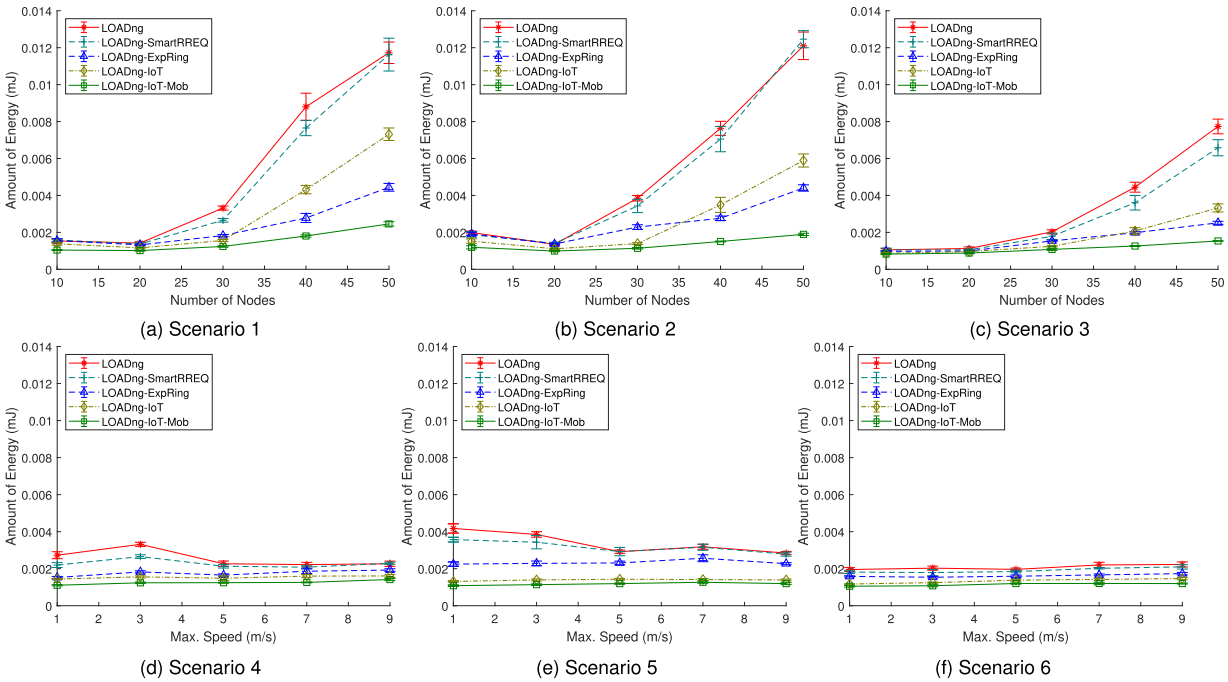


FIGURE 10. Energy spent to delivery each data bit in function of the number of nodes (a, b, c) and maximum nodes' speed (d, e, f) for LOADng, LOADng-SmartRREQ, LOADng-ExpRing, LOADng-IoT, and LOADng-IoT-Mob.

metric presents the energy efficiency of the network showing the average amount of energy, in millijoules (mJ), required to provide each data bit successfully. The value is obtained with the ratio between the amount of energy spent by the whole network and the amount of data bit delivered. Thus, a high PDR together with a low energy consumption represents

better ESB values. In this work, the values obtained by the ESB metric were a ''shadow'' of the values obtained for the COB metric. These results exposed that in the studied scenarios, the majority of the energy consumption was related to the transmission and reception of control messages. Thus, the proposal with lower overhead generated also presented lower energy consumption. The results obtained exposed that the proposed solution attained better energy efficiency compared to the other approaches in all the studied scenarios. Also, the proposed approach has exposed to be less affected by the network density growth and the increase in the number of mobile devices in the scenario when compared with the other solutions. LOADng-IoT-Mob, by using different methods for reducing and controlling the flooding of control messages, was able to require fewer transmissions, which implies a lesser energy consumption, to execute the route discovery processes. Thus, together with the high PDR values obtained, the proposed solution was able to attain the best ESB values. Regarding the scenarios with variable speed, all the studied solutions presently do not show considerable changes with the increase in the average speed of the nodes.

TABLE 5. Memory usage.

Proposal	Flash (kB)	RAM (kB)
LOADng	29.47	5.28
LOADng-SmartRREQ	29.79 (+ 1.09 %)	5.28
LOADng-ExpRing	29.87 (+ 1.35 %)	5.28
LOADng-IoT	31.94 (+ 8.38 %)	5.33 (+ 0.85 %)
LOADng-IoT-Mob	32.35 (+ 9.77 %)	5.36 (+ 1.55 %)

Table 5 presents the memory usage of the implemented and studied routing proposals. The exposed values, expressed in kilobytes (kB), cover all codes deployed in the nodes comprising the application, routing protocol, network stack, and operational system. The Flash column represents the amount of memory used for code implementation, *i.e.*, the read-only data. The RAM column, in contrast, shows the read-write data that is manipulated according to the code execution. The percentage inside the parenthesis gives the increased memory usage about the default LOADng implementation. The proposed LOADng-IoT-Mob required almost 10 % more Flash memory, which represents 2.88 kB, in comparison with the default LOADng. Further, the proposed approach increased the RAM usage by only 1.55 %. The presented memory usage increase is already expected due to the code increments required to reach the goals of the proposal and provide several benefits to the network. Considering that the performance assessment study has used Tmote Sky motes, which have 48 kB of Flash capacity and 10 kB of RAM, it is possible to observe that the proposed approach has consumed 67.3 % of the device Flash memory while the default LOADng has consumed 61.3 %. Concerning RAM usage of Tmote Sky, the proposed solution required 53.6 % while the default proposal demanded 52.8 %. Thus, although it has presented a higher memory usage, the proposed mobility improvement does not jeopardize all devices' memory and leaves a considerable portion of memory free to be used by more complex applications and future updates in the device firmware. Also, the increased memory usage can be seen as a reasonable trade-off or an ''necessary evil'' by all improvements provided to the network performance. Based on the exposed results, the next section in dedicated to present the conclusions of the work and show important future issues to be developed.

Based on the exposed results, the next section presents the conclusion of the work and shows important future issues that need to be studied.

VI. CONCLUSION AND FUTURE WORK

This work proposed a new improvement method for LOADng protocol in IoT scenarios with mobile devices. The LOADng-IoT-Mob proposal was designed to cover the lacks presented by the default LOADng regarding the mobility of nodes and to provide an efficient and reliable routing solution for the considered scenarios. The proposed solution merged features of other important enhancements for LOADng in the state-of-the-art and proposed new mechanisms for reaching a suitable performance in mobility environments.

The LOADng-IoT-Mob adopted the flooding controlling mechanism from the Expanding Ring and the SmartRREQ and inherited the Internet route discovery mechanism from the LOADng-IoT to create a solution that is able to find routes for different traffic types with low overhead and high reliability. Also, the proposal introduced a new mechanism of control message harnessing to allow the nodes to manage the discovered routes and self-adapt to the topology changes provoked by the movement of devices. Thus, the LOADng-IoT-Mob permitted the nodes to manage the information about the next hop of the paths existent on the Routing Set without additional fields in the control messages. Further, within of the procedure of control message harnessing, a new mechanism was introduced to permitted the nodes to shorten the path for destinations that had moved to near to them, reducing the route to one-hop. Considering that the control message harness mechanism depends on the overhead generated by the nodes, a new HELLO_MOB message was also introduced to be used when the usual network overhead is not sufficient to maintain the information of the neighbors' nodes updated in the Routing Set. Finally, a new weakRSSI routing metric was introduced to allow the nodes to construct more reliable and nearest paths to avoid an earlier break in the route. The proposed solution was compared with different approaches existent in the literature in scenarios with various node densities and speeds. For the metric of PDR, PLL, COB, and ESB, the proposed solution was able to present a significant and considerable performance increase. Thus, based on the obtained results in the studied scenarios, the LOADng-IoT-Mob was able to attain better data delivery reliability, lower latency, lower control overhead, and better power efficiency compared with the other studied solutions in the considered mobile IoT network scenarios. As a negative point, the proposed solution required increased memory

usage; however, it can present a good trade-off based on the obtained benefits.

For future work, the authors intend to optimize the source code implementation to reduce the required memory usage and assess the proposed solution in real testbeds. The authors also detach the necessity of the development of new studies and improvements for the LOADng becomes more reliable and efficient, mainly concerning the flooding mechanism.

REFERENCES

[1] E. Borgia, "The Internet of Things vision: Key features, applications and open issues," *Comput. Commun.*, vol. 54, pp. 1–31, Dec. 2014.

[2] I. Bisio, C. Garibotto, A. Grattarola, F. Lavagetto, and A. Sciarrone, "Exploiting context-aware capabilities over the Internet of Things for industry 4.0 applications," *IEEE Netw.*, vol. 32, no. 3, pp. 101–107, May/Jun. 2018.

[3] Q. Liu, Y. Ma, M. Alhussein, Y. Zhang, and L. Peng, "Green data center with IoT sensing and cloud-assisted smart temperature control system," *Comput. Netw.*, vol. 101, pp. 104–112, Jun. 2016.

[4] L. Sanchez, L. Munoz, J. A. Galache, P. Sotres, J. R. Santana, V. Gutierrez, R. Ramdhany, A. Gluhak, S. Krco, E. Theodoridis, and D. Pfisterer, "SmartSantander: IoT experimentation over a smart city testbed," *Comput. Netw.*, vol. 61, pp. 217–238, Mar. 2014.

[5] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[6] T.-A. Do, S.-W. Jeon, and W.-Y. Shin, "How to cache in mobile hybrid IoT networks?" *IEEE Access*, vol. 7, pp. 27814–27828, 2019.

[7] M. Z. Hasan, F. Al-Turjman, and H. Al-Rizzo, "Analysis of cross-layer design of quality-of-service forward geographic wireless sensor network routing strategies in Green Internet of Things," *IEEE Access*, vol. 6, pp. 20371–20389, 2018.

[8] R. Alexander, A. Brandt, J. P. Vasseur, J. Hui, K. Pister, P. Thubert, P. Levis, R. Struik, R. Kelsey, and T. Winter, *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, document RFC 6550, Mar. 2012. [Online]. Available: https://rfc-editor.org/rfc/rfc6550.txt

[9] O. Iova, P. Picco, T. Istomin, and C. Kiraly, "RPL: The routing standard for the Internet of Things... or is it?" *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 16–22, Dec. 2016.

[10] M. Bouaziz, A. Rachedi, and A. Belghith, "EKF-MRPL: Advanced mobility support routing protocol for Internet of mobile things: Movement prediction approach," *Future Gener. Comput. Syst.*, vol. 93, pp. 822–832, Apr. 2019.

[11] K. Q. A. Fadeel and K. El Sayed, "ESMRF: Enhanced stateless multicast RPL forwarding for IPv6-based low-Power and lossy networks," in *Proc. Workshop IoT Challenges Mobile Ind. Syst. (IoT-Sys)*, New York, NY, USA, 2015, pp. 19–24.

[12] G. G. Lorente, B. Lemmens, M. Carlier, A. Braeken, and K. Steenhaut, "BMRF: Bidirectional multicast RPL forwarding," *Ad Hoc Netw.*, vol. 54, pp. 69–84, Jan. 2017.

[13] W. Gan, Z. Shi, C. Zhang, L. Sun, and D. Ionescu, "MERPL: A more memory-efficient storing mode in RPL," in *Proc. 19th IEEE Int. Conf. Netw. (ICON)*, Dec. 2013, pp. 1–5.

[14] C. Kiraly, T. Istomin, O. Iova, and G. P. Picco, "D-RPL: Overcoming memory limitations in RPL point-to-multipoint routing," in *Proc. IEEE 40th Conf. Local Comput. Netw. (LCN)*, Oct. 2015, pp. 157–160.

[15] M. Zhao, A. Kumar, P. H. J. Chong, and R. Lu, "A comprehensive study of RPL and P2P-RPL routing protocols: Implementation, challenges and opportunities," *Peer-Peer Netw. Appl.*, vol. 10, no. 5, pp. 1232–1256, Sep. 2017.

[16] T. Clausen, J. Yi, and U. Herberg, "Lightweight on-demand ad hoc distance-vector routing—Next generation (LOADng): Protocol, extension, and applicability," *Comput. Netw.*, vol. 126, pp. 125–140, Oct. 2017.

[17] Y. B. Zikria, M. K. Afzal, F. Ishmanov, S. W. Kim, and H. Yu, "A survey on routing protocols supported by the Contiki Internet of Things operating system," *Future Gener. Comput. Syst.*, vol. 82, pp. 200–219, May 2018.

[18] S. R. Das, C. E. Perkins, and E. M. Belding-Royer, *Ad hoc On-Demand Distance Vector (AODV) Routing*, document RFC 3561, Jul. 2003. [Online]. Available: https://rfc-editor.org/rfc/rfc3561.txt

[19] D. Sasidharan and L. Jacob, "A framework for the IPv6 based implementation of a reactive routing protocol in ns-3: Case study using LOADng," *Simul. Model. Pract. Theory*, vol. 82, pp. 32–54, Mar. 2018.

[20] G. Premsankar, M. Di Francesco, and T. Taleb, "Edge computing for the Internet of Things: A case study," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1275–1284, Apr. 2018.

[21] J. Yi, T. Clausen, and Y. Igarashi, "Evaluation of routing protocol for low power and Lossy networks: LOADng and RPL," in *Proc. IEEE Conf. Wireless Sensor (ICWISE)*, Dec. 2013, pp. 19–24.

[22] J. V. V. Sobral, J. J. P. C. Rodrigues, K. Saleem, and J. Al-Muhtadi, "Performance evaluation of LOADng routing protocol in IoT P2P and MP2P applications," in *Proc. Int. Multidisciplinary Conf. Comput. Energy Sci. (SpliTech)*, Jul. 2016, pp. 1–6.

[23] J. Yi and T. Clausen, "Collection tree extension of reactive routing protocol for low-power and lossy networks," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 3, 2014, Art. no. 352421.

[24] S. Elyengui, R. Bouhouchi, and T. Ezzedine, "A comparative performance study of the routing protocols RPL, LOADng and LOADng-CTP with bidirectional traffic for AMI scenario," in *Proc. Int. Conf. Smart Grid Clean Energy Technol. (ICSGCE)*, Oct. 2015, pp. 43–49.

[25] D. Sasidharan and L. Jacob, "Improving network lifetime and reliability for machine type communications based on LOADng routing protocol," *Ad Hoc Netw.*, vol. 73, pp. 27–39, May 2018.

[26] J. L. Sobrinho, "Algebra and algorithms for QoS path computation and hop-by-hop routing in the Internet," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, vol. 2, Apr. 2001, pp. 727–735.

[27] J. V. V. Sobral, J. J. P. C. Rodrigues, N. Kumar, C. Zhu, and R. W. Ahmad, "Performance evaluation of routing metrics in the loadng routing protocol," *J. Commun. Softw. Syst.*, vol. 13, no. 2, pp. 87–95, 2017.

[28] J. Yi, T. Clausen, and A. Bas, "Smart route request for on-demand route discovery in constrained environments," in *Proc. IEEE Int. Conf. Wireless Inf. Technol. Syst. (ICWITS)*, Nov. 2012, pp. 1–4.

[29] A. Bas, J. Yi, and T. Clausen, "Expanding ring search for route discovery in loadng routing protocol," in *Proc. 1st Int. Workshop Smart Technol. Energy, Inf. Commun.*, 2012, pp. 1–8.

[30] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, K. Saleem, and V. Furtado, "LOADng-IoT: An enhanced routing protocol for Internet of Things applications over low power networks," *Sensors*, vol. 19, no. 1, p. 150, 2019.

[31] *2.4 GHz IEEE 802.15.4 / ZigBee-Ready RF Transceiver*, Chipcon Products, Texas Instrum., Dallas, TX, USA, 2006.

[32] A. Booranawong, K. Sengchuai, and N. Jindapetch, "Implementation and test of an RSSI-based indoor target localization system: Human movement effects on the accuracy," *Measurement*, vol. 133, pp. 370–382, Feb. 2019.

[33] R. D. Gomes, D. V. Queiroz, A. C. L. Filho, I. E. Fonseca, and M. S. Alencar, "Real-time link quality estimation for industrial wireless sensor networks using dedicated nodes," *Ad Hoc Netw.*, vol. 59, pp. 116–133, May 2017.

[34] J. Eriksson, F. Österlind, N. Finne, N. Tsiftes, A. Dunkels, T. Voigt, R. Sauter, and P. J. Marrón, "COOJA/MSPSim: Interoperability testing for wireless sensor networks," in *Proc. 2nd Int. Conf. Simulation Tools Techn. (Simutools)*. Brussels, Belgium: Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, 2009, pp. 27:1–27:7.

[35] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki—A lightweight and flexible operating system for tiny networked sensors," in *Proc. 29th Annu. IEEE Int. Conf. Local Comput. Netw.*, Nov. 2004, pp. 455–462.

[36] G. Porcu, J. Buron, and A. Brandt, *Home Automation Routing Requirements in Low-Power and Lossy Networks*, document RFC 5826, Apr. 2010. [Online]. Available: https://rfc-editor.org/rfc/rfc5826.txt

[37] C. Bettstetter, G. Resta, and P. Santi, "The node distribution of the random waypoint mobility model for wireless ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 2, no. 3, pp. 257–269, Jul. 2003.

[38] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, and M. Schwamborn, "BonnMotion: A mobility scenario generation and analysis tool," in *Proc. 3rd Int. ICST Conf. Simulation Tools Techn. (SIMUTools)*. Brussels, Belgium: Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, 2010, pp. 51:1–51:10.

[39] J.-C. Kuo, W. Liao, and T.-C. Hou, "Impact of node density on throughput and delay scaling in multi-hop wireless networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 10, pp. 5103–5111, Oct. 2009.

JOSÉ V. V. SOBRAL received the B.S. degree in computer science from the Centro de Ensino Unificado de Teresina (CEUT), Teresina, Brazil, and the M.Sc. degree in computer science from the Federal University of Piauí (UFPI), Teresina. He is currently pursuing the Ph.D. degree with the University of Beira Interior (UBI), Covilhã, Portugal. He is also with the Federal Institute of Maranhão (IFMA), São Luís, Brazil, where he is an Assistant Professor, and he is also a member of NetGNA Research Group. His research interests include the Internet of Things (IoT), routing protocols for low power and lossy networks, wireless sensors networks, RFID systems, and computational intelligence.

JOEL JOSÉ P. C. RODRIGUES (S'01–M'06–SM'06) is currently a Professor with the Federal University of Piauí, Brazil, and a Senior Researcher with the Instituto de Telecomunicações, Portugal. He is also the Leader of the Internet of Things Research Group (CNPq). He has authored or coauthored over 750 papers in refereed international journals and conferences, three books, and he holds two patents. He is a Licensed Professional Engineer (as a Senior Member), a member of the Internet Society, and a Senior Member of ACM. He is also a member of many international TPCs and participated in several international conferences organization. He is the Director for the Conference Development—the IEEE ComSoc Board of Governors, the IEEE Distinguished Lecturer, the President of the Scientific Council at ParkUrbis - Covilhã Science and Technology Park, the Past-Chair of the IEEE ComSoc Technical Committee on eHealth, the Past-Chair of the IEEE ComSoc Technical Committee on Communications Software, Steering Committee Member of the IEEE Life Sciences Technical Community and Publications Co-Chair, and a Member Representative of the IEEE Communications Society on the IEEE Biometrics Council. He had been awarded the several Outstanding Leadership and Outstanding Service Awards by the IEEE Communications Society and several best papers awards. He has been the General Chair and a TPC Chair of many international conferences, including the IEEE ICC, GLOBECOM, and HEALTHCOM. He is the Editor-in-Chief of the *International Journal of E-Health and Medical Communications* and the *Journal of Multimedia Information System*, and an Editorial Board Member of several high-reputed journals.

RICARDO A. L. RABÊLO received the B.Sc. degree in computer science from the Federal University of Piauí, Brazil, in 2005, and the Ph.D. degree in power systems from the São Carlos Engineering School, University of São Paulo, Brazil, in 2010. His research interests include smart grid, the Internet of Things, intelligent systems, and power quality.

KASHIF SALEEM received the B.Sc. degree in computer science from Allama Iqbal Open University, Islamabad, Pakistan, in 2002, the P.G.D. degree in computer technology and communication from Government College University, Lahore, Pakistan, in 2004, and the M.E. degree in electrical engineering—electronics and telecommunication and the Ph.D. degree in electrical engineering from University Technology Malaysia, in 2007 and 2011, respectively. He took professional trainings and certifications from the Massachusetts Institute of Technology (MIT), IBM, Microsoft, and Cisco. He is currently an Associate Professor with the Center of Excellence in Information Assurance (CoEIA), King Saud University, Riyadh, Saudi Arabia. He has authored or coauthored over 100 papers in refereed international journals and conferences. His research interests include ubiquitous computing, mobile computing, the Internet of Things (IoT), machine-to-machine (M2M) communication, wireless mesh networks (WMNs), wireless sensor networks (WSNs) and mobile ad hoc networks (MANETs), intelligent autonomous systems, information security, and biologically inspired optimization algorithms. He has organized, co-organized, and served as a technical program committee member in numerous renowned international workshops and conferences. He acquired several research grants in KSA, EU, and the other parts of the world. He is an Associate Editor of the *Journal of Multimedia Information System* (JMIS), IEEE Access, *International Journal of E-Health and Medical Communications* (IJEHMC), and *International Journal of Cyber-Security and Digital Forensics* (IJCSDF).

SERGEI A. KOZLOV received the engineering degree (Hons.) in quantum electronics from the Leningrad Institute of Fine Mechanics and Optics (currently ITMO University), Saint Petersburg, Russia, in 1982, and the Ph.D. and Dr. Sci. Phys. and Maths. degrees from the Saint Petersburg State University, Saint Petersburg, in 1986 and 1997, respectively. From 1986 to 2002, he was with ITMO University, Saint Petersburg, as an Engineer, an Assistant Professor, an Associate Professor, and a Full Professor with the Physics Department, Natural Science Faculty, where he has been a Full Professor and the Head of the Photonics and Optoinformatics Department, and the Dean of the Photonics and Optoinformatics Faculty, since 2002. He has also been the Head of the International Institute of Photonics and Optoinformatics, ITMO University, Saint Petersburg, since 2013. He has authored more than 250 articles. His research interests include femtosecond optics and femtotechnologies, nonlinear optics of few-cycle pulses and ultrafast data transmission, terahertz optics and biophotonics, and quantum informatics. He is a member of SPIE and D.S. Rozdestvenskiy Optical Society.

Chapter 6

Multicast Improvement for LOADng in Internet of Things Networks

This chapter consists in the following paper:

Multicast Improvement for LOADng in Internet of Things Networks

José V.V. Sobral, Joel J. P. C. Rodrigues, Ricardo L. Rabêlo, and Jalal Al-Muhtadi

Measurement, Elsevier, ISSN: 0263-2241, 2019.

DOI: doi.org/10.1016/j.measurement.2019.106931

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 3.364

ISI Article Influence Score (2019): 0.516

Journal Ranking (2019): 22/91 (Engineering, Multidisciplinary)

Journal Ranking (2019): 13/64 (Instruments & Instrumentation)

Measurement 148 (2019) 106931

Contents lists available at ScienceDirect

Measurement

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

Multicast improvement for LOADng in Internet of Things networks

José V. V. Sobral [a,b], Joel J. P. C. Rodrigues [a,c,d,*], Ricardo A. L. Rabêlo [c], Jalal Al-Muhtadi [d]

[a] Instituto de Telecomunicações, Universidade da Beira Interior, Covilhã, Portugal
[b] Federal Institute of Maranhão (IFMA), São Luís, MA, Brazil
[c] Federal University of Piauí, Teresina, PI, Brazil
[d] College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 12372, Saudi Arabia

ARTICLE INFO

Article history:
Received 28 May 2019
Received in revised form 1 August 2019
Accepted 5 August 2019
Available online 12 August 2019

Keywords:
Internet of Things
LOADng
Low power and Lossy Network
Multicast
Routing Protocol

ABSTRACT

The Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation (LOADng) is an emerging routing solution for low-power wireless networks, which is receiving much attention and is being adopted in Internet of Things (IoT) networks. Although the current literature has introduced improvements to allow the protocol to fulfill the requirements of a higher set of IoT applications, the multicast support stills a deficiency of the protocol. Thus, to overcome this limitation, this study proposes a multicast improvement for LOADng. The proposal, named Multicast LOADng (M-LOADng) introduces a new route discovery process to enable the construction of a multicast routing tree and perform forwarding multicast data messages. The proposed solution also introduces mechanisms to reduce the control message overhead and increase the reliability of the route discovery process. Further, M-LOADng allows multicast data transmission in three different modes. The conducted experiments, through a real testbed, show that the proposed solution can reach a reliable and constant performance in terms of quality-of-service, energy consumption, and memory usage, even when increasing the multicast group size.

1. Introduction

The Internet of Things (IoT) paradigm has emerged in recent years due to its high potential to provide facilities both for citizens and enterprises [1]. In the IoT concept, things (objects), which are equipped with micro-controllers, become devices connected to the Internet with the capacity of exchange messages with any device using the World Wide Web [2,3]. In general, IoT scenarios can involve different technologies such as Radio Frequency Identification (RFID) [4,5], Long Term Evolution (LTE/4G) [6], Wi-Fi [7], and Wireless Sensors Networks (WSN) [8], making emerge integrated heterogeneous scenarios [9]. Thus, IoT gives rise to a significant new set of applications, varying from simple automatized lighting systems [10] to full control and management of industrial plants [11,12], smart healthcare systems [13,14], or smart cities [15–17]. In this context, the reliability and efficiency of communication among network devices become an essential task.

Routing protocols are responsible for constructing and maintaining routes among network nodes. Thus, network performance is strongly dependent on the behavior of routing protocols. Although recent literature presents several protocols and improvements for wireless networks, only a limited number of them are especially designed to fulfill the requirements of IoT applications. Among them, it is possible to highlight the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [18] and the Lightweight On-demand Ad hoc Distance-vector Routing Protocol – Next Generation (LOADng) [19,20]. While RPL is considered the standard routing protocol for IoT [21], LOADng is still in draft and has emerged as a more lightweight and less complex solution.

Several IoT applications, including Industrial IoT [22–24], can take advantage and require the sending of data messages to groups of nodes, generating a multicast data traffic. As an example, a central manager device can send multicast messages to a specific group of nodes to adjust their operations and reach the desired temperature. Also, a central device can start or stop machines operating simultaneously using a single transmission. The Internet Engineering Task Force (IETF) has confirmed the multicast need for support in the Request for Comments (RFC) 5673 [25] document, which discusses the functional requirements of routing protocols for Industrial LLNs. However, the two main routing approaches for the above-mentioned IoT networks does not present feasible and efficient support for this traffic pattern. RPL, in its Mode of Operation (MOP) 3, can send multicast messages but requires high

* Corresponding author at: Federal University of Piauí, PPGEE Campus Universitário Ministro Petrônio Portella Bairro Ininga, Teresina - PI CEP: 64049-550, Brazil.
E-mail addresses: jose.sobral@it.ubi.pt (J.V.V. Sobral), joeljr@ieee.org (J.J.P.C. Rodrigues), ricardoalr@ufpi.edu.br (R.A.L. Rabêlo), jalal@ccis.edu.sa (J. Al-Muhtadi).

https://doi.org/10.1016/j.measurement.2019.106931
0263-2241/

memory usage and complex implementations of an unclear description [26].

In contrast to RPL, the LOADng does not present any support for multicast message forwarding. As aforementioned, multicast data communication represents an essential feature for IoT applications since it allows sending of similar data messages to a specific group of nodes through a reduced number of transmissions. Thus, a routing protocol designed for LLN must offer multicast support efficiently and reliably. Although recent approaches have improved the LOADng functioning and added new features to the main protocol core, to the best of the authors' knowledge, the state-of-the-art does not present an approach supporting multicast messages on the LOADng. Based on this limitation, this work proposes a new mechanism to overcome this drawback and increase LOADng adoption. The proposal, named Multicast LOADng (M-LOADng), introduces a new multicast route discovery mechanism that allows multicast group leaders (hereinafter also called sink nodes or gateways) to create a routing tree to a set of network nodes that decide to join a multicast group. The built tree is used to forward multicast data messages, but it also can be harnessed by all the other nodes to send data to the sink. The M-LOADng also introduces mechanisms to reduce control message usage without affecting the process of route discovery. Finally, the proposal allows the use of different forwarding modes, allowing it to attend to the requirements of various applications. Thus, the main contributions of this work are the following:

- Support to LOADng protocol in building a multicast routing tree to perform routing of multicast messages with efficient performance;
- Allow the use of the built multicast routing tree to perform data message forwarding from the nodes to the sink without needing expensive new route discovery processes;
- Reduce the control message overhead and improve the reliability of the route discovery process;
- Offer different multicast data message forwarding modes to cover and fulfill the requirements of different IoT applications; and
- Present a comprehensive performance evaluation analyzes of the proposed solution conducted through a real testbed.

Throughout the document, the term downward data traffic will be adopted to refer to the multicast data flow sent from sink nodes to other network nodes. Moreover, the term upward data traffic will be used to indicate the data traffic that flows from simple nodes to the sink.

The rest of this work is organized as follows. Section 2 shows the relevant related works on the topic and Section 3 highlights important features of LOADng that are required to detail the proposed solution and presents the collection tree extension for LOADng. Section 4 introduces the proposed M-LOADng, detailing the approach features and operations, while the performance assessment study is presented in Section 5. Section 6 concludes the paper and suggests future work.

2. Related work

The initial multicast solutions for wireless networks with constrained hardware emerged with the growth of MANET concepts. In [27,28], the authors studied and classified these solutions in a survey work. More recently, the Multicast Protocol for LLNs (MPL) [29] was defined as the IETF multicast standard solution for LLNs through the RFC 7731. In MPL functioning, nodes do not create a logical topology to perform message forwarding. Thus, multicast data forwarding is executed with the use of a flooding mechanism controlled by Trickle timers. To avoid loops and the transmission of duplicated packets, MPL uses an additional field at the control payload of the multicast messages, indicating the sender identifier and the message sequence number. Network nodes store information about the already received messages and regularly exchange control packets for maintaining network consistency.

To reduce the overhead and constant exchange of control messages required by MPL, the Stateless Multicast RPL Forwarding (SMRF) [30] was proposed. The introduced solution uses the tree structure build by RPL to propagate the information about a multicast nodes group. Thus, using the RPL control messages, the node informs the multicast group to its parent in the tree structure. In the SMRF, advising a multicast group is only done upward in the routing tree. Hence, the proposal only supports sending multicast messages in the downward flow (from above to below in the routing tree). The Enhanced SMRF (ESMRF) [31] was proposed to overcome the limitation of SMRF and allow both upward and downward multicast sending. However, the ESMRF solution involves the use of the root node to enable upward multicast traffic requiring a higher number of transmissions to perform multicast data forwarding. To surpass this expensive solution, the Bidirectional Multicast RPL Forwarding (BMRF) was introduced in [26]. BMRF merges important features of RPL and SMRF, and offers the use of different transmission modes according to the number of neighbors interested at the reception of a multicast data packet. However, the solution presented by BMRF can increase high end-to-end delay and energy consumption. Further, the necessity of more memory efficiency in RPL multicast approaches is highlighted in [32].

As an alternative to RPL, the LOADng was proposed as a lightweight and less-complicated routing solution for LLNs and IoT scenarios. Recent enhancements have improved the LOADng, making it reach a performance similar to RPL [33] and, in some cases, even better [34]. In [35], the SmartRREQ mechanism is proposed to enhance the control message flooding of the LOADng, which is admittedly high due to the reactive features of the protocol. The solution allows nodes to reduce the number of broadcast transmission during the route discovery, contributing to a lower control overhead. In [36], the authors present the Expanding Ring flooding mechanism, which, used together with SmartRREQ, reduces the range of control messages permitting the protocol, decreasing the number of transmissions and packet collisions.

A collection-tree extension (CTP) for the LOADng is proposed in [33]. The proposal allows the LOADng to construct a routing tree started from a root node to improve the protocol performance in data collection applications. Also, the LOADng-CTP enables sending data in different directions using the built routing tree. The LOADng-CTP is better presented and discussed in the next section. In [37], the authors present an improvement for the LOADng focused on IoT scenarios. The proposal enables the nodes without an Internet connection to autonomously find gateways. The proposal, which requires a low control message overhead, also introduces a routing cache mechanism helping the route discovery process by contributing to reducing collisions and energy consumption.

Multicast solutions focused and designed for IoT networks are presented in [38–40]. Approaches based on the Stainer tree algorithm to create a multicast routing tree were proposed in [38,39]. In [40] the authors proposed a geographic-based multicast routing protocol for IoT applications. This proposal presents preconditions for its adoption in networks whose nodes are location-aware (*i.e.* equipped with GPS) and statically positioned. Thus, based on their set of initial conditions and requirements for being deployed, these solutions become not very interesting for being adopted in practical IoT environments.

The recent improvements presented to LOADng make the protocol more reliable and efficient, mainly for IoT scenarios. However, as above-mentioned, there are no state-of-the-art solutions to provide multicast support for LOADng. With the consent that multicast support solutions for IoT are based on RPL, in general, and they tend to require high memory usage and complex implementations, this work proposes a new improvement for the LOADng supporting multicast data routing in a more lightweight way.

3. LOADng and LOADng-CTP

The proposed solution introduced in this work uses the core structure of LOADng routing protocol. Thus, to facilitate the understanding of the proposal functioning, this section presents the main features of the LOADng. Further, it also presents the LOADng-CTP extension, which is currently the most suitable extension for IoT applications.

3.1. LOADng

The LOADng is a routing protocol designed for wireless networks based on devices with reduced hardware capacity. The protocol was inspired by the well-known AODV and presents a reactive route discovery mechanism in which the path among the nodes is created on-demand. The core structure of the LOADng is composed by four control messages and three data structures, which should be maintained by each network device. The control messages, which are used for both the creation and maintenance of paths among nodes, are Route Request (RREQ), Route Reply (RREP), Route Reply Acknowledgement (RREP_ACK), and Route Error (RERR). The fields of RREQ, RREP, and RREP_ACK messages, which are the most important and used during the protocol functioning, are presented in Table 1. The data structures, namely Routing Set, Blacklisted Neighbor Set, and Pending Acknowledgement (Ack) Set, are used to store information about the neighbor devices and the routes to the known destinations. Table 2 shows the fields of Routing Set and Pending Ack Set, the two data structure more frequently used during the LOADng functioning.

Table 1
Fields of the RREQ, RREP, and RREP_ACK messages.

Field	Size (bits)	Description
		RREQ and RREP messages fields
`type`	8	type of message
`addr-length`	4	length of originator and destinations addresses
`seq-num`	16	sequence number that uniquely identifies each message of an originator
`metric-type`	8	metric type used to construct the route
`route-metric`	32	value computed based on the used metric type
`hop-count`	8	number of times that message was transmitted
`hop-limit`	8	maximum number of hops permitted
`originator`	var	address of the message originator
`destination`	var	address of the message destination
`ack-required`	1	if flagged, indicates the need of an acknowledgment message (only used with RREP)
		RREP_ACK message fields
`addr-length`	4	length of originator and destinations addresses
`seq-num`	16	sequence number of the RREP that will be confirmed
`destination`	var	address of the message destination, *i.e.*, the RREP originator address

Legend: var = variable according to the adopted addressing scheme (*e.g.*, IPv4, IPv6, or Rime).

Table 2
Routing Set and Pending Acknowledgement Set fields.

Field	Size (bits)	Description
		Routing Set
`R_dest_addr`	var	address of route destination
`R_next_addr`	var	address of next hop to the destination
`R_metric_type`	8	metric type used to compute the route metric
`R_metric`	32	value computed to the route based on the metric type
`R_hop_count`	8	number of hops to reach the destination
`R_seq_num`	16	sequence number of the message used to create/refresh the route entry
`R_bidirectional`	1	boolean flag that if TRUE, indicates a bidirectional route
`R_local_iface_addr`	var	address of the interface used to communicate with the destination
`R_valid_time`	16	valid time of the route
		Pending Ack Set
`P_next_hop`	var	address of the node which the RREP was sent
`P_originator`	var	address of the RREP originator
`P_seq_num`	16	value of the `seq-num` field of the sent RREP
`P_ack_received`	1	boolean flag set as TRUE when the corresponding RREP_ACK is received
`P_ack_timeout`	16	time to expire the entry

Legend: var = variable according to the adopted addressing scheme (*e.g.*, IPv4, IPv6, or Rime).

In protocol functioning, when a node wants to find a route to a destination, it should create an RREQ message to start the route discovery. Thus, the generated RREQ, indicating the address of the desired destination, is sent in broadcast. Each node that receives the RREQ should update some message fields, store information about the message originator and the previous hop node on its Routing Set, and check the message destination. If the node address is different from the RREQ destination, the node should check the maximum hop limit of the message and broadcast it to the node neighbors. Otherwise, the node should reply to the received route request using an RREP message destined to the RREQ originator. The generated RREP is sent in unicast through the reverse path from which the RREQ was received. Each node that receives a RREP message should update the message fields and refresh its Routing Set. The message should then be forwarded to the next hop in the path to the message destination using the information recorded in the Routing Set during the RREQ propagation. This process is performed until the RREP reaches its goal. Finally, when the RREQ originator receives the RREP, the route discovery process is concluded, and the data messages can be sent through the constructed path.

Aiming to improve the reliability of the constructed paths, the LOADng optionally offers the use RREP_ACK message to confirm the reception of RREP messages. Thus, at the moment of an RREP generation, the node can opt to set a flag in the message named *ack_required*. Whenever an RREP message with *ack_required* flag set is sent, the sender node should insert an entry in the Pending Acknowledgement Set with information about the RREP next hop node and a valid time to wait for the RREP reply. The node that receives an RREP with *ack_required* flag set should reply to the message sender with an RREP_ACK to confirm the RREP reception. If the entry inserted in the Pending Acknowledgement Set expires without RREP_ACK reception, the next hop of the sent RREP should be inserted in the Blacklisted Neighbor Set, and future messages received from it should be ignored. The LOADng route discovery process using RREQ, RREP, and RREP_ACK messages is illustrated in Fig. 1.

LOADng reactive functioning makes this protocol simple, lightweight, and easily implementable. Also, using the on-demand

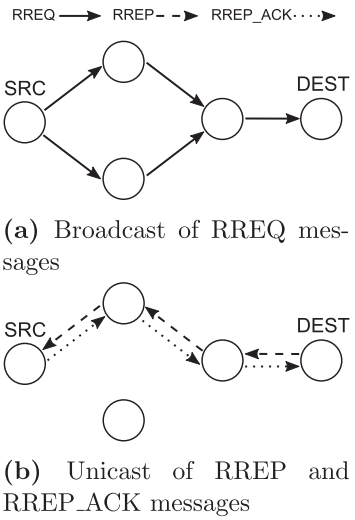


Fig. 1. LOADng route discovery process; (a) Source (SRC) node broadcasts an RREQ message that is forwarded until reaching the destination (DEST) node; (b) DEST node unicasts an RREP message to SRC through the path created by the RREQ. Each intermediate node that receives a RREP, before forwarding it to SRC, sends an RREP_ACK to the previous hop.

route discovery process, the support on mobile devices in the network is inherently provided. However, to allow a feasible adoption in IoT scenarios, some improvements should be performed. The next subsection details the LOADng-CTP cited in Section 2.

3.2. LOADng-CTP

The LOADng-CTP protocol proposes a routing tree-based extension to improve the performance of LOADng in data collection applications in which data messages generally flow from nodes to a central device. The new extension allows a central node (also named a sink node, root node, or gateway) to work proactively and create a routing tree at the beginning of the network functioning. Thus, all the network nodes can previously construct a path to the center node and avoid the execution of route discovery at the time of data sending.

The LOADng-CTP proposal requires the inclusion of two new fields on the RREQ message named *collection_tree_trigger* and *collection_tree_build*, a new control message termed HELLO, and a new data structure called Neighbor Set. In the protocol functioning, the central node begins the collection tree building through the generation of an RREQ_TRIGGER message (RREQ with the *collection_tree_trigger* flag set). In the RREQ_TRIGGER, the fields used to indicate the message originator and destination should carry the same value, which is the address of the message originator node. This adoption allows that message to flow through the whole network without being replied to by a specific destination. After being created, the RREQ_TRIGGER is broadcasted.

Each node that receives an RREQ_TRIGGER message should insert the sender address on its Neighbor Set with the status HEARD, which indicates a uni-directional link, and to schedule the sending of a HELLO message according to a predefined random jitter. Finally, if the same RREQ_TRIGGER was not already received from another sender, the node should forward it in a broadcast.

The HELLO message sent after the defined jitter should carry the addresses of all the neighbors (at Neighbor Set) for which that node has received an RREQ_TRIGGER. The nodes that receive a HELLO should verify whether its address is included in the message. If true, the node should insert the message sender on its Neighbor Set with the status SYM, which indicates a bi-directional link. To avoid duplicated entries in the Neighbor Set, only the addresses already available in the set are updated.

At the moment of RREQ_TRIGGER sending, the root node should schedule the generation of an RREQ_BUILD message (RREQ with the *collection_tree_build* flag set) for 2×NET_TRAVERSAL_TIME, which is a parameter to define the expected time for a message traversing the whole network. After being created, the RREQ_BUILD message is sent via broadcast. Each receiver of the RREQ_BUILD should initially verify its Neighbor Set and check if the message was received from a bi-directional link (SYM). In a negative case, the message is dropped without any additional processing. Otherwise, if the same message was not received and processed previously, the receiver should insert a new entry on the Routing Set with the destination equal to the RREQ_BUILD originator and the next hop equal to the message sender (the previous hop of the RREQ_BUILD). After that, the message is updated and broadcasted again. This procedure continues until all of the network nodes receive the RREQ_BUILD and conclude the construction of the routing tree.

The initial procedure of LOADng-CTP can create a routing tree to facilitate the sending of data messages from the nodes to a central device (the root of the routing tree). This traffic pattern, which is sometimes named upward or MP2P traffic, is expected by an extensive set of current IoT applications. However, some applications, such as the Industrial IoT, can require the sending of messages from the root to other devices, creating a downward data flow. In this case, LOADng-CTP allows the creation of these "reverse paths" with the use of the RREP_REQUIRED parameter. Hence, the nodes configured with the RREP_REQUIRED parameter as true should send an RREP message after receiving an RREQ_BUILD. Similarly to the default LOADng, the RREP is forwarded in unicast through the reverse path from the received request. Thus, when the RREP reaches the root device, the route is built and the downward traffic can be more easily forwarded without the need of new discovery processes.

The routing tree building process of LOADng-CTP only occurs at the begin of network functioning. However, due to the lossy environment of LLNs, some control messages can be lost, and devices cannot be included in the constructed tree. In this case, the node out of the tree should work following the default LOADng and perform the route discovery process to the central device. This process is also applied to new nodes that want to join the network and nodes, which, for some reason, have lost their route to the sink and need to re-discover the path to the root. Although CTP extension can improve LOADng functioning and make the protocol more suitable for IoT networks, it still fails to support an important traffic pattern required by a large set of applications: multicast traffic. Thus, the next section presents a proposal to provide multicast support for LOADng.

4. Proposed Multicast LOADng

LOADng is an emerging routing protocol that has been receiving attention from the research community due to its simplicity, easy implementation, and efficiency. However, the protocol still presents several limitations, mainly related to multicast data traffic forwarding support. Although LOADng-CTP presents an adaption that can facilitate the creation of routes both upward and downward, the proposed method can, substantially, increase the network overhead. Furthermore, a high memory usage can be necessary to maintain the path to each device interested in receiving downward traffic.

To overcome the described limitation, this work proposes the M-LOADng improvement. The proposal aims to allow the LOADng protocol to work using the concepts of multicast groups for performing the routing of multicast data messages in a feasible and lightweight manner. Thus, M-LOADng introduces a route discovery mechanism that allows central nodes to build a multicast routing tree for the sending and forwarding of multicast messages. The created routing tree can also be used by nodes to forward data to the central node facilitating upward data sending. Besides, to reduce the overhead of the route discovery, M-LOADng proposes a simple mechanism to reduce the sending of reply message and to ensure they are received. Finally, to better fit the requirements of different applications and the efficient use of network resources, M-LOADng allows multicast data forwarding in three different modes: unicast, broadcast, and mixed.

The next subsections present a detailed description of M-LOADng characteristics and functioning. First, the required structure applied over the LOADng core for the multicast support provision is introduced. Next, the M-LOADng modules and procedures are detailed.

4.1. M-LOADng required additional structures and parameters

M-LOADng requires some increments in the LOADng core structure to allow the creation of paths for multicast data forwarding. Thus, the proposed solution introduces two new types of control messages, named Multicast RREQ (M_RREQ) and Multicast RREP (M_RREP). Both message types use the same structure of RREQ and RREP messages, respectively, plus an additional field termed `multicast_group` used to indicate the multicast group of the message. Further, the proposed solution also adds three new fields on the Routing Set fields: `R_multicast_group`, `R_multicast_route_type`, `R_to_root`. The first field, `R_multicast_group`, is used to define the multicast group related to the route entry. The second, `R_multicast_route_type`, indicates the type of multicast route and can assume two values MULTICAST_ORIG and MULTICAST_DEST. The last additional field, `R_to_root`, is used to indicate whether the route destination is the root node.

The proposed solution introduces two new mechanisms to reduce the use of M_RREPs and ensure their delivery by employing a transmission retry approach. Thus, the Pending Ack Set is improved to allow the storage of M_RREP messages and a short field, named `P_retry_count`, to record the number of sending retries. All the additional fields required by M-LOADng are presented in Table 3. The proposed M-LOADng also introduces four new parameters to allow the proposal configuration:

- MULTICAST_FOWARDING_MODE: defines the mode of multicast data transmission. This parameter can assume three different values: BROADCAST_MODE, UNICAST_MODE, and MIXED_MODE.
- MIXED_MODE_UNICAST_TH: sets a threshold for the use of multiple unicast transmission. This parameter is only used when MULTICAST_FORWARDING_MODE is defined as MIXED_MODE.
- MAX_RREP_RETRY: defines the maximum number of tries for successfully transmitting a M_RREP message.
- PERIODICAL_MAINTENANCE_INTERVAL: defines the time interval (in seconds) between for performing the maintenance of the built multicast routing trees.

The use of all the fields and parameters introduced by the M-LOADng are better explained in the following subsections, in the course of the description of the proposed solution. Notice that this subsection is limited to presenting the additional fields and parameters introduced by M-LOADng. Thus, other message fields and parameters mentioned in the following subsections that were not shown here are part of the default LOADng core presented in [19].

4.2. M-LOADng multicast route discovery

The main objective of M-LOADng is to introduce a lightweight mechanism to allow LOADng to forward multicast data messages with efficiency and high reliability. To achieve this goal, the proposal adopts the concept of multicast groups, in which each network node can be a member of a multicast group or can decide to join in one (or more) groups. Thus, under M-LOADng, the originator of multicast messages (also termed as multicast group leader) should build a multicast routing tree composed by the nodes interested in being a part of the group to, later, forward data messages. Multicast routing tree building is performed through a multicast route discovery process introduced by M-LOADng.

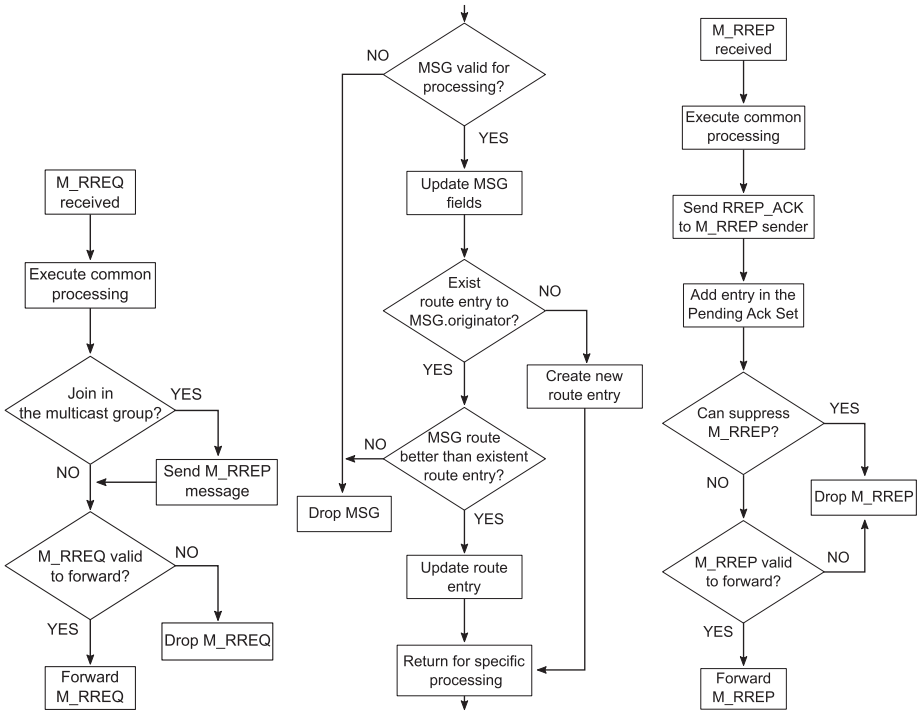
The multicast route discovery process starts when a multicast leader, hereinafter named sink node, wants to send data messages to a specific multicast group formed by nodes that are still unknown. Thus, for building the multicast routing tree composed by nodes interested in the data messages, the sink generates a new M_RREQ with its own address on the originator and destination fields of the message, and with the desired multicast group in the `multicast_group` field. Afterwards, the M_RREQ is sent in broadcast to flood the whole network. Each node that receives the M_RREQ message should process it according to the flowchart presented in Fig. 2a. Hence, initially, the received message is submitted to the common processing used for both M_RREQ and M_RREP messages.

During the common processing, detailed in the flowchart of Fig. 2b, it is first verified whether the received message is valid for processing. On this verification, the length of the addresses contained in the message are checked as well as whether a node has already received a message of the same originator with a higher sequence number, or the originator is on the Blacklisted Neighbor Set. If any of these checks are not attended to, the message should be dropped with no additional processing. In sequence, if the message is valid for processing, the node updates its message fields related to the hop count and routing metric. Next, the node checks its Routing Set to verify if exist some route entry in the message originator. In M-LOADng, routes are differentiated by a pair destination-multicast_group. Hence, a path is considered to be existent when an entry with a destination equal to the message originator and the entry multicast group corresponding to the message multicast group is found. Thus, if the route does not exist, a

Table 3
Additional fields required by M-LOADng over LOADng core.

Field	Size (bits)	Description
		M_RREQ and M_RREP messages
`multicast_group`	8	multicast group of the message
		Routing Set
`R_multicast_group`	8	multicast group of the route entry
`R_multicast_route_type`	1	type of multicast route entry
`R_to_root`	1	boolean flag that if TRUE indicates the route destination is the root node
		Pending Ack Set
`P_m_rrep_msg`	var	RREP message stored until the reception of the corresponding RREP_ACK
`P_retry_count`	8	counter of the number of RREP sending attempts

Legend: var = variable according to the adopted addressing scheme (*e.g.*, IPv4, IPv6, or Rime).



(a) M_RREQ processing (b) M_RREQ and M_RREP common processing (c) M_RREP processing

Fig. 2. M-LOADng control message processing flowchart.

new route entry is created and inserted in the Routing Set. Otherwise, the node should verify if the path from which the message was received is better than the existent in the Routing Set. If true, the route entry is updated. A creation or update of a route entry is done according to the type of received message. If the message received is a M_RREQ, the created/updated entry should have the `R_to_root` field set as `true` and the `R_multicast_route_type` field set as MULTICAST_ORIG. If the received message is an M_RREP then, the created/updated entry should have the `R_to_root` field set as `false` (in an entry update, if the field is `true`, it should not be changed), and the field `R_multicast_route_type` set as MULTICAST_DEST. All the other fields of the route entry are inserted/updated according to the default LOADng. If a message is not used to create or update a route entry, it should be dropped. Otherwise, after being processed, the message is returned to specific message processing.

Following M_RREQ processing, after returning from common handling, a node verifies the message multicast group field and should decide whether to join the group for receiving the multicast data messages. If the node chooses to participate in the multicast group, it should reply to the received request with a M_RREP message. Otherwise, the node checks whether the message is valid to be re-transmitted based on the hop count and hop limit fields. If valid, the message is transmitted in broadcast. Otherwise, the message is dropped.

A node that has decided to join the multicast group should generate a M_RREP message destined to the M_RREQ originator and within the same multicast group of the received message. Afterwards, the message is sent by unicast through the reverse path created by M_RREQ. Each intermediate node that receives an M_RREP should process it according to the flowchart presented in Fig. 2c. Thus, a message is initially subjected to common processing, as explained previously. After that, the receiver node must send an RREP_ACK message to the previous hop of the received M_RREP. Notice that, different to the default LOADng, in the multicast route discovery process of M-LOADng, RREP_ACK is not optional but mandatory. A node should then add a new entry in the Pending Ack Set and verify the possibility of suppressing forwarding of M_RREP message. The reply suppressing mechanism is better described in the next subsection. For now, considering that a message is not suppressed, the node should check whether the received M_RREP is valid to forward and perform transmission to the next hop in the path to its destination. This forwarding process can be performed until an M_RREP reaches its goal. However, different for the standard route discovery, in M-LOADng multicast route discovery, the process is not finished when the first reply reaches its destination. In this proposed solution, the multicast routing tree construction is completed $2\times$NET_TRAVERSAL_TIME after the M_RREQ is sent at the beginning of the discovery process. After finishing this process, the sink node can check whether at least one route to the desired multicast group was created. According to an application requirement, the sink node can start a new discovery process or accept that no node has opted to join the group.

The sink node that has created a multicast routing tree (i.e., the M_RREQ originator) should perform the maintenance of the built multicast tree. Thus, according to the time interval defined in the PERIODICAL_MAINTENANCE_INTERVAL parameter, a node sends a new M_RREQ with information about the multicast group to be maintained. The receivers of the M_RREQ that have previously

joined the multicast group do not need to reply to it, but they should use the message to update its Routing Set. New nodes that have been inserted in the network, or that want to join the group, should follow the regular procedure of M_RREQ processing and reply to it using an M_RREP. After being processed, M_RREQ should be sent (in broadcast) through the network to allow all nodes to ensure their paths to the multicast group leader are maintained and refreshed.

As in the default LOADng, in M-LOADng, each request and reply control message generated receives a new unique sequence number. This number prevents a message being processed more than once by the same node and avoids loops generation in the network. Furthermore, keeping the versatility of the LOADng, M-LOADng also permits the creation of routes using different routing metrics as an alternative to the default hop count.

As previously described, in M-LOADng, each node can decide if it wants to join a multicast group at the moment of M_RREQ reception. The current proposal implementation uses an unsigned integer value with 8 bits, allowing the use of 256 different multicast groups inside the network. Performing a cross-layer approach, each multicast group number can be identified by the nodes according to the application particularities. For example, in an Industrial IoT environment, multicast group 1 could be represented by the devices with a temperature sensor and multicast group 2 could be reserved for devices responsible for controlling the gas flow. In more complex scenarios, M-LOADng can be easily adapted to use queries as multicast groups and allow the creation of multicast routing trees using expressions as "nodes with last measured temperature higher than x °C". Thus, nodes that fulfilled the query can decide to join the multicast tree and send a reply to the request originator.

The multicast route discovery process, as aforementioned, is used to create a multicast routing tree and allow downward traffic. However, the flooding of M_RREQ messages also permits the nodes to construct routes to the sink node. Thus, the built routing tree can also be used by nodes to send data to the sink generating an upward flow, similar to the tree created by the LOADng-CTP. Hence, the proposed M-LOADng enables the sending of multicast data, but also provides support for the data collection application using the same routing structure.

4.3. M-LOADng route reply suppressing and retry mechanisms

In the course of the multicast route discovery process of M-LOADng, nodes can suppress sending M_RREP messages to reduce the control message overhead using the reply suppressing mechanism (RSM). Thus, even when a valid M_RREP is received, during the message handling presented in the previous subsection, a node should check if it is possible to suppress reply message forwarding.

The M_RREP suppression decision is made based on the Routing Set and the multicast group of the message. A received M_RREP can only be suppressed if an intermediate node has already sent an equivalent M_RREP to the same destination and multicast group. Thus, after properly adding the M_RREP originator to its Routing Set, the message receiver checks its Routing Set to look for an entry to the M_RREP destination and verifies whether it joined the multicast group indicated in that message. If both conditions are fulfilled, the intermediate node assumes that it has already joined the same multicast group and, consequently, has previously sent an equivalent M_RREP to the same destination of the received message. The proposed RSM is exemplified in Fig. 3. A node that suppresses a M_RREP becomes responsible for forwarding multicast data messages to the suppressed message originator. The M-LOADng multicast data forwarding mechanism is presented in the next subsection.

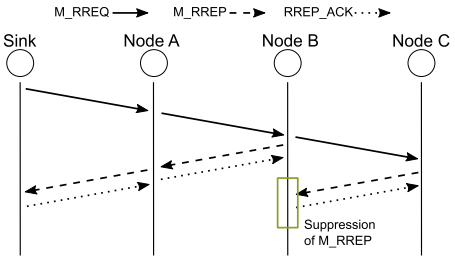


Fig. 3. M-LOADng reply suppressing mechanism. Node B suppresses M_RREP from C once it has already sent a similar reply to the same request. Node A chose not to join the group, but it is used as a forwarder.

RSM can significantly reduce the number of M_RREP message transmissions, mainly in network with a high number of nodes joining the multicast groups. Consequently, once the reception of an M_RREP leads to including or updating route entries in the Routing Set, the suppression mechanism also contributes to reducing the memory consumption of nodes. With the use of the RSM, the loss of one M_RREP message can imply a considerable negative effect. For example, as shown in Fig. 3, if the M_RREP from node A to the sink is lost, nodes B and C will not be able to receive multicast data messages from the sink. Thus, considering the lossy nature of LLNs and seeking to ensure a reliable routing performance, M-LOADng introduces a new reply retry mechanism (RRM).

In the proposed RRM, nodes should store each M_RREP message in a Pending Ack Set before sending it. Using the enhanced structure of the Pending Ack Set previously presented, nodes store the M_RREP with a predefined valid time. When an RREP_ACK is received confirming the delivery of M_RREP, the correspondent entry is removed from the Pending Ack Set. If the entry valid time expires without the reception of an RREP_ACK, the stored M_RREP message should be sent again. RRM tries to perform the M_RREP delivery according to the MAX_RREP_RETRY parameter. The proposed RRM is exemplified in Fig. 4.

The combination of both proposed mechanisms by the M-LOADng helps to reduce the network overhead and contributes to protocol energy efficiency. Although the RRM can increase the memory usage by temporarily storing M_RREP, it provides reliability in the transmission of reply messages, guaranteeing the path creation. Furthermore, the extra memory used by RRM is compensated by reducing the number of Routing Set entries provided by the suppressing mechanism. Thus, both mechanisms work in a harmonized manner to lead M-LOADng for efficient and reliable performance.

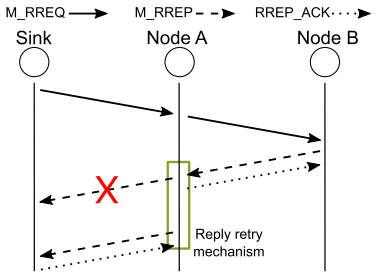


Fig. 4. M-LOADng reply retry mechanism. Node A uses the proposed RRM to resend the stored M_RREP short time after not receiving the RREP_ACK from the sink.

4.4. M-LOADng multicast data forwarding

Multicast data messages are not destined to one node but a group. Thus, multicast messages, in addition to the data payload used by the application, require overhead to indicate the message originator, a sequence number, and the destination multicast group. These extra data are used during the message routing process to differentiate them, avoiding an infinite flooding.

After performing a multicast route discovery process, it is expected that all nodes that intended joining the multicast group have replied to the request and a multicast routing tree is created. Hence, the multicast message can be sent according to the discovered paths. Thus, in M-LOADng, a sender node (the sink or an intermediate node) verifies its Routing Set for creating a list of the next hop nodes that should receive the multicast data message. The node should insert, into the list, the next hop addresses whose entries cumulatively attend (logical AND) the following conditions: i) the message multicast group is equal to the route entry multicast group; ii) the previous hop of the message is different from the route entry for next hop; iii) the message originator different from the route entry destination; and iv) the route entry multicast route type is equal to MULTICAST_DEST. To reduce the number of transmissions, the node should prevent the inclusion of duplicated addresses in the list of next hops.

After consulting the whole Routing Set and creating the next hop list, a node should check whether at least one next hop was found, and perform the message forwarding. This process is executed by each node that receives a multicast data message. Thus, forwarding is performed neighbor-by-neighbor without the need for the message originator to know an entire path to destinations of a multicast data. Fig. 5 exemplifies the M-LOADng multicast message forwarding mechanism. Considering the sink node has initiated and concluded a multicast route discovery process for multicast group 1, data forwarding can be performed using the built multicast tree. Table 4 complements the mentioned figure exposing the Routing Set of some nodes after the discovery process conclusion. Due to the use of RSM, the Routing Set of sink node does not have any entry to the nodes *G* and *F*, although these had sent an M_RREP to the sink. However, once the multicast forwarding is focused on the group instead of a destination, node *D*, having suppressed the M_RREP from *F* and *G*, after receiving the message from the sink, consults its Routing Set and can find the paths to both nodes.

Aiming to cover a large set of IoT applications and different performance requirements, the proposed M-LOADng provides three different modes for executing the transmission of multicast messages:

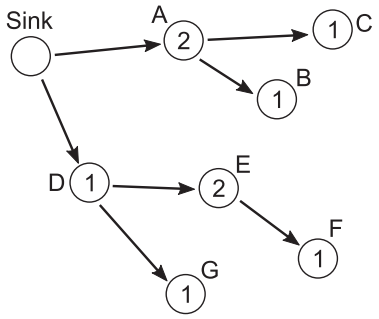


Fig. 5. M-LOADng multicast data message forwarding. The number inside the nodes represent their multicast group.

Table 4
Routing Set of nodes after M-LOADng multicast route discovery.

Dest	N_H	M_T	M_G
	Routing Set of sink node		
B	A	DEST	1
C	A	DEST	1
D	D	DEST	1
	Routing Set of node A		
Sink	Sink	ORIG	1
B	B	DEST	1
C	C	DEST	1
	Routing Set of node D		
Sink	Sink	ORIG	1
F	E	DEST	1
G	G	DEST	1

Legend: Dest = `R_dest_addr`; N_H = `R_next_addr`; M_T = `R_multicast_route_type`; M_G = `R_multicast_group`; DEST = MULTICAST_DEST; ORIG = MULTICAST_ORIG.

- *Broadcast mode*: forwarding multicast messages are always performed in broadcast. Thus, a node, after consulting its Routing Set looking for destinations, independent from the number of found next hops, transmits a message in broadcast.
- *Unicast mode*: transmission of multicast data is performed through one or multiples unicast. Thus, after obtaining the list of next hops, a node forwards data messages individually in unicast to each next hop in the list.
- *Mixed mode*: the transmission of multicast messages is performed according to the number of addresses available at the next hop list. Hence, based on the MIXED_MODE_UNICAST_TH parameter, a node should choose between sending only one broadcast (as in the Broadcast mode), or transmitting multiple unicasts (as in the Unicast mode). If the next hop list is higher than MIXED_MODE_UNICAST_TH, a node should use a broadcast. Otherwise, forwarding should be done through multiple unicasts.

The choice of forwarding mode should be done through the configuration of the MULTICAST_FOWARDING_MODE. When a broadcast or mixed mode is adopted, nodes should maintain a data structure to store the most recent multicast data messages received and prevent the forwarding of previously received and processed messages. Furthermore, it is essential to consider the application performance requirements during the choice of forwarding mode. In the next section, the performance of the proposed M-LOADng is evaluated considering the different contemplated forwarding modes. The effects of a forwarding mode choice are discussed and explained.

5. Performance assessment and results

This section presents the performance evaluation studies of the proposed solution and the analysis of the obtained results. The behavior of the M-LOADng with different forwarding modes (M-LOADng-Unicast, M-LOADng-Broadcast, and M-LOADng-Mixed) is compared with the LOADng-CTP. Following subsections show the configuration setup for the experiments and results obtained per performance metric related to data delivery reliability, energy and network efficiency, and memory usage, which are discussed and analyzed.

5.1. Experiment setup and scenarios

The experiments were performed in the FIT IoT-LAB testbed [41] using 50 M3 nodes. The adopted nodes are equipped with an ARM Cortex M3 microcontroller unit (MCU), which provides

72 MHz of processing capacity, 64 Kbytes of RAM (random-access memory), and 512 Kbytes of flash memory. M3 nodes are equipped with IEEE 802.15.4 PHY radio, which is regularly used for communication in Wireless Sensor Networks [42,43]. The devices communicate using the 2.4 GHz frequency and are power supplied by a 3.7 V LiPo battery of 650 mAh. The network nodes were deployed in a grid 2×25 in an area of 5×15 m, in which the sink node was positioned at the bottom-left of the grid. To allow multi-hop communication and better assessment of the route creation capacity for the studied solutions, the transmission power and reception RSSI threshold of nodes was set to 3 dBm and -72 dBm, respectively.

The comparison of the considered proposals was deployed on Contiki operating system [44]. The application developed to produce data traffic considered a network with a sink node and several regular nodes. The sink node generates multicast messages to a group of nodes at random time intervals ranging from 10 to 15 s, producing the downward traffic (or P2MP). Regular nodes sent simple data messages to the sink at random time intervals, changing between 50 and 60 s, creating the upward traffic (or MP2P). Thus, based on the generated traffic pattern, the developed application can cover a wide range of IoT applications, including those used in industrial environments as described in [25]. For example, this generated traffic can represent an industrial environment where a central device sends multicast data to control the temperature of a set of steam boilers, transmitting data downward (P2MP). Furthermore, each steam boiler periodically sends data to the central computer informing their current conditions and fluid produced, producing upward traffic (MP2P). Both central devices and steam boilers can be accessed through the Internet, allowing a worldwide industry monitoring and control its plants across the whole globe in a remote way. Table 5 presents the most important parameters for the experiments. All the other configuration parameters not described here were defined as the default of Contiki 3.0.

At the beginning of network functioning, in the scenarios using the M-LOADng, the sink node starts the multicast route discovery process to construct the routing tree and perform downward data sending. This procedure, as previously introduced in subSection 4.2, also creates a collection tree that permits the sending of upward traffic. Considering that LOADng-CTP does not support multicast traffic, the designed application was adapted to the sink node previously to know the addresses of the nodes in the multicast group. Thus, the LOADng-CTP performed its normal functioning to create the collection tree and permit upward traffic. Further, to permit the creation of routes from the sink to the simple nodes and allow downward traffic, the nodes used the RREP_REQUIRED as `true` to reply to the requests. Table 6 presents the parameters used in the LOADng-CTP. Table 7, plus the parameters of Table 6, presents the parameters adopted in the M-LOADng proposal.

The performance of the considered solutions was studied considering the growth of the multicast group size. Thus, it was created different scenarios by changing the percentage of simple nodes joining the multicast group in 20%, 40%, 60%, and 80%. The studied performance routing metrics were packet delivery ratio, data throughput, end-to-end latency, inter-packet latency, spent energy per delivered data bit, control overhead per delivered data bit, Routing Set, and memory usage. Each experiment scenario was executed 30 times and the obtained results, which are discussed in the next subsections, presented a confidence interval of 95%.

Table 5
Parameters of testbed experiments.

Parameter	Value
Network Area	5×15 m
Number of Nodes	50
Size of Multicast Groups	20%, 40%, 60%, 80%
Execution Time	600 s
Multicast Data Message Frequency	10 s–15 s
Simple Data Message Frequency	50 s–60 s
Data Message Length	512 bits
Medium Access Control (MAC) Protocol	Carrier Sense Multiple Access (CSMA)
Radio Duty Cycle (RDC) Protocol	ContikiMAC
Check Channel Rate (CCR)	32 Hz

Table 6
Parameters of LOADng-CTP and M-LOADng protocols.

Parameter	Value
Network Traversal Time	2 s
Routing Set Size	16
Route Entry Hold Time	60 s
Blacklist Entry Hold Time	4 s
Pending Ack Set Size	4
RREP_ACK Required	TRUE
RREP_ACK Timeout	2 s
Routing Metric	Hop Count
RREP Required	TRUE
RREQ Retries	1
RREQ Min. Interval	2 s
RREQ Max. Jitter	1 s
HELLO Min. Jitter	2 s
HELLO Max. Jitter	4 s

Table 7
Parameters of M-LOADng.

Parameter	Value
Multicast Forwarding Mode	Unicast, Broadcast, Mixed
Max. Reply Retries	2
Periodical Maintenance Inter.	60 s

5.2. Packet delivery ratio and data throughput

The results obtained for the packet delivery ratio are shown in Fig. 6. The obtained results for the multicast and simple data traffic are presented separately to allow the evaluation of the proposal for different traffic patterns. Using simple data traffic (Fig. 6a), LOADng-CTP and M-LOADng have presented close results, showing that the efficiency of the built data collection tree for both proposals is equivalent. Also, the variation of the M-LOADng forwarding mode has not implied significant changes in the delivery ratio of simple messages. Fig. 6b presents the packet delivery ratio of studied proposals for multicast data traffic. In contrast to the results of simple messages, LOADng-CTP presents an impracticable performance with the growth of the multicast group size. Contrarily, M-LOADng has reached a feasible and stable performance for the different scenarios. The proposed solution was able to deliver more than 90% of multicast packets, apart from the adopted forwarding mode. The introduced multicast discovery process allowed the creation of a multicast routing tree that facilitated the forwarding process. Further, the high delivery ratio shows that the introduced RRM was able to ensure the sending of reply messages for successful multicast tree building. In contrast, the low performance of LOADng-CTP is justified by the need of the sink node to maintain a route entry to all the nodes that have replied the route request. This behavior overloads the Routing Set and forces the replacement of already existent entries. In the performed experiment, the Routing Set size was set to 16 for all scenarios. Thus, when the number of nodes joined in the multicast group is greater that Routing Set size, LOADng-CTP begins to deliver a deficient performance and, consequently, limits protocol scalability. The referred limitation of LOADng-CTP does not affect the M-LOADng. In the proposed solution, although each node that joined the multicast group sends

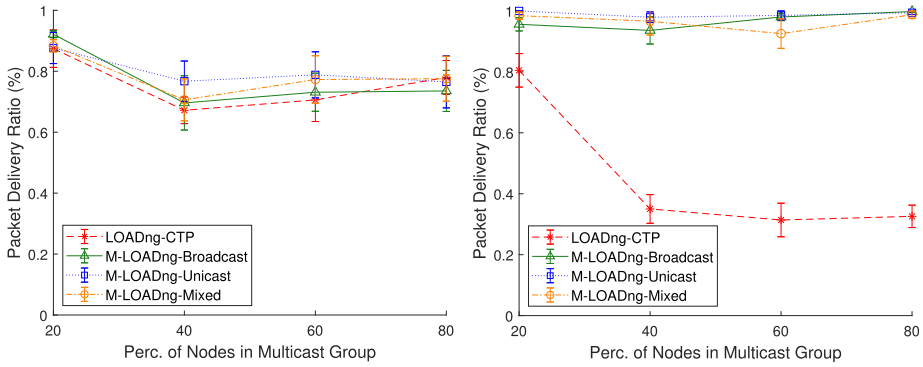


Fig. 6. Packet delivery ratio in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

a reply to the sink node, these messages can be suppressed by an intermediate node. This feature avoids growth in the number of entries in the Routing Set without affecting the construction of the multicast routing tree.

Fig. 7 presents the results obtained for the data throughput metric. Complementing the packet delivery results, data throughput shows the average number of data bits delivered successfully per second during the network execution. The metric considers the overall data traffic, thus, both simple and multicast data messages are considered in the metric computation. The obtained results show that the proposed solution overcomes the LOADng-CTP in all the studied scenarios. The efficiency of M-LOADng for both types of data message led the proposed solution to a high throughput, while the low scalability of LOADng-CTP made the protocol to reach the wrong results.

5.3. End-to-end and inter-packet latency

Fig. 8 presents the results obtained for the average end-to-end latency metric. This metric can be affected by several aspects, such as the number of hops between message source and destination, the density of nodes in the area, radio check channel rate, and so on. Considering the studied scenarios, the obtained results showed that M-LOADng overcomes the LOADng-CTP apart from the used forwarding mode. It was noticed that using a mechanism based on the use of several unicast transmissions can affect the latency

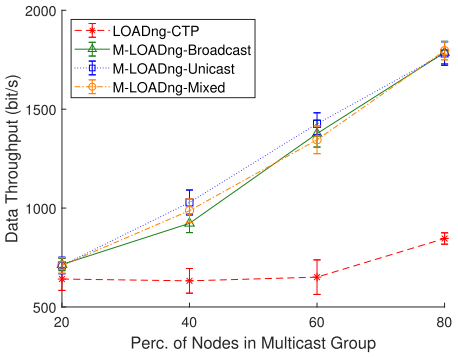


Fig. 7. Data throughput in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

both simple and multicast traffic messages. The current literature mentions that, depending on the adopted radio duty cycle (RDC) and medium access control (MAC) protocols, unicast transmissions can be faster than broadcasts [26,45]. However, in the studied dense network scenario, the use of several unicast transmissions produced a different effect. The adoption of unicast as the unique transmission mode increased the number of access to the frequency spectrum shared by the nodes radio. Thus, the MAC protocol made nodes spend more time waiting to access the wireless medium and perform the transmission with lower collisions probability.

In some cases, the computed end-to-end latency time also can involve the process of route creation and recovery. Thus, problems that may have occurred during the route discovery process can affect the latency when a route needs to be created at the moment of routing data messages. This behavior justifies the high latency presented by the LOADng-CTP, mainly to the multicast data traffic. Once the Routing Set of the nodes is quickly exhausted by the high number of reply message (as mentioned in the previous subsection), new routing discovery processes are necessary to forward the packets. It then increases the time to message delivery.

The inter-packet latency metric presents the time difference between the first and last multicast message with the same sequence number successfully delivered. The results using this metric are presented in Fig. 9. It allows the evaluation of the equality of message delivery time, which can be strongly sensible for some applications, such as Industrial IoT. Obtained results showed that the M-LOADng-Broadcast, followed by M-LOADng-Mixed, was able to deliver data messages with the lower latency differences among packets. Hence, the obtained results indicate that the approaches that support broadcast transmission can better fit applications requiring low latency.

5.4. Spent energy and control overhead per delivered data bit

The results obtained for the energy spent per delivered data bit (ESDB) metric are presented in Fig. 10. The ESDB metric shows the average amount of energy expended by the whole network to successfully deliver each data bit. Hence, the metric computation considers the energy consumed by all network nodes and all the delivered data messages (both simple and multicast). Thus, it is possible to measure the performance of studied approaches regarding energy efficiency. The obtained results showed that, for all studied scenarios, the proposed solution was able to spend less energy to deliver data bits compared to the LOADng-CTP. The high packet delivery and throughput of M-LOADng, together with the

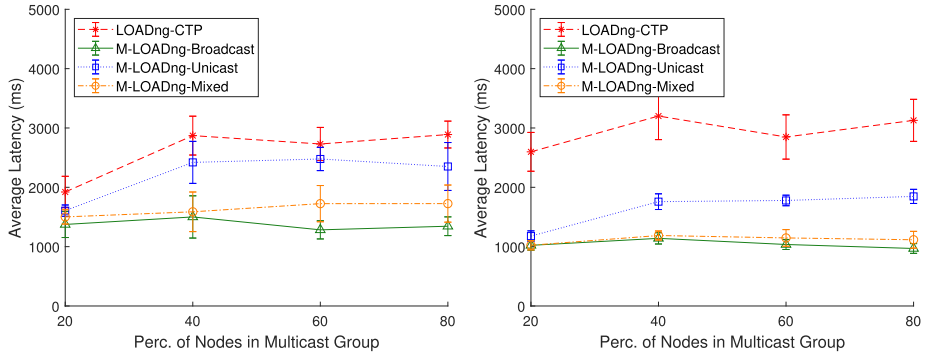


Fig. 8. End-to-end latency in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmissions modes.

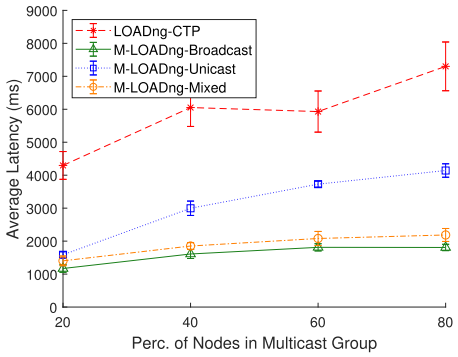


Fig. 9. Inter-packet latency for multicast data traffic in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

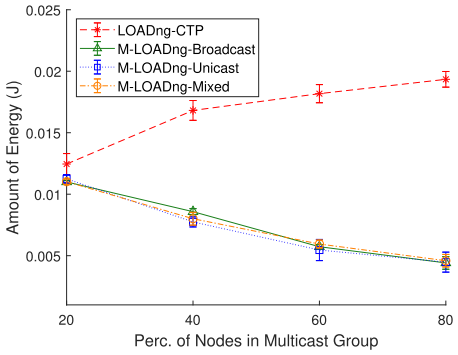


Fig. 10. Energy spent per delivered data bit in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

low required overhead for the construction of the paths, led the proposed approach to a performance, in some cases, three times better than the LOADng-CTP.

Fig. 11 exhibits the results obtained for the metric of control bit overhead per delivered data bit (CODB). The metric presents the average number of control bits used by the protocol to construct the routes and forwarding both simple and multicast data messages. The CODB computation considers all network overhead generated both for building path among the nodes and recovering the broken routes. Based on the performed experiments for the studied scenarios, it was able to verify that M-LOADng required a lower overhead than the LOADng-CTP. The proposed multicast route discovery mechanism, which can build both upward and downward routes in only one process, is shown to be more efficient than the proposal presented by LOADng-CTP. Further, the exhaustion of Routing Set entries of LOADng-CTP made the protocol perform more route discovery processes, producing a higher overhead and energy consumption.

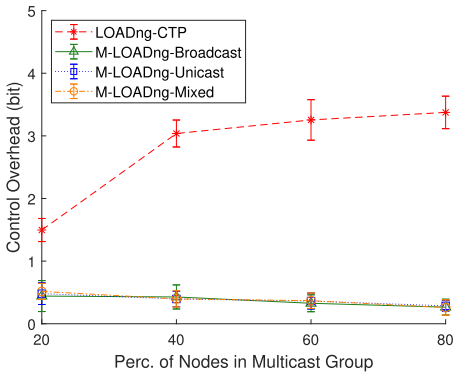


Fig. 11. Control bit overhead per delivered data bit in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

5.5. Routing set size and memory usage

Fig. 12 presents the results obtained for the Routing Set usage metric, which show the average Routing Set size of the network nodes during the performed experiments. The Routing Set usage represents an important factor that can limit network scalability, which is a crucial requirement of several IoT applications.

In the course of conducted studies, LOADng-CTP presented a high number of entries in the Routing Set, once the protocol needed to create a whole path to perform the sending of multicast messages individually to each destination. During the performed experiments, the maximum Routing Set size was previously defined in 16 entries. Thus, when the number of nodes in the multicast group was no higher than the maximum Routing Set size, the

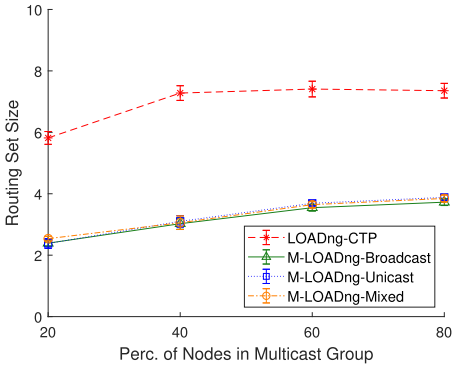


Fig. 12. Routing set usage in function of percentage of nodes in multicast group for LOADng-CTP and M-LOADng with broadcast, unicast, and mixed transmission modes.

LOADng-CTP reached reasonable results. However, when the multicast group size was increased, the protocol performance suffered a considerable reduction, mainly for the multicast data traffic. Thus, as previously presented, the need for constructing paths to nodes replaced in the Routing Set made LOADng-CTP perform new route discovery processes. This effect increases the network overhead, energy consumption, and end-to-end latency, further reducing the packet delivery ratio. In contrast, in the M-LOADng, the introduced RSM helps to reduce the number of entries in the Routing Set and avoids the occurrence of the same negative effect suffered by LOADng-CTP. In the proposed protocol, the reduced number of entries does not affect the forwarding mechanism once the multicast message originator does not need to have a route entry to each multicast destination. During the M-LOADng functioning, each node knows which neighbor is interested in a multicast message, which is a multicast group member, or only forwarder. Thus, the message is forwarded neighbor-by-neighbor until it reaches all of the nodes of the multicast group, using the routes in a "local scope" created by the multicast route discovery process.

To provide improvements for LOADng, both LOADng-CTP and M-LOADng requires increments in the base protocol core. Thus, Table 8 presents memory usage of the studied and implemented solutions over the LOADng core. The memory usage, which is shown in Kilobytes (kB), exhibit the amount of RAM and Flash memory consumed by the whole implemented code to evaluate the proposal. Thus, displayed values include, further to the routing solution, the designed application, MAC protocol, radio drivers, and operational systems. The Flash memory is responsible for storing the read-only code, while the RAM represents the read-write data used during the code execution.

The collected measures for the memory usage metric showed that, as expected, all studied proposals presented an increasing memory requirement compared to the LOADng core. The M-LOADng has required an increase of around 4 kB of Flash memory and 0.5 kB of RAM in comparison with the LOADng core. The proposed solution implementation has involved the creation of whole new multicast route discovery and multicast data forwarding mechanisms. This implementation, although it has an increased source code size, has not required a considerable RAM consumption increasing. The main elements responsible for increased RAM was the route reply mechanism, which required an improved structure for storing M_RREP messages. However, the benefits presented by the proposed solution for all studied scenarios have overcome the additional memory usage requirement. Besides, the extra memory required by the M-LOADng was not a significant problem when considering the memory capacity of devices used in the conducted experiments.

Table 8
Memory usage of LOADng, LOADng-CTP, and M-LOADng with broadcast, unicast, and mixed transmission modes.

Proposal	Flash (kB)	RAM (kB)
LOADng	59.08	11.40
LOADng-CTP	60.88	12.23
M-LOADng-Broadcast	63.01	11.94
M-LOADng-Unicast	63.06	11.95
M-LOADng-Mixed	63.25	11.95

6. Conclusion and future works

This work proposed an improvement for multicast support in the LOADng protocol. The proposal was designed to focus on low power IoT applications, especially those used in industrial scenarios in which sending multicast data represents an important requirement. The proposed M-LOADng introduced a new multicast route discovery mechanism able to build a multicast routing tree that can be used to forward both upward and downward data traffic. The new solution is complemented by reply suppressing and reply retry mechanisms that reduce the control message overhead, decrease the number of entries in the routing table, and ensure reply message transmissions, contributing to a more efficient path creation process. Moreover, the M-LOADng introduces a new multicast forwarding mechanism in which the message transmission is performed neighbor-by-neighbor, using a vision of local forwarding for which fewer Routing Set entries are required. The proposed forwarding mechanism also allows the use of different transmission modes to better fit the requirements of different applications.

The proposed solution, with different forwarding modes, was experimented in a real testbed and compared to the LOADng-CTP, which is currently the most appropriate LOADng improvement to perform this comparison. For the studied scenarios, the number of nodes that joined the multicast group was being increased to study the proposal's scalability. These scenarios also considered different traffic flows. Simple traffic (upward) and multicast traffic (downward) was sent by regular nodes and the sink node, respectively, during a random time interval. Thus, the data traffic generated during the experiments was able to represent a practical application scenario as described in the experiment setup subsection.

The obtained results showed that proposed M-LOADng was able to give a better performance regarding quality-of-service (QoS), network functioning efficiency, and memory usage compared to LOADng-CTP. The new mechanism introduced by the M-LOADng was able to build a multicast routing tree with low overhead and energy consumption. Thus, this feature led the solution to a very consistent performance. Moreover, it was concluded that different forwarding modes mainly affect end-to-end latency. The broadcast forwarding mode presented the lower end-to-end latency values, whereas the use of the unicast mode gave a slightly better packet delivery ratio. The adoption of the mixed mode has demonstrated performance in line with that of the other two methods. Furthermore, it was perceived that the forwarding method was unable to imply an important difference in energy consumption and network overhead.

The main advantage of M-LOADng utilization is the support to send multicast data packets based on a lightweight protocol easily implementable in devices with severe hardware limitations. The proposed solution reached, in some cases, packet delivery rates over 95% for multicast data traffic, demonstrating high reliability and efficiency. Further, the lower latency rates observed with the

use of broadcast mode offers an option to practical applications where the fast packets delivery is mandatory. The proposed M-LOADng also exposed high data throughput rates, satisfactory levels of energy consumption and control overhead efficiency, and stable memory usage. In a practical view, the use of the proposed solution can offer a more reliable and fast communication capacity for IoT environments where the network devices, although tiny and hardware restricted, need to provide a consistent and efficient service to the application layer. Thus, M-LOADng can lead the LOADng to a new level of applicability in real scenarios, offering multicast support required for different IoT scenarios.

As a limitation, the proposed approach stills need to offer a more efficient solution for the maintenance of the created multicast routing trees. The maintenance method currently adopted requires sending periodical control messages to avoid broken the created paths and keep them updated. This issue leads to an extra control overhead that may increase the nodes' energy consumption. Furthermore, the network devices can expend time and processing capacity by handling the additional control messages transmitted during this maintenance process. However, even with this hindrance, M-LOADng is able to offer a more feasible performance when compared to the other studied solution.

For future work, the authors highlight the need for improving the efficiency of route maintenance mechanism previously detached as a limitation of the proposal. Moreover, performing studies considering other multicast protocols that can send different types of messages, not only multicast, and optimizing the proposal deployment for reducing memory usage, are suggested. The execution of new experiments to evaluate the behavior of this proposal in mobile scenarios and adding security modules at the routing level can also be considered.

Acknowledgments

This work has been supported by the Brazilian National Council for Research and Development (CNPq) via Grants No. 201155/2015-0 and 309335/2017-5; by the National Funding from the FCT (Fundação para a Ciência e a Tecnologia) through the UID/EEA/500008/2019 Project; and by the International Scientific Partnership Program ISPP at King Saud University through ISPP #0129.

References

[1] A.H. Alavi, P. Jiao, W.G. Buttlar, N. Lajnef, Internet of things-enabled smart cities: state-of-the-art and future trends, Measurement 129 (2018) 589–606, https://doi.org/10.1016/j.measurement.2018.07.067.

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: a survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutorials 17 (4) (2015) 2347–2376, https://doi.org/10.1109/COMST.2015.2444095.

[3] H. Magsi, A.H. Sodhro, F.A. Chachar, S.A.K. Abro, G.H. Sodhro, S. Pirbhulal, Evolution of 5g in internet of medical things, in: 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2018, pp. 1–7, https://doi.org/10.1109/ICOMET.2018.8346428.

[4] L. Mo, C. Li, Passive uhf-rfid localization based on the similarity measurement of virtual reference tags, IEEE Trans. Instrum. Meas. 68 (8) (2019) 2926–2933, https://doi.org/10.1109/TIM.2018.2869408.

[5] M. Liu, H. Wang, Y. Yang, Y. Zhang, L. Ma, N. Wang, Rfid 3-d indoor localization for tag and tag-free target based on interference, IEEE Trans. Instrum. Meas. (2018) 1–15, https://doi.org/10.1109/TIM.2018.2879678.

[6] S. Sardar, A.K. Mishra, M.Z.A. Khan, Performance evaluation of lte-commsense system for discriminating the presence of multiple objects in outdoor environment, IEEE Trans. Instrum. Meas. (2019) 1–10, https://doi.org/10.1109/TIM.2019.2904332.

[7] M. Bassoli, V. Bianchi, I. De Munari, P. Ciampolini, An iot approach for an aal wi-fi-based monitoring system, IEEE Trans. Instrum. Meas. 66 (12) (2017) 3200–3209, https://doi.org/10.1109/TIM.2017.2753458.

[8] P. Giri, K. Ng, W. Phillips, Wireless sensor network system for landslide monitoring and warning, IEEE Trans. Instrum. Meas. 68 (4) (2019) 1210–1220, https://doi.org/10.1109/TIM.2018.2861999.

[9] X. Lu, J. Liu, H. Zhao, Collaborative target tracking of iot heterogeneous nodes, Measurement 147 (2019) 106872, https://doi.org/10.1016/j.measurement.2019.106872.

[10] G. Shahzad, H. Yang, A.W. Ahmad, C. Lee, Energy-efficient intelligent street lighting system using traffic-adaptive control, IEEE Sens. J. 16 (13) (2016) 5397–5405, https://doi.org/10.1109/JSEN.2016.2557345.

[11] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, W. Zhao, A survey on internet of things: architecture, enabling technologies, security and privacy, and applications, IEEE Internet Things J. 4 (5) (2017) 1125–1142, https://doi.org/10.1109/JIOT.2017.2683200.

[12] G. Dinardo, L. Fabbiano, G. Vacca, A smart and intuitive machine condition monitoring in the industry 4.0 scenario, Measurement 126 (2018) 1–12, https://doi.org/10.1016/j.measurement.2018.05.041.

[13] A.H. Sodhro, F.K. Shaikh, S. Pirbhulal, M.M. Lodro, M.A. Shah, Medical-QoS based telemedicine service selection using analytic hierarchy process, Springer International Publishing, Cham (2017) 589–609, https://doi.org/10.1007/978-3-319-58280-1_21.

[14] A.H. Sodhro, A.S. Malokani, G.H. Sodhro, M. Muzammal, L. Zongwei, An adaptive qos computation for medical data processing in intelligent healthcare applications, Neural Comput. Appl. (Jan 2019), https://doi.org/10.1007/s00521-018-3931-1.

[15] A.H. Sodhro, S. Pirbhulal, Z. Luo, V.H.C. de Albuquerque, Towards an optimal resource management for iot based green and sustainable smart cities, J. Clean. Prod. 220 (2019) 1167–1179, https://doi.org/10.1016/j.jclepro.2019.01.188.

[16] F. Abate, M. Carrat, C. Liguori, V. Paciello, A low cost smart power meter for iot, Measurement 136 (2019) 59–66, https://doi.org/10.1016/j.measurement.2018.12.069.

[17] T. Addabbo, A. Fort, M. Mugnaini, E. Panzardi, A. Pozzebon, V. Vignoli, A city-scale iot architecture for monumental structures monitoring, Measurement 131 (2019) 349–357, https://doi.org/10.1016/j.measurement.2018.08.058.

[18] R. Alexander, A. Brandt, J. Vasseur, J. Hui, K. Pister, P. Thubert, P. Levis, R. Struik, R. Kelsey, T. Winter, RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks, RFC 6550 (2012), https://doi.org/10.17487/RFC6550.

[19] T. Clausen, J. Yi, U. Herberg, Lightweight on-demand ad hoc distance-vector routing - next generation (loadng): protocol, extension, and applicability, Comput. Netw. 126 (2017) 125–140, https://doi.org/10.1016/j.comnet.2017.06.025.

[20] J.V.V. Sobral, J.J.P.C. Rodrigues, R.A.L. Rablo, J. Al-Muhtadi, V. Korotaev, Routing protocols for low power and lossy networks in internet of things applications, Sensors 19 (9) (2019), https://doi.org/10.3390/s19092144.

[21] O. Iova, P. Picco, T. Istomin, C. Kiraly, Rpl: The routing standard for the internet of things...or is it?, IEEE Commun Mag. 54 (12) (2016) 16–22, https://doi.org/10.1109/MCOM.2016.1600397CM.

[22] H. Boyes, B. Hallaq, J. Cunningham, T. Watson, The industrial internet of things (iiot): an analysis framework, Comput. Ind. 101 (2018) 1–12, https://doi.org/10.1016/j.compind.2018.04.015.

[23] M. Abdel-Baset, V. Chang, A. Gamal, Evaluation of the green supply chain management practices: a novel neutrosophic approach, Comput. Ind. 108 (2019) 210–220, https://doi.org/10.1016/j.compind.2019.02.013.

[24] A.H. Sodhro, S. Pirbhulal, V.H.C. de Albuquerque, Artificial intelligence-driven mechanism for edge computing-based industrial applications, IEEE Trans. Indus. Inf. 15 (7) (2019) 4235–4243, https://doi.org/10.1109/TII.2019.2902878.

[25] K. Pister, T. Phinney, P. Thubert, S. Dwars, Industrial Routing Requirements in Low-Power and Lossy Networks, RFC 5673 (2009), https://doi.org/10.17487/RFC5673.

[26] G.G. Lorente, B. Lemmens, M. Carlier, A. Braeken, K. Steenhaut, Bmrf: Bidirectional multicast rpl forwarding, Ad Hoc Netw. 54 (2017) 69–84, https://doi.org/10.1016/j.adhoc.2016.10.004.

[27] L. Junhai, Y. Danxia, X. Liu, F. Mingyu, A survey of multicast routing protocols for mobile ad-hoc networks, IEEE Commun. Surv. Tutorials 11 (1) (2009) 78–91, https://doi.org/10.1109/SURV.2009.090107.

[28] R.C. Biradar, S.S. Manvi, Review of multicast routing mechanisms in mobile ad hoc networks, J. Netw. Comput. Appl. 35 (1) (2012) 221–239, https://doi.org/10.1016/j.jnca.2011.08.003, Collaborative Computing and Applications.

[29] J. Hui, R. Kelsey, Multicast Protocol for Low-Power and Lossy Networks (MPL), RFC 7731 (2016), https://doi.org/10.17487/RFC7731.

[30] G. Oikonomou, I. Phillips, T. Tryfonas, Ipv6 multicast forwarding in rpl-based wireless sensor networks, Wireless Pers. Commun. 73 (3) (2013) 1089–1116, https://doi.org/10.1007/s11277-013-1250-5.

[31] K.Q. Abdel Fadeel, K. El Sayed, Esmrf: enhanced stateless multicast rpl forwarding for ipv6-based low-power and lossy networks, in: Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems, IoT-Sys '15, New York, NY, USA, 2015, pp. 19–24, https://doi.org/10.1145/2753476.2753479.

[32] Y.B. Zikria, M.K. Afzal, F. Ishmanov, S.W. Kim, H. Yu, A survey on routing protocols supported by the contiki internet of things operating system, Future Gener. Comput. Syst. 82 (2018) 200–219, https://doi.org/10.1016/j.future.2017.12.045.

[33] J. Yi, T. Clausen, Collection tree extension of reactive routing protocol for low-power and lossy networks, Int. J. Distrib. Sens. Netw. 10 (3) (2014), https://doi.org/10.1155/2014/352421 352421.

[34] J. Yi, T. Clausen, Y. Igarashi, Evaluation of routing protocol for low power and lossy networks: Loadng and rpl, in: 2013 IEEE Conference on Wireless Sensor (ICWISE), 2013, pp. 19–24, https://doi.org/10.1109/ICWISE.2013.6728773.

[35] J. Yi, T. Clausen, A. Bas, Smart route request for on-demand route discovery in constrained environments, in: 2012 IEEE International Conference on Wireless Information Technology and Systems (ICWITS), 2012, pp. 1–4, https://doi.org/10.1109/ICWITS.2012.6417755.

[36] A. Bas, J. Yi, T. Clausen, Expanding ring search for route discovery in loadng routing protocol, in: Proceedings of The 1st International Workshop on Smart Technologies for Energy, Information and Communication (IW-STEIC), 2012.
[37] J.V.V. Sobral, J.J.P.C. Rodrigues, R.A.L. Rablo, K. Saleem, V. Furtado, Loadng-iot: an enhanced routing protocol for internet of things applications over low power networks, Sensors 19 (1) (2019), https://doi.org/10.3390/s19010150.
[38] G. Li, D. Zhang, K. Zheng, X. Ming, Z. Pan, K. Jiang, A kind of new multicast routing algorithm for application of internet of things, J. Appl. Res. Technol. 11 (4) (2013) 578–585, https://doi.org/10.1016/S1665-6423(13)71565-7.
[39] J. Huang, Q. Duan, Y. Zhao, Z. Zheng, W. Wang, Multicast routing for multimedia communications in the internet of things, IEEE Internet Things J. 4 (1) (2017) 215–224, https://doi.org/10.1109/JIOT.2016.2642643.
[40] M.-S. Pan, S.-W. Yang, A lightweight and distributed geographic multicast routing protocol for iot applications, Comput. Netw. 112 (2017) 95–107, https://doi.org/10.1016/j.comnet.2016.11.006.
[41] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pissard-Gibollet, F. Saint-Marcel, G. Schreiner, J. Vandaele, T. Watteyne, Fit iot-lab: a large scale open experimental iot testbed, in: 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015, pp. 459–464, https://doi.org/10.1109/WF-IoT.2015.7389098.
[42] A.H. Sodhro, Y. Li, M.A. Shah, Energy-efficient adaptive transmission power control for wireless body area networks, IET Commun. 10 (2016) 81–90, https://doi.org/10.1049/iet-com.2015.0368.
[43] A.H. Sodhro, S. Pirbhulal, G.H. Sodhro, A. Gurtov, M. Muzammal, Z. Luo, A joint transmission power control and duty-cycle approach for smart healthcare system, IEEE Sens. J. (2019), https://doi.org/10.1109/JSEN.2018.2881611, 1–1.
[44] A. Dunkels, B. Gronvall, T. Voigt, Contiki – a lightweight and flexible operating system for tiny networked sensors, in: 29th Annual IEEE International Conference on Local Computer Networks, 2004, pp. 455–462, https://doi.org/10.1109/LCN.2004.38.
[45] M. Uwase, M. Bezunartea, J. Tiberghien, J. Dricot, K. Steenhaut, Experimental comparison of radio duty cycling protocols for wireless sensor networks, IEEE Sens. J. 17 (19) (2017) 6474–6482, https://doi.org/10.1109/JSEN.2017.2738700.

Chapter 7

Conclusion and Future Work

This chapter presents the main conclusions of the work performed in the course of this thesis. Besides, it shows important research topics to be considered as future works.

7.1 Final Remarks

This thesis addressed the challenges of the routing protocols for IoT networks. The work identified different limitations of the current routing solutions available in the literature for LLNs and proposed several contributions to overcome or reduce the impact of these weaknesses in the network performance of IoT scenarios.

In the second chapter of this work, a survey reviewing the state-of-the-art considering the main routing solutions for LLNs adopted for IoT networks was presented. The conducted work mainly studied routing improvements introduced to the two primary available routing protocols for LLNs: RPL and LOADng. In the performed work, more than 30 proposals to enhance the routing performance in IoT scenarios were thoroughly studied and their main strengths and weaknesses were highlighted. The study concluded that most new proposals seek to solve problems inherent to support features that are generally weakly attended by the main protocols, such as P2P, mobility, and multicast (P2MP). Further, it was also identified that several works aim to improve routing protocol performance through the increase of QoS and energy efficiency without additional feature support. This survey also identified important existent open issues and detached guidelines to design new routing proposals for IoT. Thus, the main concerns were related to devices heterogeneity, seeking a satisfactory trade-off among network aspects, the creation of proposals with low complexity and reduced memory usage, and the consideration of the IETF's recommendations.

Chapter 3 introduced a framework especially designed for heterogeneous IoT networks composed by low power nodes and RFID elements. In the studied IoT scenario, the existence of wireless sensor nodes equipped with IEEE 802.15.4 communication interface and RFID readers was considered. The named Reader-Sensor Nodes (RSNs) were responsible for RFID tag reading and identification processes, further sending and routing data messages from other network nodes. Based on the limitations that emerged from the integration of these technologies, the proposed framework was designed. The proposed tool was composed by two elements. The first one introduced a new route classification algorithm based on fuzzy systems, while the second presented a novel RFID anti-collision protocol used during the RFID tag reading process. Through experiments conducted to evaluate the performance of the proposed framework, it was concluded the introduced solution can present a better performance when compared to scenarios without the adopted proposal. The framework was able to obtain a tag reading query success rate that was about 25% more efficient and 115% faster than the standard RFID anti-collision protocol. Moreover, due to the consideration of several routing metrics (including the number

of RFID tags in the node area) using fuzzy systems in the process of a best route selection, the introduced framework was able to reduce the packet loss rate by around 75% and improve the network load balance by approximately 55%.

The fourth chapter presented the LOADng-IoT, an improved version of LOADng that allows network nodes to find routes to Internet-connected devices in an autonomous way without a previous definition of a gateway. The proposed solution was designed considering the different node capabilities in IoT scenarios. Due to the deployment cost, it is expected that not all IoT network devices can have a direct connection to the Internet (provided through Wi-Fi, LTE, etc.). Thus, nodes with external connection capability should be responsible for providing an Internet connection to the limited nodes. In this context, the LOADng-IoT introduces a new route discovery mechanism to allow nodes without Internet connectivity to find an Internet-based device without needing a previous gateway configuration. The proposed method allows nodes to create shortened paths, contributing to improving the network performance. Two mechanisms complement LOADng-IoT functioning. The first is a route cache system that is used to reduce the route discovery overhead, and the second is a new error code message adopted to prevent data packet loss when the Internet connection of the Internet-connected node is lost. This improvement was compared to the standard LOADng and LOADng-SmartRREQ in grid, random, and mobility scenarios. The obtained results showed that LOADng-IoT was able to reduce the control message overhead and end-to-end latency, increase packet delivery ratio, and improve network energy efficiency compared to the other studied solutions.

The studies conducted in Chapter 4, which further introduced the LOADng-IoT, showed the limitation of LOADng in mobile IoT networks. To overcome this, Chapter 5 proposed a novel mechanism to enhance LOADng performance in IoT scenarios, as considered and studied in the fourth chapter, but with a higher presence of mobile nodes. The new proposal, named LOADng-IoT-Mob, adapted the route discovery mechanisms from LOADng-IoT, SmartRREQ, and Expanding Ring, to better manage the Routing Set of nodes and avoid the use of paths broken due to nodes' movement. The proposed adaption entails the inclusion of an additional field on LOADng control messages and the Routing Set to allow nodes to store valid time data about their neighbors. The approach also includes a mechanism to harness the control messages exchanged in the network to keep the valid time information of the nodes' neighbors updated. LOADng-IoT-Mob also introduces a new routing metric that allows nodes creating paths considering the distance and reliability of the link among devices. In different scenarios, considering the increasing number of nodes, the percentage of mobile nodes, and the maximum speed of mobile devices, LOADng-IoT-Mob was compared to the standard LOADng and other improved versions, including LOADng-IoT. Obtained results showed that proposed solution was able to achieve better results when compared to different approaches for the metrics packet delivery ratio, latency, control bit overhead per delivered data bit, and energy spent per delivered data bit. Further, it was also noticed that the variation in the nodes' maximum speed, which ranged between 1 m/s and 9 m/s, was unable to significantly affect the performance of the studied solutions. Finally, during a memory usage observation, LOADng-IoT-Mob consumed around 10% more flash memory and 1.55% more RAM compared to the standard LOADng. However, the benefits presented by this proposal, which can sometimes double network performance, compensate for the extra memory required.

Chapter 6 introduced an important improvement for LOADng to support multicast data

message sending and forwarding. The proposal, named M-LOADng, presented a novel route discovery mechanism dedicated to building multicast routing trees. During the multicast tree construction process, the nodes can decide to join the tree based on the multicast group information indicated in the control message. Also, in the course of this process, nodes can opt to suppress control messages to reduce the overhead and number of entries in the routing table. The suppression of these messages does not affect or reduce network efficiency since it is only permitted when a similar message was previously handled. After the tree building process, the multicast message originator node can use three data message transmission modes: broadcast, unicast, and mixed. The proposed M-LOADng, with its different transmission modes, was compared with the LOADng-CTP in a real prototype. The application executed over the network generated P2MP and MP2P data traffic seeking to represent a big set of IoT applications, including an industrial environment. The obtained results showed that regardless of the used transmission mode, M-LOADng could reach a more feasible and stable performance for metrics related to energy and network functioning efficiency, QoS, and memory usage. Furthermore, M-LOADng with unicast transmission mode obtained the best results for packet delivery ratio, while the adoption of the broadcast mode led to better latency performance. The mixed transmission mode presented balanced results between the other two modes.

The main purpose of this thesis and the defined objectives have been successfully attended. Based on the observed limitations and weaknesses of the most relevant routing solutions for IoT networks, new proposals were presented. Specifically, novel solutions were introduced to promote and enhance the support to devices' heterogeneity, multicast (P2MP) packet sending, and mobility. The proposed approaches also allowed significant performance increase for a different number of metrics, such as packet delivery ratio, end-to-end latency, energy efficiency, control message overhead, and memory usage. Thus, it is possible to affirm that the contributions presented in this thesis may lead the routing protocols for LLNs (mainly, the LOADng) to a new level of reliability and efficiency, making them more appropriate and able to fulfill the requirements of IoT applications.

7.2 Future Work

To conclude this research work, based on the found results at routing level for IoT, the following topics are suggested for future research directions:

- To perform the optimization of the proposed solutions to better fit the limited memory resources of low-power devices present in IoT environments;
- To consider the requirements of IoT applications related to security and seek to provide efficient and low-complex solutions at routing level;
- To design smart and context-aware routing solutions to attend the performance needs of real-time IoT applications;
- To propose the use of reliable movement prediction techniques to help the routing protocols better support the topology changes caused by IoT devices movement.

Appendix A

Performance Evaluation of Routing Metrics in the LOADng Routing Protocol

This appendix consists in the following paper:

Performance Evaluation of Routing Metrics in the LOADng Routing Protocol

José V. V. Sobral, Joel J. P. C. Rodrigues, Neeraj Kumar, Chunsheng Zhu, and Raja W. Ahmad

Journal of Communications Software and Systems, Croatian Communications and information Society, in cooperation with FESB, University of Split, Croatia, ISSN: 1845-6421, Vol. 13, No. 2, June 2017, pp. 87-95.

DOI: doi.org/10.24138/jcomss.v13i2.376

Appendix A: Performance Evaluation of Routing Metrics in the LOADng Routing Protocol

JOURNAL OF COMMUNICATIONS SOFTWARE AND SYSTEMS, VOL. 13, NO. 2, JUNE 2017

Performance Evaluation of Routing Metrics in the LOADng Routing Protocol

José V. V. Sobral, Joel J. P. C. Rodrigues, *Senior Member, IEEE*, Neeraj Kumar, *Member, IEEE*, Chunsheng Zhu, and Raja W. Ahmad

***Abstract*— LOADng (Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation) is an emerging routing protocol that emerged as an alternative to RPL (IPv6 Routing Protocol for Low power and Lossy Networks). Although some work has been dedicated to study LOADng, these works do not analyze the performance of this protocol with different routing metrics. A routing metric is responsible for defining values for paths during the route creation process. Moreover, based on these metrics information a routing protocol will select the path to forward a message. Thus, this work aims to realize a performance assessment study considering different routing metrics applied to LOADng. The scenarios under study consider different traffic patterns and network sizes. The routing metrics are evaluated considering the packet delivery ratio, average energy spent per bit delivered, average latency, and number of hops. The results reveals that routing metrics used by this protocol may influence (directly) the network performance.**

***Index Terms*—Internet of Things, LOADng, Low power networks, Performance, Routing metric, Routing protocol.**

I. Introduction

THE concept of Internet of Things (IoT) have emerged with the growing of physical objects connected to the Internet. Predictive studies reveal that the number of interconnected objects through IoT can reach 26 billion until 2020 [1]. The IoT application field is very broad and can cover the sector of industrial manufacturing, energy, transport, e-health, smart cities, agriculture, among others.

Manuscript received February, 25, 2017; revised May, 11, 2017. Date of publication June 1, 2017.

This work has been partially supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) – Brazil through the grant 201155/2015-0, by Instituto de Telecomunicações, Next Generation Networks and Applications Group (NetGNA), Covilhã, Portugal, by National Funding from the FCT – Fundação para a Ciência e a Tecnologia through the UID/EEA/500008/2013 Project, and by Finep, with resources from Funttel, Grant No. 01.14.0231.00, under the *Centro de Referência em Radiocomunicações* - CRR project of the *Instituto Nacional de Telecomunicações* (Inatel), Brazil.

José V. V. Sobral is with the Instituto de Telecomunicações, Universidade da Beira Interior, Portugal and Federal Institute of Maranhão (IFMA), Maranhão, Brazil (e-mail: jose.sobral@it.ubi.pt)

Joel J. P. C. Rodrigues is with the National Institute of Telecommunications (Inatel), Brazil; Instituto de Telecomunicações, Universidade da Beira Interior, Portugal; and University of Fortaleza (UNIFOR), Ceará, Brazil (e-mail: joeljr@ieee.org)

Neeraj Kumar is with the Thapar University, Patiala (Punjab), India (e-mail: neeraj.kumar@thapar.edu)

Chunsheng Zhu is with the The University of British Columbia, Vancouver, Canada (e-mail: chunsheng.tom.zhu@gmail.com)

Raja W. Ahmad is with the University of Malaya, Kaula Lumpur (e-mail: wasimraja@ciit.net.pk)

Digital Object Identifier (DOI): 10.24138/jcomss.v13i2.376

A great set of IoT applications is composed by devices with strict restrictions on energy, processing, and bandwidth. These devices exchange data using wireless communication and create a particular type of network called Low Power and Lossy Networks (LLN). Aiming the use of Internet IPv6 on these kind of networks, the Internet Engineering Task Force (IETF) presented several RFCs (Request for Comments) documents. One of these documents defines the RPL protocol (IPv6 Routing Protocol for Low power and Lossy Networks). Proposed in August 2009, RPL was defined as the standard routing protocol for 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks), in March 2012 [2]. From this date, it is common to consider RPL as the standard routing solution for IoT [3]. Although defined as a standard, different studies have exposed that RPL presents some drawbacks [4], [5]. Thus, considering the existing RPL limitations, novel routing protocols have been proposed. Among these new emerging routing solutions, it is possible to detach the LOADng (Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation) [6], which is a reactive routing protocol designed for LLNs based on the well-know AODV (Ad hoc On-demand Distance Vector routing) [7].

Since the emergence of LOADng as an alternative to RPL, several works have exposed performance assessment studies comparing the two approaches. In [8] the authors compare RPL and LOADng in a home automation scenario with different traffic patterns. In [9] the authors perform the comparison between a reactive (LOADng) and a proactive (RPL) routing protocols in different LLN network topologies. The two protocols are also studied and compared in [10]. Studies considering just LOADng are presented in [11] and [12].

The aforementioned performance studies about LOADng present limitations because they only analyze the default version of the protocol. However, the performance of a routing protocol is influenced by the used routing metrics [13]. A routing metric defines how a routing protocol must compute the weight of each path and select the best route. Thus, based on the necessity of understanding the real potential of the protocol, this work present a performance evaluation study that considers the use of different routing metrics on LOADng. This work differs from the presented in [12] by considering a high variety of routing metrics detaching its importance in the routing performance improvement.

The rest of the document is organized as follows. Section

II presents the features and operation of LOADng. Section III describes the routing metrics considered in this work while the considered simulation scenarios and networks configurations are addressed at Section IV. The obtained results for each routing metric on different networks are detailed in Section V and Section VI concludes the paper and suggests future works.

II. LOADng Routing Protocol

The LOADng is a reactive routing protocol based on AODV that can be used on Low Power and Lossy Networks (LLNs). Similar to AODV, a route is only created by LOADng when two nodes need to exchange a data message between them. Thus, the control messages perform the route creation [14]. Under this process, a set of information stored by nodes during the protocol operation is also used. Then, the following subsections present the LOADng control messages, its information base, and its operation considering the latest version of the protocol presented in [6].

LOADng uses four control messages: Route Request (RREQ), Route Reply (RREP), Route Reply Acknowledgment (RREP_ACK), and Route Error (RERR). RREQ message is always used when a node needs to send a data message to a destination. RREP is used by a destination node that receives an RREQ as an answer to the route request. RREP_ACK is used to answer an RREP message when it requires an acknowledgment. RRER is used when a node fails at the moment of forwarding a data message to the next hop. RERR can also be used when the destination node of a data message is not known by the forwarding node.

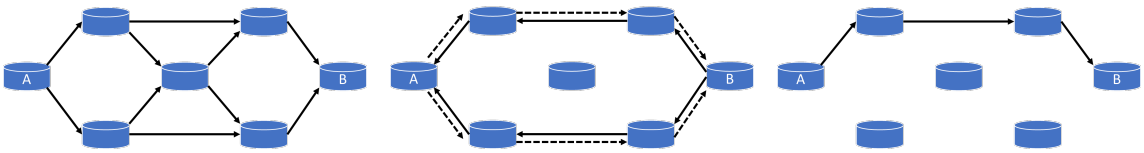
Each node that uses LOADng must maintain an Information Base for controlling the routing processes and the information about the other network nodes. The main elements of the Information Base are the Routing Set, Blacklisted Neighbor Set, and Pending Acknowledgment Set. The Routing Set is composed by a set of routing tuples that stores data about the neighbor nodes, such as the next hop and the number of hops for reaching a destination. Moreover, the valid time of the tuple is stored in the routing set. A routing tuple must be specified as invalid or removed from the Routing Set after its valid time expires. The Blacklisted Neighbor Set stores the nodes address with possible faults that communication has not been possible. Generally, it stores the nodes address that are not able to deliver a required acknowledgment message in a sequence of a communication fault. Each stored address has a valid time that indicates when the information should expire. The Pending Acknowledgment Set records information about the RREP messages that were sent and requires an acknowledgment message (RREP_ACK). Each tuple of the set contains information about the next hop of the RREP, its originator, a sequence number, a flag for indicating if the RREP_ACK was received, and a valid time. If the valid time expires and the RREP_ACK was not received yet, the address of the next hop should be inserted in the Blacklisted Neighbor.

In the LOADng operation, when a node needs to send a data message to a destination, it creates an RREQ message with the same destination of the data message. After creating the RREQ, the node broadcasts the message to all its neighbors. The RREQ is forwarded until reaching its destination (Figure 1a). When the RREQ destination node is reached, an RREP message is created to answer the originator of the RREQ. The created RREP is sent in unicast, hop by hop, using the information recorded at the Routing Set at the moment of RREQ broadcasts. If necessary, the node that receives an RREP can send an RREP_ACK message to the RREP originator (Figure 1b). When the RREP reaches its destination, the node that generated the RREQ will be able to send the data packets using the newly created route (Figure 1c).

Although it may seems simple, the processing of RREQ and RREP messages is composed by a set of verifications that seeks to ensure a good protocol operation avoiding loops. Thus, Figure 2 presents a flowchart that shows as a node processes these messages. After receiving an RREQ/RREP message, first, the node verifies whether it is valid for processing. Among others, the validation process verifies if the length of the address (in the message) is different from the length of the receiver node and if the sequence number of the message is lower than the previously received by the message originator. If one of these conditions is true, the message must be dropped. If the message is valid, the used routing metric is computed and updated. Following, the node verifies the existence of a routing tuple for the message originator inside the Routing Set. If a matching tuple is not found, a new routing tuple is created and inserted in the Routing Set. In the next step, the created or found routing tuple is compared with some field of the message to verify if the tuple will be refreshed and if the message will be considered for forwarding. If the expected conditions are not attended, the message is discarded and it is neither considered for forwarding nor refreshing. However, if the conditions are attended, the fields of the message and the refreshed routing metric are used for refreshing the routing tuple. In the next step, the nodes should verify the type of message. If it is an RREQ message, the node verifies if it is the destination. If true, a new RREP message is generated with the originator of the RREQ as a destination. Otherwise, the message is forwarded. On the other hand, if the message is an RREP, the node must verify if an acknowledgment message is required. Then, if true, an RREP_ACK message must be sent to the previous hop of RREP message. Finally, the node verifies if it is the destination of the RREP message. If yes, the process of route creation is finished and the data packets are able to be sent. If not, the message is forwarded to the next hop.

It is also important to note that, for each message received, the field of hop count is incremented and the field of hop limit is decremented. Optionally, if another metric type is used, the field with the value of routing metric may also be updated. Thus, all forwarded messages must be sent with these updated values. It is important to mention that a RREP_ACK message is sent in unicast and cannot be forwarded. A node that receives a RREP_ACK processes it and, after, it is discarded.

LOADng routing protocol is fully based on the AODV. However, some aspects are simplified in order to reduce the protocol complexity and the quantity of computational resources required to execute it. One of the main features refers that only the destination node of an RREQ can answer



(a) RREQ transmission (b) RREP and RREP_ACK (dashed line) transmission (c) Data transmission

Fig. 1: LOADng operation and use of control messages.

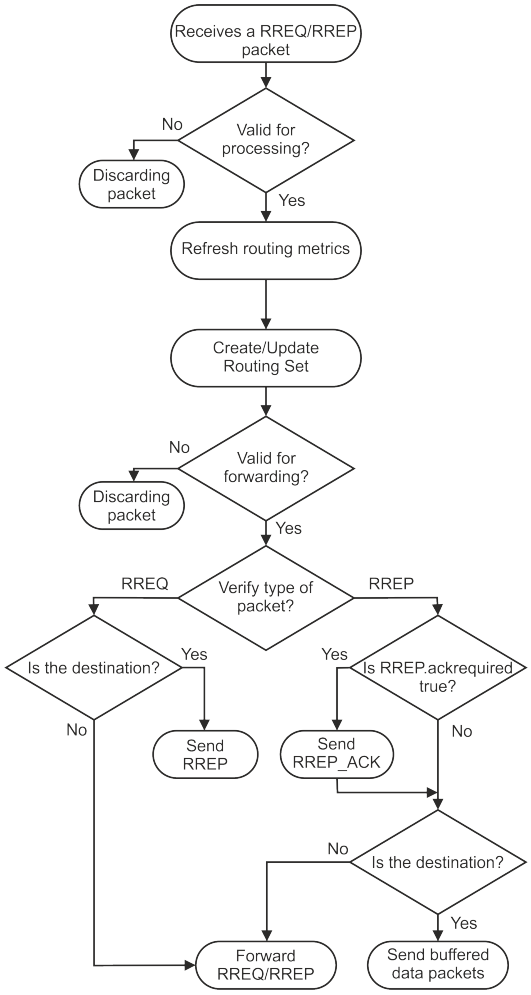


Fig. 2: Illustration of the LOADng Flowchart.

the request with an RREP message. In addition, the nodes do not maintain a list with the address of precursor nodes. In summary, among other features of LOADng, it is possible to highlight the following:

- It supports different lengths of addresses (e.g. IPv6 or IPv4);
- It does not use periodical control messages (e.g., HELLO messages of the AODV);
- It supports the use of different routing metrics optionally to the default hop count.

The possibility of using alternative routing metrics in the LOADng is the primary focus of this work. In the next Section some alternatives to the default hop count that can be used to enhance the routing performance are described.

III. Routing Metrics

By default, the LOADng protocol uses the hop count metric for selecting the path (the shortest path, in this case) between two nodes. However, as above-mentioned, it is possible to use different information for computing the weight of the routes. The performance of a routing protocol is strongly related to the routing metric under use. During the route creation process (transmission of RREQs and RREPs), the LOADng uses information of the control messages or calculated at the moment of the signal received for computing the weight of each path according to the routing metric. The calculated values are used to update the routing table with the best route to a destination. After that, these values are forwarded to the other nodes inside of control messages. Thus, the routing table stores the best route to a destination based on the routing metric information.

The available routing metrics are extremely diverse and are categorized into two types: node metrics and link metrics [15]. Node metrics consider aspects related to the node, as processing capacity, remaining energy, etc. On the other hand, the link metrics address information about the connection among nodes, such as delay, link quality, and throughput [16]. The following subsections introduce different link and node metrics that are used with LOADng in this work.

A. Hop Count

The Hop Count (HC) metric is one of the most commonly used routing metrics. It is the default routing metric of LOADng. HC represents the number of times that a message was sent until reach the destination [17]. Each link in a route counts as one unity without considering neither node nor link characteristics. In an “optimal” scenario (without interference, noises, collision, and energy restrictions), HC can represent the best path due to the use of a small number of transmissions resulting in a low latency and low energy consumption. However, in a real scenario, sometimes, the shortest path may not be the best. Wireless communications commonly suffer from noises and interference that can reduce the link quality among nodes causing packets loss. Thus, the

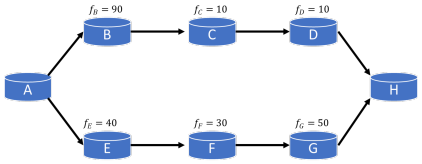


Fig. 3: Example of network composed by nodes with different remaining energies.

HC can not represent the best route since does not consider node nor link quality aspects.

B. Minimum Battery Cost Routing (MBCR)

The Minimum Battery Cost Routing (MBCR) is a routing metric based on the nodes energy. The MBCR considers the remaining battery capacity of the node for calculating the best path between two nodes [18]. The strategy of the MBCR is to avoid the routes with low remaining energy aiming to reduce the packet loss and decrease the power consumption of the overall network. Thus, the cost of each node n that composes a route r is computed by $f_n(E_n)$, where f_n is a battery cost function and E_n is the current remaining node battery. The f_n can be computed using Equation 1.

$$f_n(E_n) = \frac{1}{E_n} \tag{1}$$

The total cost of a route r is computed with the sum of all $f_n(E_n)$ (Equation 2). As the MBCR is a minimized metric, the routing protocol should select the route with the minimum total cost.

$$\sum_{n \in r} f_n(E_n) \tag{2}$$

Although MBCR may provide a network load balancing, it can select routes with low remaining energy nodes. Figure 3 exposes an example where the route A-B-C-D-H is chosen because it presents the lower total sum. However, node B has a very low remaining battery and probably should break the route rapidly.

C. Min-Max Battery Cost Routing

Similar to MBCR, the Min-Max Battery Cost Routing (MMBCR) is also a routing metric based on the node energy. However, to solve the main fault of MBCR, the idea of MMBCR is to avoid the use of a route when the nodes have low remaining battery.

Considering that high values obtained with $f_n(E_n)$ represent a low remaining node battery, the idea of MMBCR is to choose the path with the minimum function cost. Different to MBCR, the cost of each path is represented by the maximum value of $f_n(E_n)$ among the nodes that compose a route. Thus, Equation 3 represents the function to obtain the best path r'.

$$r' = \underset{r \in R}{min}(\underset{n \in r}{max}(f_n(E_n))) \tag{3}$$

For example, considering the routes of Figure 3, the cost of route A-B-C-D-H computed by MMBCR is 90 (node B) because it is the higher $f_n(E_n)$ value along the path. Likewise, the cost of route A-E-F-G-H defined by MMBCR is 50 (node G). To choose the best route (r'), MMBCR gets the path with the minimum cost among those available ones, in this case, the route A-E-F-G-H.

The approach presented by MMBCR may avoid the use of routes composed by nodes with low remaining energy and increase the network lifetime (when the lifetime of the first dead node is considered as the network lifetime). However, the use of MMBCR may not ensure a good network performance. Since that does not take into account information about the link quality among nodes, the selected best route may have nodes with communication faults provoking the packet loss.

D. LQI Weaklinks

The Link Quality Indicator Weaklinks (LQI_WL) is a routing metric based on the quality of communication among nodes. The LQI is a real value provided by the physical layer of the standard IEEE 802.15.4. This value, which ranging between 0 (worst) and 255 (best), is computed by a node every time that it receives a message. The calculated value is highly dynamic and may change due to several factors. Thus, the LQI value computed at the transmission $A \rightarrow B$ almost never is equal to the value calculated at the transmission $B \rightarrow A$.

The LQI represents the quality of communication between two nodes (point-to-point). Thus, using the LQI to measure the quality of an end-to-end route, the LQI_WL uses the WeakLinks role. The WeakLinks approach uses a threshold to distinguish links between bad or good. During the route creation process, each node verifies if the computed LQI value is lower than the previously defined threshold (LQI_{th}). If positive, the WeakLinks counter is increased by 1, else the WeakLinks counter does not change. In this way, the best route between a sender and a destination node is the one with the lower number of WeakLinks among those available ones.

E. MAX-LQI

The MAX-LQI is a routing metric based on LQI, such as LQI_WL. Working like MMBCR, the MAX-LQI aims to choice the path with the best worst link. Thus, the worst LQI value of the links that composes a route represents the path cost. The routing protocol must select the path with the best cost (maximum LQI) among those available ones.

Since it considers just the worst LQI value of a path, MAX-LQI may not select a route with a low number of hops. As a consequence, the number of transmissions used to deliver a message to its destination may be high and the spent energy increased [19].

F. ETX

The Expected Transmission Count (ETX) [20] is one of the most used routing metrics in LLNs. With a cross-layer approach, ETX determines the expected number of transmissions that a node should perform for the message in order

to reach its destination successfully. To compute ETX, the nodes store information about the MAC layer packets sent to their neighbors and the acknowledgment packets received. Therefore, it is possible to calculate the number s of packets sent successfully and the number f of packets lost (without receiving acknowledgment). Thus, Equation 4 presents the ETX computation of the link between the nodes i and j [13].

$$ETX_{i,j} = \frac{s+f}{s} > 0 \quad (4)$$

Similar to LQI, ETX defines a weight for a link where the ETX of $A \rightarrow B$ may not be equals to the ETX of $B \rightarrow A$. Thus, the total cost of a path is represented by the sum of all ETX values of the links that compose the route.

The ETX is a minimized routing metric that, for being additive, considers the number of hop among the nodes implicitly. Hence, the use of ETX allows the routing protocol to select a short path composed by reliable links. In contrast, the information stored by nodes to compute the ETX value of its neighbors may exhaust its limited memory resource and restrict the network scalability.

IV. Performance Evaluation

This work aims to evaluate the performance of primary routing metrics applied to the LOADng routing protocol. The performance assessment study was conducted through simulation using Castalia [21]. Simulations considering two IoT applications with MP2P and P2P traffic pattern were performed. The MP2P traffic pattern is characterized by the network data traffic flows from the nodes to a central unit (sink or gateway). On the other hand, in P2P traffic pattern, the network data traffic is directly exchanged among nodes [3].

In the simulated MP2P application, a sink node was placed at the center of the simulation area for receiving messages sent from the other network nodes. In the P2P application, just one node sent messages to a receiver node. The sender was located at the bottom-right of the network, and the receiver was placed at the top-left of the network. In both applications the nodes were static. Table I exposes the simulations parameters used in this study.

TABLE I: Simulation parameters.

Simulation parameters	
Parameter	Value
SimTime	600s
Initial Energy	20 J
Application	MP2P and P2P
Routing	LOADng
Mac Protocol	802.15.4
Radio	CC2420
Data message rate	0.5 msg/s
Numbe of nodes	16, 36, 64
Simulation area (m^2)	50, 50, 100
Network deployment	4x4, 6x6, 8x8 grids
Packet length	
Type of packet	Lenght
RREQ	240 bits
RREP	272 bits
RERR	240 bits
RREP_ACK	144 bits
Data Packet Size	512 + 64 (overhead) bits

The performance assessment study was conducted considering four evaluation metrics: packet delivery ratio, average latency, average spent energy per bit, and number of hops. These metrics are described as follows:

- **Packet delivery ratio**: represents the quantity of data messages that were delivered to their destination node successfully. A low delivery ratio exposes a fault network efficiency and a limited quality of service. The used routing metric can affect the packet delivery ratio directly since the forwarded messages through routes composed by links with low quality may cause packet loss.
- **Average spent energy per bit**: represents the amount of energy that network consumes to deliver each data bit successfully. The metric value is obtained through the ratio of the amount of spent energy by the network and the quantity of data bits received by the nodes. Great values of the averaged spent energy per bit show that network is using a high amount of energy to deliver few data messages. Thus, the routing metric should allow that routing protocol optimizes the route selection process to ensure a high packet delivery ratio with an efficient power consumption.
- **Average latency**: measures the time spent by the network to deliver a data message to its destination. Several aspects can contribute for a low latency such as the physical distance between the sender and the destination node, the nodes workload, the quality of the links, etc. The routing protocol, based on the used routing metric, should avoid paths with a high number of hops and low link quality for trying to ensure an acceptable average latency able to attend the application requirements.
- **Number of hops**: exposes the number of times that a data message was forwarded until reaching its destination. A high number of hops implies a significant number of the message forwarding and, consequently, a high energy consumption. Hence, it is important that the routing protocol uses short paths but without ignoring the others route aspects as link quality and nodes energy.

Based on the performed experiments using the above-described scenarios and corresponding parameters, next section is dedicated to the results analyses obtained for the four evaluation metrics.

V. Results Analysis

A. MP2P Application

Figure 4 depicts the results obtained for the packet delivery ratio metric. In the small network (16 nodes), all the routing metrics can deliver around 90% of the sent packets. However, with the network growth, the routing metrics have a considerable performance reduction due to the increment of the control and data packets traffic. In a scenario where all the nodes send messages to just a central unit, increasing the number of nodes means congestion of the nodes close to the sink and decrease the packet delivery ratio apart the routing metric under use.

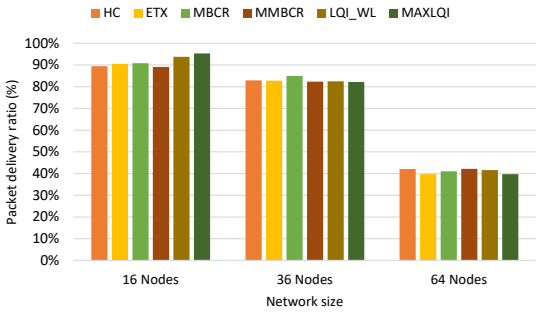


Fig. 4: Packet delivery ratio in the MP2P scenario.

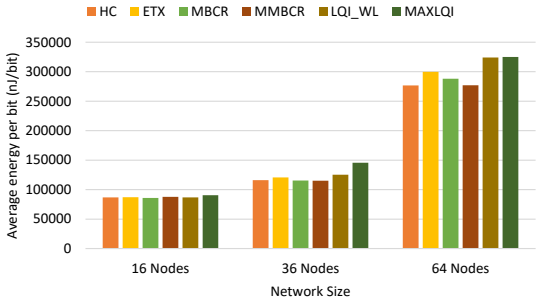


Fig. 5: Average spent energy per bit in the MP2P scenario.

Figure 5 shows the results obtained for the average spent energy per bit metric. The results present that, for all the studied metrics, the amount of energy spent to deliver data packets is lower in small networks. In the network with 64 nodes, the data packets require a higher number of forwarding messages to reach its destination thus, as consequence of the high radio usage, the consumed energy is increased. Hence, the high values of average spent energy per bit are justified due to the low packet delivery ratio (as may be seen in Figure 4) and the high power consumption. It is also possible to note that in the greater studied network, the link quality based routing metrics have the worst average spent energy per bit due to the use of routes with a big number of hops (Figure 7). Still considering the 64 nodes network, the MMBCR was able to ensure the better performance using less energy and an acceptable packet delivery ratio.

The results for the average latency metric are presented in Figure 6. As may be seen, the quantity of packets (in percentage) delivered on each latency interval (represented in ms), e.g. five percent of the data messages delivered using MAXLQI have reached its destination with a latency between 100 and 150 milliseconds. For all the studied networks, the link quality based routing metrics presents the worst latency performances. The operation mode of these metrics (previously presented) seek to select routes with the best link quality without considering the distance (in hops) between nodes. This feature allows the routing protocol to forward messages through paths with a high number of hops increasing the average latency. Note that high latency is not directly related to a packet loss. Although the link quality based routing metrics have bad results for average latency, they can provide a packet delivery ratio with similar or better performance when compared with other studied routing metrics.

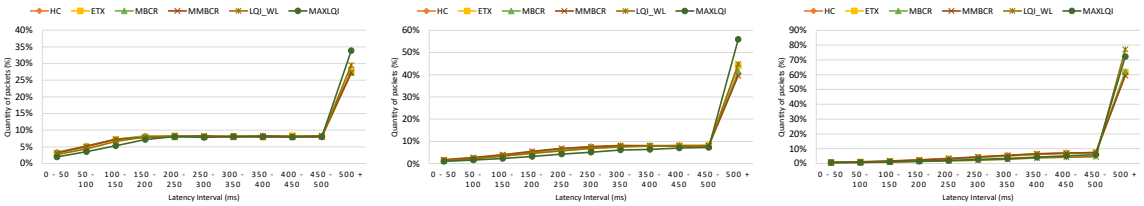
Figure 7 presents the results for the number of hops metric. In these figures, each color inside bars represents each number of hops value. The size of the color representation in the bar shows the quantity of packets (in percentage) delivered with that number of hops. For example, for the MBCR routing metric in the network with 16 nodes, about 60% of the packets were delivered using one hop, 30% were delivered using two hops, and 10% with three hops. The results expose that, in the networks with 16 and 36 nodes, the HC, ETX, MBCR, and MMBCR deliver more than 50% of their packets using just one hop. In contrast, the LQI_WL and MAXLQI, when compared with the other routing metrics, have presented the use of bigger paths to reach the message destination in all the studied networks. As already explained, the search for paths with better link quality can raises the use of greater paths causing high energy consumption and long average latency.

B. P2P Application

The results for the packet delivery ratio metric in the P2P scenario are available at Figure 8. According to these results, the link quality based routing metrics have a better performance compared to the other metrics in the network with 16 and 36 nodes. However, the MAXLQI suffer a significant performance decrease with the network growth exposing its low scalability in the P2P scenario. These results show that the strategy of using just one LQI value to represent the quality of a route can fail when the paths are composed by various links. On the contrary, the LQI_WL was able to maintain a high performance even with the increment of the number of nodes in the network. The strategy to count the number of bad links allows the routing protocol to avoid the use of paths with a big number of weak links implying the packet delivery ratio increase. The results obtained by ETX are very close to the results of HC. ETX is an additive metric in function of the messages exchanged by the nodes at the MAC layer. Hence, when the information extracted from the link layer are few, ETX metric will operate just a “hop count” metric.

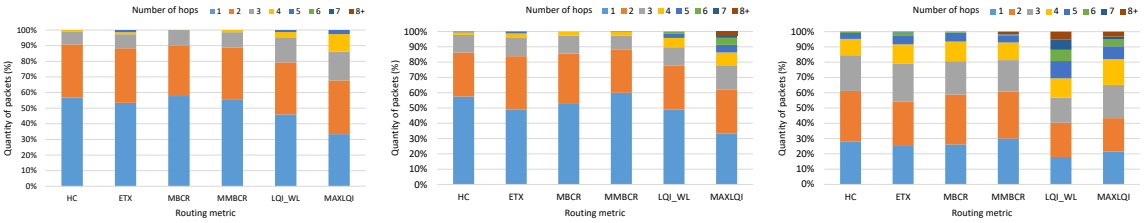
The results for the average energy spent per bit are exposed in Figure 9. According to the simulation experiments, in many scenarios, the link quality based routing metrics have better results, excepted in the network with 64 nodes where the MAXLQI had the lower performance compared to the other routing metrics. This worst performance of MAXLQI is a consequence of the low packet delivery ratio and the high energy consumption produced by the use of routes with a big number of hops. The energy based metrics were able to keep a low power consumption. However, its low packet delivery ratio implied high values of spent energy per delivered bit. In all studied network sizes, LQI_WL, although presenting a power consumption higher than the obtained by the energy based metrics, was able to obtain good results due to high packet delivery ratio.

Figure 10 depicts the average latency for the P2P application. The results show that, in the majority of the studied scenarios, MAXLQI has the higher average latency. Although



(a) 16 nodes (b) 36 nodes (c) 64 nodes

Fig. 6: Average latency in the MP2P scenario.



(a) 16 nodes (b) 36 nodes (c) 64 nodes

Fig. 7: Number of hops in the MP2P scenario.

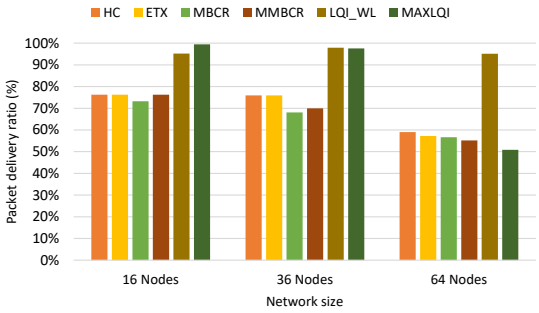


Fig. 8: Packet delivery ratio in the P2P scenario.

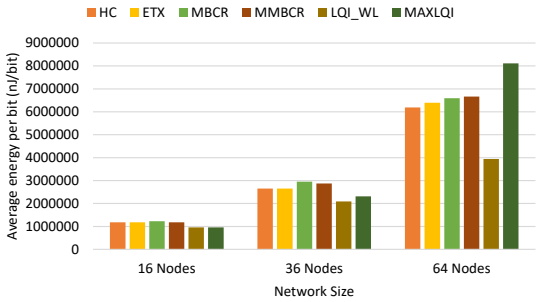


Fig. 9: Average spent energy per bit in the P2P scenario.

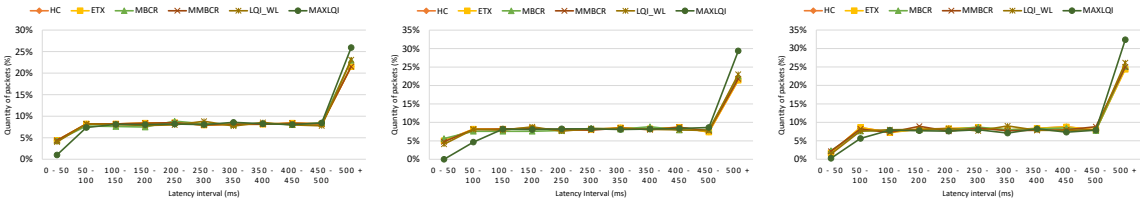
LQI_WL used routes with a high number of hops (presented in Figure 11), the metric stills able to achieve a latency close to the faster performed metrics. Although the number of hops may influence the average latency, this result shows that it is not the single factor that can contribute to increase the time needed to the message till reaching its destination.

The results for the number of hops metric may be seen at Figure 11. According to the simulation results, the metrics based on link quality use routes with a high number of hops if compared with the other routing metrics, as expected. These results can directly affect the power consumption but they are not determinant for a packet delivery ratio reduction (8). Due to the distance between the sender and the receiver nodes, the number of hops used to deliver the messages is higher than in the MP2P scenario. In the network with 64 nodes, the minimum number of hops used by MAXLQI was five. In contrast, the maximum number of hops used by HC, ETX, and MBCR was four. Moreover, observing the results, it is possible to perceive that, although the network has static nodes, the distance (in hops) between the nodes may change. This effect may be caused by control messages loss at the moment of a route creation. Another justification for this behavior is the LOADng functioning. The routing protocol uses sequence numbers on its control messages that can force a node to update its routing table with a new value even the prior route being better than the current one.

VI. Conclusion and Future Works

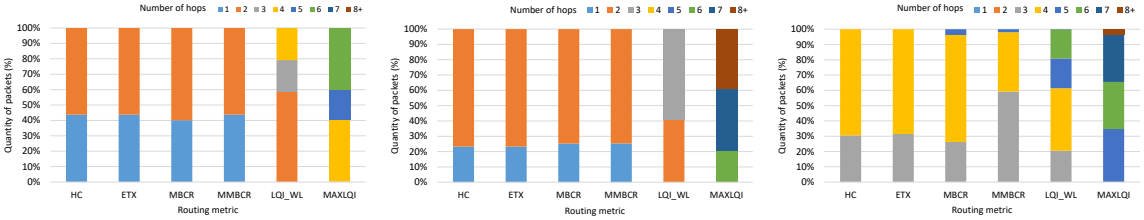
This work presented a performance assessment study considering several routing metrics applied to the LOADng routing protocol. LOADng is a routing protocol that has been emerged as an alternative to fulfil the drawbacks identified on the standard RPL (in LLNs). From its creation, the default version of LOADng has been studied in different scenarios and applications. However, few works are dedicated to study the impact of using different routing metrics with this protocol.

In this paper, five different routing metrics (ETX, MBCR, MMBCR, LQI_WL and MAXLQI) and the default routing metric of LOADng (HC) were studied in network scenarios with MP2P and P2P traffic patterns. The obtained results show that link based routing metrics were able to provide



(a) 16 nodes (b) 36 nodes (c) 64 nodes

Fig. 10: Average latency in the P2P scenario.



(a) 16 nodes (b) 36 nodes (c) 64 nodes

Fig. 11: Number of hops in the P2P scenario.

a high packet delivery ratio due to the use of most reliable paths. In contrast, these metrics have shown that, in some cases, the selection of reliable paths can use a high number of hops and cause a power consumption increase. Moreover, the use of great routes may cause a high average latency decreasing the quality of service. The ETX and the energy based routing metrics present results that, in most cases, are close to the default HC. However, the performed simulations are not enough to discard the potential of these metrics.

In an overview of the obtained results for the evaluated scenarios, LQI_WL revealed the most reliable performance when compared to the other approaches. However, the selection of a routing metric should be made considering the network requirements and the application objectives. As future work, the authors intend to extend this work considering more routing metrics and application scenarios. The authors also plan to discuss the creation of new metrics considering the ones available in the literature aiming to perform a detailed study of LOADng.

References

[1] R. Want, B. N. Schilit, and S. Jenson, "Enabling the internet of things," *Computer*, vol. 48, no. 1, pp. 28–35, 2015.

[2] T. Winter, A. Brandt, J. Hui, R. Kelsy, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "Rpl: Ipv6 routing protocol for low-power and lossy networks," Internet Requests for Comments, RFC Editor, RFC 6550, 2012.

[3] O. Iova, G. P. Picco, T. Istomin, and C. Kiraly, "Rpl, the routing standard for the internet of things... or is it?" *IEEE Communications Magazine*, vol. 17, 2016.

[4] C. H. Barriquello, G. W. Denardin, and A. Campos, "A geographic routing approach for ipv6 in large-scale low-power and lossy networks," *Computers & Electrical Engineering*, 2015.

[5] H. Fotouhi, D. Moreira, and M. Alves, "Mrpl: Boosting mobility in the internet of things," *Ad Hoc Networks*, vol. 26, pp. 17–35, 2015.

[6] T. Clausen, J. Yi, A. Niktash, Y. Igarashi, H. Satoh, U. Herberg, C. Lavenu, T. Lys, and J. Dean, "The lightweight on-demand ad hoc distance-vector routing protocol-next generation (loadng)," Working Draft, IETF Secretariat, Internet-Draft draft-clausen-lln-loadng-14.txt, 2016.

[7] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing," Internet Requests for Comments, RFC Editor, RFC 3561, 2003.

[8] M. Vučinić, B. Tourancheau, and A. Duda, "Performance comparison of the rpl and loadng routing protocols in a home automation scenario," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2013, pp. 1974–1979.

[9] J. Tripathi, J. C. De Oliveira, and J.-P. Vasseur, "Proactive versus reactive routing in low power and lossy networks: Performance analysis and scalability improvements," *Ad Hoc Networks*, vol. 23, pp. 121–144, 2014.

[10] J. Yi, T. Clausen, and Y. Igarashi, "Evaluation of routing protocol for low power and lossy networks: Loadng and rpl," in *2013 IEEE Conference on Wireless Sensor (ICWISE)*. IEEE, 2013, pp. 19–24.

[11] S. Elyengui, R. Bouhouchi, and T. Ezzedine, "Loadng routing protocol evaluation for bidirectional data flow in ami mesh networks," *Int. Journal of Emerging Technology and Advanced Engineering*, 2015.

[12] J. V. Sobral, J. J. Rodrigues, K. Saleem, and J. Al-Muhtadi, "Performance evaluation of loadng routing protocol in iot p2p and mp2p applications," in *International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*. IEEE, 2016, pp. 1–6.

[13] P. Karkazis, P. Trakadas, H. C. Leligou, L. Sarakis, I. Papaefstathiou, and T. Zahariadis, "Evaluating routing metric composition approaches for qos differentiation in low power and lossy networks," *Wireless networks*, vol. 19, no. 6, pp. 1269–1284, 2013.

[14] T. Clausen, J. Yi, C. Lavenu, A. Lys, A. Niktash, Y. Igarashi, and H. Satoh, "The lln on-demand ad hoc distance-vector routing protocol-next generation (loadng)," Working Draft, IETF Secretariat, Internet-Draft draft-clausen-lln-loadng-00.txt, 2011.

[15] J.-P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing metrics used for path calculation in low-power and lossy networks," Internet Requests for Comments, RFC Editor, RFC 6551, 2012.

[16] A. Brachman, "Rpl objective function impact on llns topology and performance," in *Internet of Things, Smart Spaces, and Next Generation Networking*. Springer, 2013, pp. 340–351.

[17] Y. Yang, J. Wang, and R. Kravets, "Designing routing metrics for mesh networks," in *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[18] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 138–147, 2001.

[19] C. Gomez, A. Boix, and J. Paradells, "Impact of lqi-based routing metrics on the performance of a one-to-one routing protocol for ieee 802.15.

4 multihop networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, p. 6, 2010.
[20] D. S. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.
[21] A. Boulis *et al.*, "Castalia: A simulator for wireless sensor networks and body area networks," *NICTA: National ICT Australia*, 2011.

José V. V. Sobral (jose.sobral@it.ubi.pt) is currently a Ph.D. student at the University of Beira Interior (UBI), Covilhã, Portugal. He received his M.Sc. degree in Computer Science from the Federal University of Piauí (UFPI), Teresina, Brazil, and B.S. degree in Computer Science from the *Centro de Ensino Unificado de Teresina* (CEUT), Teresina, Brazil. José Sobral is an assistant professor at the Federal Institute of Maranhão (IFMA), São Luís, Brazil, and a member of NetGNA Research Group. His research interests include Internet of Things (IoT), routing protocols for low power and lossy networks, wireless sensors networks, RFID systems, and computational intelligence.

Joel J.P.C. Rodrigues (joeljr@ieee.org) [S'01, M'06, SM'06] is a professor and senior researcher at the National Institute of Telecommunications (Inatel), Brazil and senior researcher at the Instituto de Telecomunicaes, Portugal. He has been professor at the University of Beira Interior (UBI), Portugal and visiting professor at the University of Fortaleza (UNIFOR), Brazil. He received the Academic Title of Aggregated Professor in informatics engineering from UBI, the Habilitation in computer science and engineering from the University of Haute Alsace, France, a PhD degree in informatics engineering and an MSc degree from the UBI, and a five-year BSc degree (licentiate) in informatics engineering from the University of Coimbra, Portugal. His main research interests include e-health, sensor networks and IoT, vehicular communications, and mobile and ubiquitous computing. Prof. Joel is the leader of NetGNA Research Group (http://netgna.it.ubi.pt), the President of the scientific council at ParkUrbis Covilh Science and Technology Park, the Past-Chair of the IEEE ComSoc Technical Committee on eHealth, the Past-chair of the IEEE ComSoc Technical Committee on Communications Software, Steering Committee member of the IEEE Life Sciences Technical Community and Publications co-Chair, and Member Representative of the IEEE Communications Society on the IEEE Biometrics Council. He is the editor-in-chief of the International Journal on E-Health and Medical Communications, the editor-in-chief of the Recent Advances on Communications and Networking Technology, the editor-in-chief of the Journal of Multimedia Information Systems, and editorial board member of several high-reputed journals. He has been general chair and TPC Chair of many international conferences, including IEEE ICC, GLOBECOM, and HEALTHCOM. He is a member of many international TPCs and participated in several international conferences organization. He has authored or coauthored over 500 papers in refereed international journals and conferences, 3 books, and 2 patents. He had been awarded several Outstanding Leadership and Outstanding Service Awards by IEEE Communications Society and several best papers awards. Prof. Rodrigues is a licensed professional engineer (as senior member), member of the Internet Society, an IARIA fellow, and a senior member ACM and IEEE.

Neeraj Kumar (neeraj.kumar@thapar.edu) received his Ph.D. in CSE from SMVD University, Katra (J & K), India, and was a postdoctoral research fellow in Coventry University, Coventry, UK. He is working as an Associate Professor in the Department of Computer Science and Engineering, Thapar University, Patiala (Pb.), India since 2014. Dr. Neeraj is an internationally renowned researcher in the areas of VANET & CPS Smart Grid & IoT Mobile Cloud computing & Big Data and Cryptography. He has published more than 150 technical research papers in leading journals and conferences from IEEE, Elsevier, Springer, John Wiley. His paper has been published in some of the high impact factors journals such as-IEEE Transactions on Industrial Informatics, IEEE Transactions on Industrial Electronics, IEEE Transactions on Information Forensics and Security, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Power Systems, IEEE Transactions on Vehicular Technology, IEEE Systems Journal, IEEE Wireless Communication Magazine, IEEE Vehicular Technology Magazine, IEEE Communication Magazine, IEEE Networks Magazine etc. Apart from the journals conferences, he has also published papers in some of the core conferences of his area of specialization such as-IEEE Globecom, IEEE ICC, IEEE Greencom, IEEE CSCWD. He has guided many research scholars leading to Ph.D. and M.E./M.Tech. His research is supported by funding from TCS, CSIT, UGC and UGC in the area of Smart grid, energy management, VANETs, and Cloud computing. He is member of the cyber-physical systems and security research group. He has research funding from DST, CSIR, UGC, and TCS. He has total research funding from these agencies of more than 2 crores. He has got International research project under Indo-Poland and Indo-Austria joint research collaborations in which teams from both the countries will visit to Thapar University, Patiala and Warsaw University, Poland, University of Innsburg, Austria respectively. He has h-index of 25 (according to Google scholar, March 2017) with 2500 citations to his credit. He is editorial board members of International Journal of Communication Systems, Wiley, and Journal of Networks and Computer Applications, Elsevier. He has visited many countries mainly for the academic purposes. He is a visiting research fellow at Coventry University, Coventry, UK. He has many research collaboration with premier institutions in India and different universities across the globe. He is a member of IEEE.

Chunsheng Zhu (chunsheng.tom.zhu@gmail.com) received the Ph.D. degree in Electrical and Computer Engineering from The University of British Columbia, Canada. He is currently a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, The University of British Columbia, Canada. He has authored over 100 papers published or accepted by refereed international journals/magazines (e.g., IEEE Transactions on Industrial Electronics, IEEE Transactions on Computers, IEEE Transactions on Information Forensics and Security, IEEE Transactions on Emerging Topics in Computing, IEEE Systems Journal, IEEE Access, and IEEE Communications Magazine) and conferences (e.g., IEEE Globecom and IEEE ICC). His current research interests mainly include wireless sensor networks, cloud computing, Internet of Things, social networks, and security.

Raja W. Ahmad (wasimraja@ciit.net.pk) is an Assistant professor at COMSATS Institute of Information Technology, Pakistan. He did his PhD in Computer Science from University of Malaya under Bright Spark Scholarship program. He started his carrier as a computer student back to 2003 by choosing computer science as major during undergraduation course from University of Azad Jammu & Kashmir, Muzaffarabad. In addition, he did his masters from COMSATS Institute of Information Technology, Abbottabad, Pakistan under "COMSATS Merit Scholarship" program. His research interests include, mobile application energy profiling, energy efficient computational offloading, cloud resource allocation, VM migration, Network performance, Application's QoS on low bandwidth networks, and energy efficient cloud data centers.

Appendix A: Performance Evaluation of Routing Metrics in the LOADng Routing Protocol

Appendix B

ERAOF: A New RPL Protocol Objective Function for Internet of Things Applications

This appendix consists in the following paper:

ERAOF: A New RPL Protocol Objective Function for Internet of Things Applications

Natanael Sousa, José V. V. Sobral, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, and Petar Solic

2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech 2017): Split, Croatia, July 12 - 14, IEEE, ISBN: 978-953-290-071-2, 2017.

ERAOF: A New RPL Protocol Objective Function for Internet of Things Applications

Natanael Sousa[1], José V. V. Sobral[2,3], Joel J. P. C. Rodrigues[2,4,5], Ricardo A. L. Rabêlo[1], and Petar Solic[6]
[1]Federal University of Piauí (UFPI), Teresina - PI, Brazil
[2]Instituto de Telecomunicações, Universidade da Beira Interior, Portugal
[3]Federal Institute of Maranhão (IFMA), São Luís - MA, Brazil
[4]National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí - MG, Brazil
[5]University of Fortaleza (UNIFOR), Fortaleza - CE, Brazil
[6]University of Split, Split, Croatia
csousa.natanael@gmail.com; jose.sobral@ubi.pt; joeljr@ieee.org; ricardoalr@ufpi.edu.br; psolic@fesb.hr

***Abstract*—Since its definition, RPL (the IPv6 Routing Protocol for Low-Power and Lossy Networks) has been emerging as the standard protocol for routing in Internet of Things (IoT) solutions. RPL is a proactive routing protocol that performs the process of route creation based on Objective Functions (OFs). The OFs are responsible for defining rules and constraints to select the best paths considering different routing metrics. In its definition, RPL does not impose the use of a default OF and indicates that an OF should be selected according to the application. Thus, this paper proposes an Energy Efficient and Path Reliability Aware Objective Function (ERAOF) for IoT applications that requires energy efficiency and reliability in data transmission. The ERAOF is based on the composition of energy and link quality routing metrics. Results show that ERAOF is able to improve the network performance when compared to other OFs available in the literature.**

I. Introduction

Internet of Things (IoT) is regularly defined as a novel paradigm that enables the communication among things (assuming that any object can be connected to the Internet) in a ubiquitous and pervasive way through different technologies [1]. IoT application scenarios can cover various environments around people and be widely diverse, such as the following: urban sensor and actuators networks [2], industrial monitoring [3], and home automation [4]. In order to allow an IoT application fulfilling its objectives with efficiency, it is necessary that a routing protocol can provide data communication in an efficient way [5]. The routing protocols should take into account the different application requirements, such as low latency, high reliability, and efficient energy consumption.

Among the routing protocols used in IoT, RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) [6] has been emerging as the *de facto* standard solution [7]. RPL is a routing solution based on IPv6 proposed by IETF (Internet Engineering Task Force) and projected for Low power and Lossy Networks (LLN). The popularity of RPL grows constantly justified by its high flexibility for different applications, QoS (Quality of Service) support, security resources, among others [8]. The high flexibility of RPL enables it to be used in several applications and fulfill different network requirements. The component responsible for providing the adaptation of the protocol to the exigencies of a given application is the Objective Function (OF). The OF allows the route selection based on the routing metrics according to the interest of each application.

Although several OFs have been proposed for RPL, few of them are designed to attend the requirements of IoT applications. Thus, this work proposes the Energy Efficient and Path Reliability Aware Objective Function (ERAOF), a new objective function for IoT applications that require high reliability and efficient energy consumption. To reach its goal, the ERAOF merges the metrics of Energy Consumed (EC) and Expected Transmission Count (ETX) at the moment of a route selection. Taking into account the EC, ERAOF can avoid the use of paths with low remaining energy levels. At the same time, the influence of ETX allows the ERAOF to select paths with a high probability of success in the packet transmission. Thus, it is expected that ERAOF can fulfill the requirements of an IoT application that demands high reliability with an efficient energy consumption.

The rest of this paper is organized as follows. Section 2 describes the RPL protocol and some important related works available in the literature are presented in Section 3. Section 4 describes the proposed ERAOF while the performance evaluation study and results are analyzed in Section 5. Finally, Section 6 concludes the paper and suggests future works.

II. IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL)

RPL is a routing protocol for LLNs, created by the RoLL working group and defined by IETF as the standard routing protocol for 6LoWPAN networks (IPv6 over Low power Wireless Personal Area Networks). RPL describes a method of constructing a logical topology called Destination Oriented Directed Acyclic Graph (DODAG) [9] where the DODAG root is the gateway (or sink) node (Figure 1). The DODAG is built based on Objective Functions (OFs) that define the best paths in the network considering several routing metrics as number of hops, latency, delivery ratio, node energy, throughput, link quality, and transmission reliability [10]. The OF allows the nodes selecting its preferred parent in a set of neighbor

reachable with just one hop. At the moment of data message forwarding, the selected preferred parent is used as a path for reaching the DODAG root.

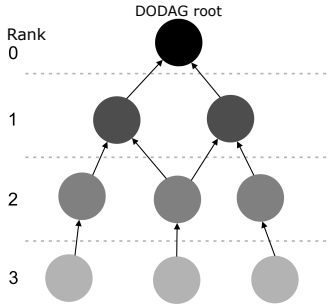


Fig. 1. Illustration of the DODAG graph.

The protocol supports three different traffic patterns: multipoint-to-point (MP2P), point-to-multipoint (P2MP) and point-to-point (P2P) [11]. In the first (MP2P), the nodes of the DODAG send the data to the root. In the second (P2MP), the DODAG root sends data/actions to the other nodes. In the last (P2P), the network nodes send messages to the other nodes (non-root) of the network.

In its definition, RPL specifies four types of control messages for network maintenance and information exchange. The first type is called DODAG Information Object (DIO) message and represents the primary source of routing control. The second is called Destination Advertisement Object (DAO) message, which is responsible for enabling the downward data traffic. The third type is DODAG Information Solicitation (DIS) message, which makes it possible for a given node to request a DIO message from any reachable neighbor node. The fourth type is the DAO-ACK message that is sent in response to a received DAO message [12].

The protocol tries to avoid loops through the computation of the position representative value of each node inside of DODAG, called Rank. The Rank value is computed based on the used OF and must comply some general properties as monotonicity.

The protocol is performed in four phases:

- **Configuration Phase**: the construction of the network topology begins at the configuration phase. On this phase, the root node broadcasts DIO messages. All the nodes that receive a DIO message, add the root in a routing table. This table stores the address of the neighbor nodes that can be used for possible upward or downward future communications.
- **Route Establishing Phase**: the nodes compute its Rank based on the information received in the DIO message. If the computed Rank is higher than the Rank inside of the DIO message, the node selects the DIO sender as its preferred parent. When a new device desires to join the network, it must send DIS messages requesting DIO of its neighbors. Thus, all the nodes in the neighborhood that receive a DIS must answer with a DIO for this new device in order to compute its Rank in the DODAG.
- **Data Communication Phase**: the data messages flow in the network with destination to the root according to the routes selected in the route establishing phase. Based on the traffic pattern, the data traffic can occur in an upward or downward fashion.
- **Path Repair Phase**: due to the inherent features of network topologies, the routes to the root change by several factors. Some reasons include changing of the preferred parent, external interferences in the communication links, and battery exhausting. The changes in the upward routes require updating of downward routes. Thus, a DAO message must be sent every time a route is updated or whether the preferred parent is changed.

III. Related Works

In the last years, several studies have been performed to the specification and deployment of objective functions in RPL using the recommended metrics for the LLNs (Low power and Lossy Networks). This section reviews the main relevant objective functions available in the related literature.

The default OF for RPL, which is called OF0 (Objective Function Zero), was designed to enable interoperability between different implementations of RPL [13]. OF0 presents a simple operation and does not use routing metrics in the rank definition. A node chooses as its preferred parent the reachable neighbor that has the lowest rank. Given a node n, its rank is defined by $R(n)$ as shown in Equation 1.

$$R(n) = R(P) + rank_{increase} \quad (1)$$

Where:
$R(n)$ is the new rank of the node (n);
$R(P)$ is the rank of the preferred parent;
$Rank_{increase}$ is a variation factor (delta) between the ranks of the parent and the node, expressed by the Equation 2.

$$rank_{increase} = (Rf \cdot Sp + Sr) \cdot MinHopRankIncrease \quad (2)$$

Where:
Rf is a configurable factor that is used to multiply the value of the link property. By default, it uses the value 1;
Sp is the step of the Rank;
Sr is the maximum value assigned to the Rank level to allow a viable successor;
$MinHopRankIncrease$ is a constant variable whose default value is 256.

In [14], the authors present the Minimal Rank with Hysteresis Objective Function (MRHOF). The MRHOF is based on the metric container concept that explains a set of metrics properties and/or constraints to be considered in the routing process. MRHOF is compatible only with additive metrics as specified in RFC 6551 [10]. Preferred parent selection is based on path cost considering the adopted metric where routes that minimize the cost associated with metric are preferred. By

default, the MRHOF uses ETX [15] for measuring the quality of links among the nodes. ETX estimates the required average number of transmissions, including retransmissions, so that a packet is correctly delivered to the destination. The ETX is defined according to Equation 3.

$$ETX = \frac{1}{Df \cdot Dr} \tag{3}$$

Where:

Df is the probability of the packet being received by the neighbor;

Dr is the probability that the acknowledgment is successfully received.

In [16], the authors present a performance evaluation study analyzing the combination of four metrics: hop count, ETX, RSSI (Received Signal Strength Indicator), and remaining energy. The metrics are combined in pairs using lexical or additive composition. In the additive composition, the node rank is calculated based on a weighted sum of the used metrics. In the lexical composition, a node selects the neighbor with the lowest (or highest) value based on the first metric, just if these values are equals, the node compares the second metric. The results shown that the performance of the studied combinations is dependent of the order of metrics priority.

In [17] is proposed an objective function based on fuzzy logic, named OF-FL. This function combines a set of metrics including point-to-point delay, ETX, hop count, and battery energy level, providing routing decisions to the network nodes during preferred parent selection. The metrics chosen by the authors were used as inputs in a fuzzy inference system resulting in a value indicative of the neighboring nodes quality. The obtained results showed that considering the studied scenarios, the OF-FL can improve point-to-point delay, packet loss rate, and network lifetime.

The Context-Aware Objective Function (CAOF) is proposed in [18]. Designed for wireless sensor networks, the CAOF is based on the remaining resources and in the change of the sensor state along the time. The proposed objective function (CAOF) performs a weighted sum of three metrics: node connectivity degree, battery energy level, and node position in the routing tree relative to the parent node. The final goal of the function proposed by this author is to find a delivery probability for each sensor node. The contributions of the above-mentioned studies are considered to propose the Energy Efficient and Path Reliability Aware Objective Function, described in the next section.

IV. Proposed Approach

Based on the current necessity of OFs able to improve the performance of RPL in IoT-based applications, this paper proposes an Energy Efficient and Path Reliability Aware Objective Function (ERAOF). ERAOF is a novel objective function for RPL based on node energy and link quality that aims to optimize the routing process to fulfilling applications that require energy efficiency and data transmission reliability.

As aforementioned, ERAOF is based on two routing metrics: energy consumed (EC) and ETX. Considering EC, ERAOF turns the RPL aware of the network power consumption. Thus, the protocol can choose the path with a low probability of link broken caused by energy exhaustion. Simultaneously, taking into account the ETX, ERAOF enables the RPL to know the quality of link among the network nodes. This feature can decrease the use of connections with less conditions and contribute to an enhanced network performance. With the use of ERAOF, each node computes a value $T(n_i)$, which represents the quality of a node i in terms of its own EC and ETX based in the link for the DIO message sender j as presented in Equation 4.

$$T(n_i) = F_{EC}(n_i) + F_{ETX}(n_i, n_j) \tag{4}$$

Where:

$F_{EC}(n_i)$ is the function that returns the energy consumed by the node i since the beginning of its operation;

$F_{ETX}(n_i, n_j)$ is the function that returns the ETX based on the link between a node i and the DIO message sender j.

Every time that receives a DIO, the node must calculate its $T(n_i)$. After computing it, the node i forwards a DIO to its neighbors with the sum of the calculated value plus $T(n_j)$ (previously received inside of the DIO message). This process allows the nodes to know the quality of its neighbors and, consequently, the quality of the route to the gateway (root) node.

The quality of a route r, in terms of EC and ETX, is defined by $Q(r)$, whose value is given by the sum of the $T(n_i)$ values of the nodes that compose it, according to Equation 5. During the network operation, RPL must select the best route for sending a data message based on the computed $Q(r)$. Thus, considering a set of available paths, the protocol must select the one with the lowest $Q(r)$ value, which represents a route with the best value of the composition of power consumption and ETX.

$$Q(r) = \sum_{i=1}^{j} T(n_i) \qquad i, j, r \in \mathbb{N}^* \tag{5}$$

Next section presents a performance assessment study of the proposed ERAOF in comparison with the most relevant OFs proposals available in the literature.

V. Performance Evaluation and Results Analysis

To evaluate the performance of the proposed OF, the experiments were realized using COOJA [19], an IoT simulation tool available at the Contiki operating system. The proposed ERAOF was evaluated in comparison to other two important objective functions available in the related literature, the OF0 (Objective Function Zero) [13] and the MRHOF (Minimum Rank with Hysteresis Objective Function) [14]. The performance of the three OF was evaluated considering the packet delivery ratio, number of hops, and spent energy for delivered data packet.

TABLE I
SIMULATION PARAMETERS.

Parameter	Value
Simulation Time	60 minutes
Initial Energy	20 J
Application	MP2P
Routing	RPL
Mac Protocol	802.15.4
Radio	CC2420
Data message rate	1 msg/min
Numbe of nodes	20, 40, 60
Network deployment	4x5, 5x8, 6x10 grids
Packet length	
Type of packet	Lenght
DIO	16 bytes
DAO	16 bytes
DAO-ACK	4 bytes
DIS	2 bytes
Data Packet	30 bytes

The evaluation study of the proposed OF was performed in three different network sizes. The physical topologies of the scenarios consider 20, 40, and 60 nodes deployed in grid. In the application considered for these networks, all the nodes sent data packets to just one gateway characterizing a multipoint-to-point (MP2P) traffic pattern. The simulations were repeated five times. The results are presented with a confidence interval of 95%. Other simulation parameters are shown in Table I.

Figure 2 exposes the results obtained for the packet delivery rate (PDR) metric. PDR represents the percentage of data packets sent from a sensor node that reaches its destination with success. According to the experiments, the use of ERAOF was able to increase the number of packets delivered when compared to the other approaches, mainly in the network with 60 nodes. Due to the use of the combination of ETX and EC metrics to select the preferred parent, the ERAOF allows the routing protocol to create routes with an efficient energy consumption and high reliability contributing to a reduction of the number of packet loss. Moreover, the proposed ERAOF was able to maintain a high PDR even with the network growing.

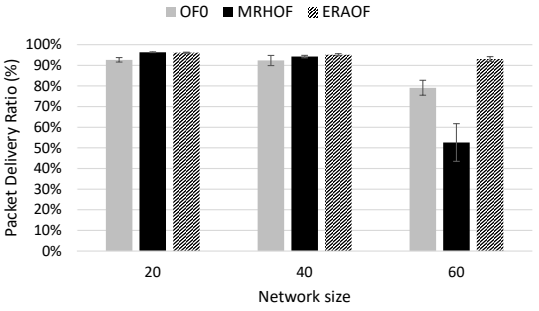


Fig. 2. Packet delivery ratio in function of the network size considering networks with 20, 40, and 60 nodes for OF0, MRHOF, and ERAOF using RPL protocol.

Figure 3 presents the results considering the number of hops. This metric represents the size of the path used by a node to achieve the message destination. Thus, the number of hops shows the number of times that a message is transmitted until reaching its destination. The experiment results reveal that, in small networks (with 20 and 40 nodes), the number of hops used by all the studied approaches is almost equal. However, in the network with the greater size (60 nodes), the MRHOF had a significant increase, in terms of number of hops, exposing its low scalability. On the contrary, ERAOF was able to demonstrate a consistent performance very similar to OF0, which is an OF that seeks the shortest path ever. It is also important to note that routes created with a high number of hops increase the energy consumption due the necessity of more messages forwarding and radio usage.

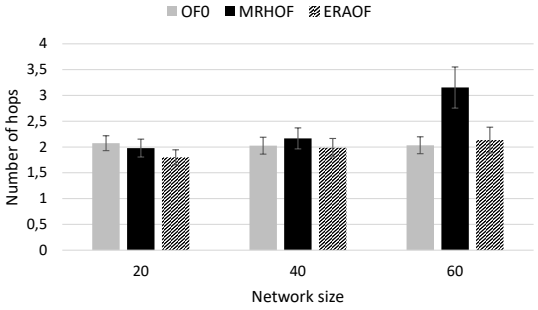


Fig. 3. Number of hops in function of the network size considering networks with 20, 40, and 60 nodes for OF0, MRHOF, and ERAOF using RPL protocol.

Figure 4 shows the results for the spent energy per delivered data packet. This metric reveals the average amount of energy that a node spent for a data message delivery to the gateway (or sink) node. The metric is calculated thought the ratio between the quantity of energy consumed and the number of data packets delivered with success. The obtained results reveal the studied OFs were able to maintain very close results, with the exception of MRHOF OF in the network with 60 nodes. Although ERAOF had obtained better results in the packet delivery ratio when compared to OF0, its performance in this metric was close of OF0. These results show that ERAOF has spent more energy for reaching a high delivery ratio.

VI. CONCLUSION AND FUTURE WORK

This work proposed a new objective function for RPL protocol. The proposal, named ERAOF, aimed to provide energy efficiency and reliable data communication for IoT applications. To reach this main goal, the ERAOF applies the composition of the routing metrics of energy consumed and ETX for select the best path to forward a data message. The performance assessment study shown the proposed OF can increase the packet delivery ratio keeping an effective energy consumption and the use of a low number of hops. Thus, the main contribution of this work is a new objective function for RPL that can offer high packet delivery ratio for IoT

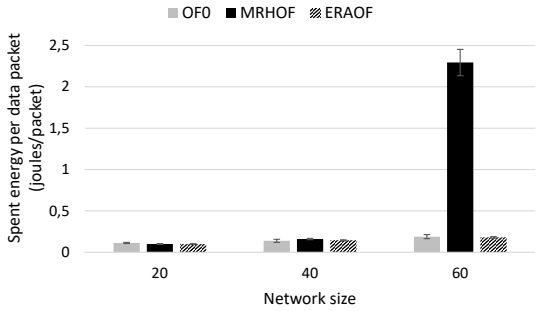


Fig. 4. Spent energy per delivered data packet in function of the network size considering networks with 20, 40, and 60 nodes for OF0, MRHOF, and ERAOF using RPL protocol.

applications with an efficient power consumption of network resources.

For future work, the authors propose a complete performance evaluation study of ERAOF considering different scenarios, metrics, and traffic patterns.

Acknowledgments

This work was supported by National Funding from the FCT - Fundação para a Ciência e a Tecnologia through the UID/EEA/500008/2013 Project, by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) – Brazil through the grant 201155/2015-0, and by Finep, with resources from Funttel, grant no. 01.14.0231.00, under the Radiocommunication Reference Center (Centro de Referência em Radiocomunicações CRR) project of the National Institute of Telecommunications (Instituto Nacional de Telecomunicações Inatel), Brazil.

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things journal*, vol. 1, no. 1, pp. 22–32, 2014.

[2] M. Dohler, T. Watteyne, T. Winter, and D. Barthel, "Routing Requirements for Urban Low-Power and Lossy Networks," RFC 5548, May 2009. [Online]. Available: https://rfc-editor.org/rfc/rfc5548.txt

[3] K. Pister, P. Thubert, S. Dwars, and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks," RFC 5673, Oct. 2009. [Online]. Available: https://rfc-editor.org/rfc/rfc5673.txt

[4] A. Brandt, J. Buron, and G. Porcu, "Home Automation Routing Requirements in Low-Power and Lossy Networks," RFC 5826, Apr. 2010. [Online]. Available: https://rfc-editor.org/rfc/rfc5826.txt

[5] J. V. Sobral, J. J. Rodrigues, K. Saleem, and J. Al-Muhtadi, "Performance evaluation of loadng routing protocol in iot p2p and mp2p applications," in *Computer and Energy Science (SpliTech), International Multidisciplinary Conference on*. IEEE, 2016, pp. 1–6.

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

[7] O. Iova, G. P. Picco, T. Istomin, and C. Kiraly, "Rpl, the routing standard for the internet of things... or is it?" *IEEE Communications Magazine*, vol. 17, 2016.

[8] O. Gaddour and A. Koubâa, "RPL in a nutshell: A survey," *Computer Networks*, vol. 56, no. 14, pp. 3163–3178, 2012.

[9] H. Fotouhi, D. Moreira, and M. Alves, "mrpl: Boosting mobility in the internet of things," *Ad Hoc Networks*, vol. 26, pp. 17–35, 2015.

[10] D. Barthel, J. Vasseur, K. Pister, M. Kim, and N. Dejean, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks," RFC 6551, Mar. 2012. [Online]. Available: https://rfc-editor.org/rfc/rfc6551.txt

[11] B. Mohamed and F. Mohamed, "Qos routing rpl for low power and lossy networks," *International Journal of Distributed Sensor Networks*, vol. 2015, 2015.

[12] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[13] P. Thubert, "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)," RFC 6552, Mar. 2012. [Online]. Available: https://rfc-editor.org/rfc/rfc6552.txt

[14] P. Levis and O. Gnawali, "The Minimum Rank with Hysteresis Objective Function," RFC 6719, Sep. 2012. [Online]. Available: https://rfc-editor.org/rfc/rfc6719.txt

[15] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '03. New York, NY, USA: ACM, 2003, pp. 134–146. [Online]. Available: http://doi.acm.org/10.1145/938985.939000

[16] P. Karkazis, H. C. Leligou, L. Sarakis, T. Zahariadis, P. Trakadas, T. H. Velivassaki, and C. Capsalis, "Design of primary and composite routing metrics for rpl-compliant wireless sensor networks," in *Telecommunications and Multimedia (TEMU), 2012 International Conference on*. IEEE, 2012, pp. 13–18.

[17] O. Gaddour, A. Koubâa, N. Baccour, and M. Abid, "OF-FL: QoS-aware fuzzy logic objective function for the rpl routing protocol," in *2014 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2014, pp. 365–372.

[18] B. Sharkawy, A. Khattab, and K. M. F. Elsayed, "Fault-tolerant rpl through context awareness," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, March 2014, pp. 437–441.

[19] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*, Nov 2006, pp. 641–648.

Appendix C

A Proposal for IoT Dynamic Routes Selection Based on Contextual Information

This appendix consists in the following paper:

A Proposal for IoT Dynamic Routes Selection Based on Contextual Information

Harilton Araújo, Raimir H. Filho, Joel J. P. C. Rodrigues, Ricardo A. L. Rabêlo, Natanael Sousa, José Carlos L. Filho, and José V. V. Sobral

Sensors, MDPI, ISSN: 1424-8220, Vol. 18, No. 2, Article 353, January 2018, pp. 1-16.

DOI: doi.org/10.3390/s18020353

According to 2019 Journal Citation Reports published by Thomson Reuters in 2020, this journal scored ISI journal performance metrics as follows:

ISI Impact Factor (2019): 3.275

ISI Article Influence Score (2019): 0.530

Journal Ranking (2019): Q1 - 15/64 (Instruments & Instrumentation)

Journal Ranking (2019): Q2 - 77/266 (Engineering, Electrical & Electronic)

sensors

MDPI

Communication

A Proposal for IoT Dynamic Routes Selection Based on Contextual Information

Harilton da Silva Araújo [1,2,3], Raimir Holanda Filho [1], Joel J. P. C. Rodrigues [1,2,4,5,*], Ricardo de A. L. Rabelo [5], Natanael de C. Sousa [6], José C. C. L. S. Filho [6] and José V. V. Sobral [2,7,8]

1 Programa de Pós-Graduação em Informática Aplicada (PPGIA), University of Fortaleza (UNIFOR), Av. Washington Soares, 1321, Edson Queiroz, 60.811-905 Fortaleza-CE, Brazil; harilton@edu.unifor.br (H.d.S.A.); raimir@unifor.br (R.H.F.)
2 Instituto de Telecomunicações, Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal; jose.sobral@it.ubi.pt
3 Department of Computer Science, University Estacio of Sá, Av. Expedicionários, 790, São João, 64.046-700 Teresina-PI, Brazil
4 National Institute of Telecommunications (Inatel), Av. João de Camargo, 510-Centro, 37540-000 Santa Rita do Sapucaí-MG, Brazil
5 International Laboratory "Technosphere Safety", ITMO University, 49 Kronverksky Pr., St. Petersburg 197101, Russia; ricardoalr@ufpi.edu.br
6 Department of Computing, Federal University of Piauí (UFPI), Department of Computing, 64.049-550 Teresina-PI, Brazil; csousa.natanael@gmail.com (N.d.C.S.); jcarloslimafilho@gmail.com (J.C.C.L.S.F.)
7 Departamento de Informática, Universidade da Beira Interior, Rua Marquês D'Ávila e Bolama, 6201-001 Covilhã, Portugal
8 Department of Education, Federal Institute of Maranhão (IFMA), Av. Getúlio Vargas, 4, Monte Castelo, 65030-005 São Luiz-MA, Brazil
* Correspondence: joeljr@ieee.org; Tel.: +55-35-3471-9200

Received: 5 December 2017; Accepted: 23 January 2018; Published: 26 January 2018

Abstract: The Internet of Things (IoT) is based on interconnection of intelligent and addressable devices, allowing their autonomy and proactive behavior with Internet connectivity. Data dissemination in IoT usually depends on the application and requires context-aware routing protocols that must include auto-configuration features (which adapt the behavior of the network at runtime, based on context information). This paper proposes an approach for IoT route selection using fuzzy logic in order to attain the requirements of specific applications. In this case, fuzzy logic is used to translate in math terms the imprecise information expressed by a set of linguistic rules. For this purpose, four Objective Functions (OFs) are proposed for the Routing Protocol for Low Power and Loss Networks (RPL); such OFs are dynamically selected based on context information. The aforementioned OFs are generated from the fusion of the following metrics: Expected Transmission Count (ETX), Number of Hops (NH) and Energy Consumed (EC). The experiments performed through simulation, associated with the statistical data analysis, conclude that this proposal provides high reliability by successfully delivering nearly 100% of data packets, low delay for data delivery and increase in QoS. In addition, an 30% improvement is attained in the network life time when using one of proposed objective function, keeping the devices alive for longer duration.

Keywords: Internet of Things; routing; 6LowPAN; context-aware; objective function; fuzzy system

Appendix C: A Proposal for IoT Dynamic Routes Selection Based on Contextual Information

1. Introduction

Internet of Things (IoT) is referred to as the interconnection of physical objects that have an IP address for Internet connectivity. A huge number of smart devices are interconnected to the Internet today. There is an exponential growth in the number of smart devices interconnected through the mobile Internet [1]. IoT can be used in architectures, such as the proposal presented in [2], which represents an architecture for two-way communication between smart utility meters and utility companies. Is a wireless sensor network that provides communication for metering devices in the neighborhood area of a smart grid. Another point that is addressed in IoT is mobility, which is a fundamental issue related to the detection of mobile nodes movement and the provision of better links through an efficient routes selection [3]. Many contributions are evidenced in the literature in the IoT topic, such as the following: study of mechanisms for the topology control [4], location and mobility in Wireless Sensor Networks (WSNs) [5] and routing based on objective functions [6–8]. It is considered also as pervasive presence of devices variety, such as sensors, Radio Frequency Identification (RFID) tags and smartphones, among other devices which can interact with each other for a common purpose [9]. In a scenario of IoT, plurality is increasing and forecasts indicate that over 40 billion devices will be connected until 2020 [2], allowing the emergence of an infinity of new applications, such as for smart cities, healthcare, smart houses, industry, agriculture, etc.

Among the most promising technologies for the IoT paradigm, RFID and Wireless Sensor Networks (WSNs) are the most common and appropriate [9,10]. WSNs have limitations on the identification of a person or object in some types of applications. However, unlike WSNs, RFID systems are unable to sense data from the place in which they are used, such as humidity, temperature and pressure that are provided by sensors. This is an indication that IoT, by means of the integration between the RFID and WSN technologies, maximizes the benefits, thus opening up new perspectives for applications that consider context information, such as temperature monitoring in remote areas and air quality of a specific city or region, vehicle control flows, among others.

The context usually refers to the collected information regarding a given situation and may include also the location. Moreover, it can also cover different information used to characterize entities involved in the interaction between the user and the application. According to [11], context-sensitive systems are able to adapt their behavior to current conditions with no explicit intervention from users.

In IoT, an amount of information, such as features of the device itself, of the environment which surrounds it and its capacity, can be used as a source of context information. Many papers available in literature are converging in efforts to address issues involving context sensitivity in IoT, especially at the route selection stage [12–14]. At this stage, devices (which have resource constraints) process contextual information locally in order to select the path that best meets the requirements of a given application. Such feature requires context-sensitive routing protocols for the fulfillment of a number of challenges during messages exchange, including shorter delay, greater reliability on data transmission and minimal power consumption. Based on these challenges, this paper proposes an improvement for the RPL protocol [15] that considers the creation of four new Objective Functions (OFs) that enable the devices to select the parent node (default route) based on the context information obtained from the application. The use of proposed OFs occurs in the process of establishing routes in order to meet the context of the applications. This is an optimization problem, which seeks to maximize the reliability of data packet delivery, minimize the delay and increase QoS and network lifetime by using one or more of the objective functions proposed in this work. The objective functions meet a set of constraints (traditionally, constraints are expressed by algebraic equations). In this way, the objective function is used as a means of quantifying and qualifying the potential solutions of the problem in question. Then, the main contributions of this work are the following: (i) a proposal of creation of four new Objective Functions (OFs) that are used dynamically and at run time and (ii) a Fuzzy System-Based Route Classifier that enable the devices to establish routes high quality order to meet the context of the applications specific.

The rest of the paper is organized as follows. Section 2 sets out the related work on the topic and Section 3 describes the proposed improvement on the RPL protocol. The performance assessment of the proposed approach is discussed in Section 4, followed by the conclusions and research directions in Section 5.

2. Related Work

According to [16], RPL makes use of metrics specified in the RFC 6551 which are suitable for IPv6 over Low Power Wireless Personal Area Networks (6LoWPAN) environments, considering the number of hops; latency; delivery ratio, node energy; throughput; level of link quality; and transmission reliability.

In [17], a standard objective function, called Objective Function Zero (F0), was proposed for RPL. It was designed to enable interoperability among different implementations of RPL. The operation of OF0 is simplified and does not use any routing metrics for the definition of the rank (position in the routing tree). A device chooses its preferred node considering the neighbor within those that presents the lowest rating.

The Minimum Rank with Hysteresis Objective Function (MRHOF) it is presented in [6]. This OF is only suitable for the metrics specified in RFC 6719. The selection of the favorite "parents" is made taking into account the adopted metrics, where the routes that minimize the cost associated with the metrics are preferable. As standard, MRHOF uses the Expected Transmission Count (ETX) metric, from [14], which assesses the quality of the link and aims to maximize the leakage of data. The ETX is defined according to Equation (1).

$$\mathrm{ETX} = \frac{1}{D_f \cdot D_{rss}} \tag{1}$$

where:

- D_{fx} is the probability of the package being received by the neighbor; and
- D_{rx} is the probability of the ACK being received successfully.

The authors of [18] proposed a context-aware approach that changes the functioning of a sensor node based on the phenomena collected from the environment, such as temperature, humidity, pressure, etc. This contribution called Situation-Aware Adaptation Approach for Energy Conservation in Wireless Sensor Network (SA-A-WSN), aims to monitoring the way the sensor node works in the environment in order to reduce the network's power consumption.

According to [13], context-sensitive computing has been gaining more market and relevance since it is currently being considered an important component of IoT, mainly because context-sensitive systems are able to adapt their behavior to current conditions with no explicit intervention from users.

A context-sensitive objective function, called Context Aware Objective Function (CAOF), proposed by [14], is based on residual resources and on changing the sensor node's status over time. The function proposed by [15] performs a weighted sum of the following metrics: degree of the node connectivity, level of battery power and position of the node in the routing tree regarding the parent node. The ultimate goal of the function proposed in [15] is to find a probability of delivery to each sensor node.

The use of fuzzy logic to calculate the objective function for RPL protocol is proposed in [19]. The authors describe that the objective function is the component used to select the paths by identifying a parent node among many existing nodes. The objective function, called QoS-aware fuzzy logic objective function (OF-FL), associates parameters with linguistic variables which are combined with fuzzy rules in order to identify the route to be selected. The parameters considered in OF-FL are hop numbers, end-to-end delay, packet loss rates and default route change rate.

In [20], a Scalable Context-Aware Objective Function (SCAOF) is proposed. It adapts the RPL protocol to the environmental monitoring within the area of agriculture with a scalable context. The performance of the SCAOF was assessed both through simulation and field testing.

The experimental results obtained confirm that SCAOF can extend the network's service life and improve the Quality of Service (QoS) in the different agriculture-oriented simulation scenarios.

The work included in [6–8] represents recent research work considering objective functions in the route selection process. In [6], the performance of two RPL objective functions (the Minimum Classification with Hysteresis Objective Function (MRHOF) and the Function Zero Objective (OFO)) are analyzed. It is observed that MRHOF offers better performance than OF0 in terms of network quality. The MRHOF is suitable for use in sensor network that require data delivery in a reliable network. In turn, OF0 is suitable for use in sensors network that require fast network connection training and low power consumption.

A Smart Energy Efficient Objective Function (SEEOF) is proposed in [7]. It was designed to create an IPv6 mesh topology for IoT based on smart metering applications. The SEEOF is based on RPL and it is designed to use energy efficiently and extend the network lifetime. It mainly takes into account the node energy. The energy consumption in RPL using SEEOF was compared with MRHOF [6]. Simulation results show that an improvement up to 27% is attained in the network life time when using the proposed objective function. More important, the results show that SEEOF balances the energy consumption more uniformly among the battery powered devices keeping them alive for longer duration.

In [8], an Energy Efficient and Path Reliability Aware Objective Function (ERAOF) for IoT applications that requires energy efficiency and reliability in data transmission is proposed. The ERAOF is based on the composition of the following metrics: node energy and link quality. Different from the proposal presented in [7], ERAOF considers not only the energy consumed but also the quality of the link in the process of route selection. This characteristic motivates its inclusion in the comparative analysis performed in this work.

The proposal of this work was based mainly on the works presented in [7,8,14,19,20]. These works motivated the authors to propose a new approach for five main reasons: (i) to use appropriate metrics for 6LoWPAN environments; (ii) to perform a weighted average of the metrics to provide quality message transmission through a sensor network without wire; (iii) to use the fuzzy logic objective function; (iv) to use an energy efficient objective function; and (v) to use Context-Aware Objective Function. The proposed approach is presented in the next section.

3. Proposed Approach

In this paper, we propose an adaptation of the Routing Protocol for Low-power and Lossy-networks (RPL)—(RFC 6550) protocol. The main purpose of this proposal is to optimize routing in order to meet the context requirements of specific applications.

3.1. The RPL Protocol

RPL is a network layer protocol and specifically designed to be used along with 6LoWPAN. It operates under several mechanisms of the connection layer, including the IEEE 802.15.4 MAC and PHY layers [17]. Please refer to Table 1 for a description of the symbols used in the remainder of this section.

Table 1. Symbols definition.

Symbol	Meaning
n	nodes connected to network
Ns	set of network nodes
L	set links between nodes
ni	node enabled to have a parent
$A(ni)$	set of progenitor nodes
ns	root node
l	number of connections of ni
$L(ni)$	set of connections to candidate nodes
aj	node selected as parent
ap	node selected as child

An IoT scenario with RPL can be modeled as a graph $G = (N, L)$, where N is the set of n network nodes (excluding the root node) and L is the set of connections that link such nodes. The graph G represents the creation of the network topology including the created routes and sent messages. For each $ni \in N$ ($i = 1, 2, ..., n$) there is a set A that contains nodes enabled to be candidates to become the parent node of another node. This process is performed based on the rank of each node. The rank is a number that identifies the position of a node in the topology relative to the other nodes and to the root of the graph. The use of ranks makes it possible for a node to differentiate its position from the parent nodes and sibling nodes. It consolidates a mechanism where a node decides its parent node according to the smaller rank, which is defined as an integer representing the individual position of the node within the graph G. This parameter increases its value as it descends in the network topology. The nodes involved in the creation of this topology are represented by Expressions (2) and (3):

$$N = \{n1, n2, \ldots, nn\} \tag{2}$$

$$A(ni) = \{a1, a2, \ldots, am\} \tag{3}$$

where:

- N is the set of nodes of a network;
- $A(ni)$ is the set of progenitor nodes of a given node.

Considering that A is a subset of the total number of nodes, then, $A(ni) \subset N$. The root node represented as ns, is the only node from the network which has no list of candidates because sent messages only downward direction; that means its subset is empty, Equation (4):

$$A(ns) = \emptyset \tag{4}$$

For each ni node there is a set of m connections l equal to the number of candidate nodes, as demonstrated in Equation (5). This means that each node has a number of connections (edges) corresponding to the number of nodes that are candidates to be progenitors. For example, if a node has a connection to six other nodes, there are six candidates to be progenitors. In a set L there is an l connection that belongs to $L(ni)$. Therefore, the conclusion is that aj belongs to $A(ni)$, where aj is the candidate node selected to be the parent node of ni.

$$L(ni) = \{l1, l2, \ldots, ln\} \tag{5}$$

In Equation (6), each connection (represented by "l") has a rank, which equals the rank calculated in the ni node in relation to a selected parent node aj ($j = 1, 2, \ldots, m$). The rank is obtained based on the position of a node in the topology regarding the other network nodes. In this case, a node decides who is its parent node according to the lowest rank.

$$l(ni, aj) = rank \tag{6}$$

Equation (7) must be followed because a candidate node can be selected as a parent node. In this case, when the rank of a node ap (receiver node) is greater than the rank of aj (sender), the ap accepts aj as its parent node, creating through it a path to ns (root node). Otherwise, it rejects the transmitter as its parent node.

$$l(ni, ap) > l(ni, aj)\ \forall aj \in l \tag{7}$$

In conclusion, the construction of the network for each ni node can be defined as follows:

$$C = \{\forall ni \in N, \ni ap \in A(ni): l(ni, ap) > l(ni, aj)\ \forall aj \in l\} \tag{8}$$

RPL protocol is implemented in 4 stages: (a) setup stage; (b) routes establishment stage; (c) data communication stage; and (d) path reconstruction stage.

At the route Setup stage, the RPL protocol (in its original version) uses the OF0 to enable the nodes to select the parent nodes (default route), based on information from the rank and on the number of hops of a given node for the sink, as described in Equation (9):

$$R(N) = R(P) + rank\_{increase} \quad (9)$$

where:

- $R(N)$, is the node new rank;
- $R(P)$, is the preferred parent rank;
- $rank\_{increase}$, is a variation factor (delta) between the node and the parent rank.

3.2. Proposed Approach

In order to improve RPL protocol, four objective functions are proposed to be used at the routes establishment stage: DQCA-OF1, DQCA-OF2, DQCA-OF3 and DQCA-OF4. Each one of these objective functions is based on the fusion of metrics for 6LoWPAN environments, used by the RPL and described in RFC 6551. The context information is used as a parameter for selecting the one that best suits a given application, among the four available objective functions.

The APPLICATIONS module represents the context of each application. The DATABASE module contains the objective functions which will be used for generating routes that meet the requirements of each application.

Figure 1 illustrates the functioning of the proposed approach, which is based on the relationship among three nodes.

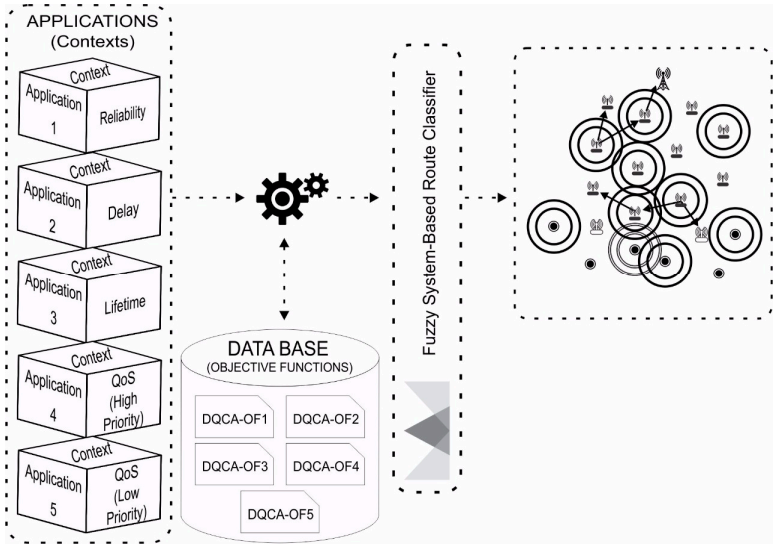


Figure 1. Functional model of the proposed approach.

3.2.1. Objective Functions

Existing objective functions have constraints, since an objective function composed by a single routing metric presents advantages and limitations. A single metric present in the objective function may not fully satisfy the requirements required by the applications. For example, while the number

of jumps allows choosing the shortest path, this can lead to the failure of one or more nodes due to power consumption because the battery level is not considered in the decision process. In addition, considering ETX as a single routing metric can lead to high latency in routing messages. When selecting routes with low ETX, the network becomes more reliable but does not reduce the latency during the message routing process. Thus, the ETX metric alone is not suitable for real-time applications because it does not take into account the delivery time requirements of the messages.

The fixed and permanent combination of two metrics is insufficient to efficiently meet all the requirements of the applications, since the requirement of each application may change eventually. In addition, considering two routing metrics it is possible to optimize routing performance but at the cost of penalizing other performance parameters. For example, using the ETX metric may help the routing process to use more reliable paths, however, it may lead to excessive use of some network nodes reducing the remaining energy. Thus, it is necessary to combine several metrics to be able to characterize the route quality more efficiently. For the best route selection, it is proposed the combination of several important routing metrics.

The objective functions proposed in this approach enable the devices to select the parent nodes (default route) based on the context information obtained from the application. Each application may have requirements that simultaneously needs up to three metrics (Expected Transmission Count (ETX), Number of Hops (*NH*), and/or Energy Consumed (*EC*)). These requirements allow rating them according to their priority level (N) considering the following levels: High = 1; Medium = 3; and Low = 5. Each metric is represented by an *F* function. Based on this information, a $T(ni)$ weight is achieved by each network device, which represents the sum of the context functions.

- Delivery Quality and Context Aware Type 1 Objective Function (DQCA-OF1):

$$T(ni) = NETX \cdot FETX(ni) + NNH \cdot FNH(ni) \tag{10}$$

The DQCA-OF1 function considers the metrics Expected Transmission Count (ETX) and Number of Hops (*NH*) for the calculation of the route selection.

- Delivery Quality and Context Aware Type 2 Objective Function (DQCA-OF2):

$$T(ni) = NETX \cdot FETX(ni) + NEC \cdot FEC(ni) \tag{11}$$

The DQCA-OF2 function includes the metrics Expected Transmission Count (ETX) and Energy Consumed (*EC*) for the route selection calculation.

- Delivery Quality and Context Aware Type 3 Objective Function (DQCA-OF3):

$$T(ni) = NNH \cdot FNH(ni) + NEC \cdot FEC(ni) \tag{12}$$

The DQCA-OF3 function uses the metrics Number of Hops (*NH*) and Energy Consumed (*EC*) for the route selection calculation.

- Delivery Quality and Context Aware Type 4 Objective Function (DQCA-OF4):

$$T(ni) = NETX \cdot FETX(ni) + NNS \cdot FNS(ni) + NEC \cdot FEC(ni) \tag{13}$$

The DQCA-OF4 function includes the metrics Expected Transmission Count (ETX), Number of Hops (*NH*) and Energy Consumed (*EC*) for the route selection calculation.

3.2.2. Fuzzy System-Based Route Classifier

According to [21], several techniques of Computational Intelligence (CI) have been used to solve routing problems in WSNs. Intelligence techniques reinforce the efficiency of routing protocols in decision making to optimize the network performance. In order to decide efficiently on the choice of the best route among those available, fuzzy logic was used to provide a rigorous algebra to deal

with inaccurate information, since it is a convenient method of combining conflicting objectives and specialized human knowledge and allowing implementation in low complexity algorithms.

To ensure greater consistency in decision making during the routing process, this paper proposes a route classifier based on fuzzy systems, as shown in Figure 2, capable of estimating the degree of quality of the routes, in order to classify them, with the purpose of assisting the selection of routes in IoT scenarios. The route classifier uses as input the values assigned to the metrics (present in the objective functions) specified in Expressions (14)–(16) to rank the routes in order of quality. The trapezoidal form was chosen for the pertinence functions, since it is widely used in fuzzy logic systems [22].

$$NETX \cdot FETX\ (ni) \tag{14}$$

$$NNS \cdot FNS\ (ni) \tag{15}$$

$$NEC \cdot FEC\ (ni) \tag{16}$$

where:

- *NETX* is the ETX priority level;
- *FETX* is the value assigned to the ETX function;
- *NNS* is the priority level of *NS*;
- *FNS* is the value assigned to the *NS* function;
- *NEC* is the priority level of *EC*;
- *FEC* is the value assigned to the *EC* function;
- (*ni*) corresponds to the network node.

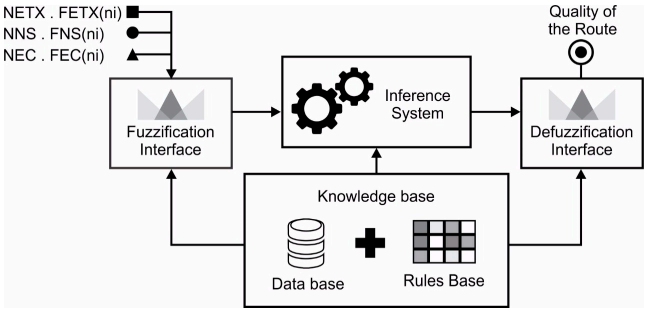


Figure 2. Structure of the Fuzzy System-Based Route Classifier.

The fuzzification interface has the function of mapping the precise inputs to the input fuzzy sets by determining the membership degree, i.e., it transforms quantitative information into qualitative information. Thus, the fuzzification interface can be seen as a function that guarantees a degree of imprecision to a numerical value, mapping the physical value of a process variable into a normalized universe of discourse [22]. For the determination membership degree of each entry, the pertinence functions are used. Several profiles of pertinence functions are found in fuzzy system implementations. The most common and easy to generate pertinence functions are triangular and trapezoidal, although there are other functions such as Gaussian, sigmoid and cubic spline [23].

The knowledge base considers two components, database and rule base. The database contains the primary terms (linguistic terms, natural language terms) for each variable and the membership function of each primary term. The rule base relates (or maps) the system inputs to their outputs, thereby implementing the policies for the estimation, control and decision making of language rules. The rule base performs the mapping of the input domain to the output domain, thus, plays an important role in generating the results produced by the fuzzy inference system.

The inference system is responsible for evaluating the input variables, applying the production rules of the knowledge base and assigning responses to the processing. This response is derived from three steps: evaluation of antecedent, implication and aggregation of the consequent [24]. In the stage of antecedents' evaluation, the activation of rules occurs, it is in this stage that the degree of pertinence of each proposition present in the antecedent of each rule is calculated. The implication consists in calculating the consequence of the rules that have degree of activation greater than zero. Most of the time, multiple rules are fired at the same time resulting in multiple output sets. Therefore, it is necessary to generate a unique response for each output variable, containing information about all fired rules. The last phase is responsible for aggregating all output sets into a single set. This junction is made according to the fuzzy predicates in the normal form (conjunctive or disjunctive).

Through the use of the route classification system proposed in this work, the messages carry information about expected transmission count, number of hops and energy consumed. This information is used in the fuzzy inference process of the route classifier to determine the quality of the route.

Table 2 shows some fuzzy rules that represent the combination of different metrics, which characterize the input fuzzy sets, i.e., EC, ETX and NS. As the fuzzy system used has three inputs with three pertinence functions for each input, our rule base includes 27 rules, converted to the associated linguistic variables such as Low, Medium and High.

Table 2. Example of fuzzy rule base used.

Energy Consumed	Number of Hops	ETX	Quality of the Route
Low	Low	Low	High
High	Medium	Low	Medium
High	High	Low	Medium
Medium	Low	High	Medium
High	Medium	Medium	Low
Low	High	Medium	Medium

For example, to elect a route with high quality, the energy consumed is low, the number of hops is low and the ETX is low. During the route selection process, fuzzy sets and rules are used according to the requirements of the application.

After the fuzzy inference process, this proposed approach obtains a fuzzy set as an answer. However, because of imprecision, frequently this set is not convenient as a final response of the system, requiring a synthetic numerical representation of the fuzzy response. Thus, function of defuzzification interface is to obtain a precise (non-fuzzy) result in the output of inference system. According to [25], there are several methods of defuzzification highlighting the following: Center of area/gravity; Center of largest area; Average of the maximum; First of the highs; Mean of the highs; and Height of maximum. It was considered in this work a simple and frequently used method, which is the centroid defuzzification method [26] that locates the equilibrium point of the diffuse region of the solution, calculating the weighted average of the diffuse region.

The route selection is based on the quality obtained in the last column of Table 1, which contains a fuzzy output variable, which corresponds to a route quality. From this indicator, the nodes decide locally which route to use to send data without incurring high network costs.

The rule implemented in the protocol suggests that, after a route is formed, a node must always send data messages using the bigger quality route as observing information contained in your cache.

4. Performance Assessment and Results

To evaluate the proposal presented in this work, the COOJA simulator [27] was used to evaluate the performance of networks with devices that run IoT-specific operating systems. In addition, the simulator has an interface to analyze and interact with the nodes, which facilitates the work and

network visualization. The COOJA, which is part of the Contiki simulation environment, do not have objective functions that cover all the metrics specified in RFC 6551, being restricted to the metrics of the objective functions adopted by RPL in its original version. For this purpose, the objective functions proposed herein were assessed and compared with the OF0, MRHOF, ERAOF and OF-FL functions. During the simulation experiments, the proposal presented at Ref. [7] was also compared with all simulated proposals in this work. However, the simulation results showed that the proposal of Ref [7] performed bottom when compared to the objective functions evaluated in this work. For this reason, it was not included in the comparative analysis presented in this article. The energy consumption model used for the performance assessment concerning energy, it is the Energest module from the Contiki [28], which measures the energy consumption within an IoT device.

4.1. Network Scenario and Used Metrics

The network scenario for simulation was built in order to validate the approach proposed in this paper. It simultaneously includes features from applications aimed at environmental monitoring, tracking and localization. The network topology is shown in Figure 3 and considers 11 sensor nodes, 6 tags (RFID), 2 reader nodes (RFID) and 1 sink node. There are 20 devices within a fixed network topology.

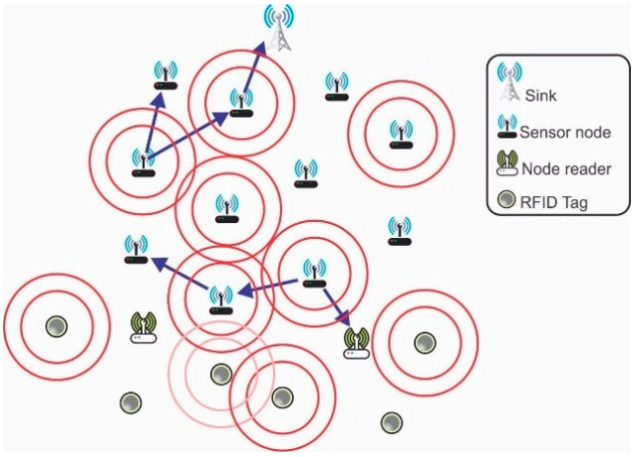


Figure 3. Topology of the network simulation scenario.

There is data flow for upward and downward. The sink node transmits DODAG Information Object (DIO) messages whenever there is a change of application and it is located as shown in Figure 3.

The use of RFID tags within the simulated scenario is justified because they can play an important role in IoT solutions and it is needed a realistic IoT scenario. According to [29], excess messages exchanged between readers and RFID tags have strong influence on the network's performance, as well as in application QoS requirements. In the studied scenario, the packets received by the readers are forwarded to the sink, which processes the information sent by readers and by sensor nodes in a consolidated manner.

The total simulation time was 35 min. To ensure that simulations converge to a steady state, each simulation experiment was repeated five times because, after the fifth experiment, no change was observed in the results. However, to evaluate the dynamic change between objective functions, the simulated time was about 140 min, since each application remains for 35 min. The messages were simulated with packets of 30 bytes (Contiki standard). The choice of the parent node for route establishment was based on the Fuzzy System-Based Route Classifier proposed in this work. The initial power of the nodes was adjusted to 200 joules above the Contiki Energest energy consumption

model [28]. In this work, the variation in the number of sensor nodes was not considered since, during the study, it was observed that there was no variation of the results changing the number of devices during the use of the proposed objective functions associated to route classifier based on fuzzy system. This is justified because the route selection process of the proposed approach uses only aspects related to energy, position in the routing tree and number of transmission estimates.

The following four metrics used by the RPL (RFC 6551), suitable for 6LoWPAN environments, were chosen for evaluation of this approach: Energy Consumed (*EC*), Number of Hops (*NH*), Delivery Ratio (TX) and Expected Transmission Count (ETX). The Expected Transmission Count (ETX) is equal to the average number of required transmissions (including retransmissions) so that a package is duly delivered to its destination.

Four applications were considered for this study. The first application requires priority in terms of reliability (represented by the delivery ratio). The second application requires a shorter delay (represented by the number of hops). The third application requires a long lifetime (represented by the energy consumed) and the fourth application requires QoS for data delivery (represented by the ETX).

4.2. Results Analysis

The results obtained show that the application that require high priority regarding lifetime and delay used, among the functions available in the DATABASE, the DQCA-OF4 function for the route selection process. This function was selected because it can forecast, in its structure, the metrics energy consumed and number of hops, allowing for the routes selection with lower energy consumed and a lower number of hops. Figure 4 shows the performance of the evaluated objective functions.

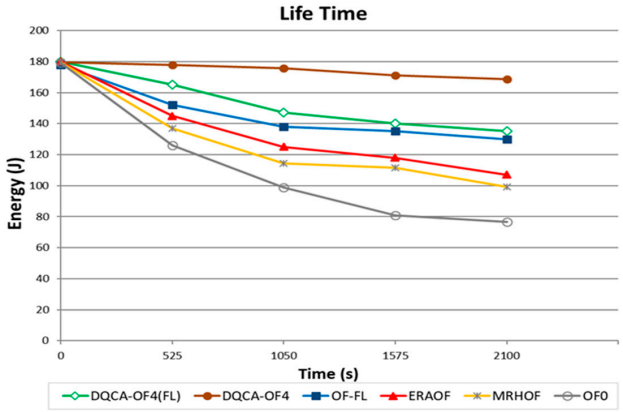


Figure 4. Energy consumption for the network nodes.

The function DQCA-OF4 achieved higher remaining energy (approximately 170 joules), at the end of the simulation, when compared to the DQCA-OF4(FL), OF-FL, ERAOF, MRHOF and OF0 objective functions which achieved consumption rates of 136 joules, 130 joules, 107 joules, 99 joules and 76 joules, respectively. The function DQCA-OF4(FL) obtained lower performance when compared to the DQCA-OF4 as a function of the energy consumed with the complexity of the fuzzy system processing.

In Figure 5, from 525 s to the end of the experiment, the DQCA-OF4(FL) function obtained a smaller delay when compared to the DQCA-OF4, OF-FL, ERAOF, MRHOF and OF0 objective functions. This occurred because the DQCA-OF4 (FL) function uses a fuzzy-based route classifier, which is not the case for the objective functions DQCA-OF4, OF-FL, ERAOF, MRHOF and OF0. Figure 5 also shows that there was no variation in results between 0 and 525 s of simulation. This is justified because the network, in this interval, is in the process of stabilization, occurring variation of the delay only after 525 s of simulation.

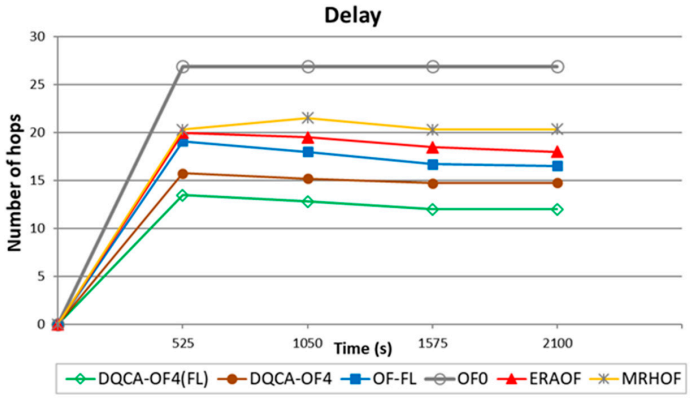


Figure 5. Message delivering delay for the sink.

For applications that require reliability in the delivery of data, Figure 6 shows that DQCA-OF1(FL) function achieved the best performance, with a delivery ratio of 94% when compared to the DQCA-OF1, OF-FL, ERAOF, MRHOF and OF0 objective functions, which achieved delivery ratios of 89%, 70%, 67.98%, 65.62% and 50.4%, respectively. This was because the expected Transmission Count (ETX) and Hop Number (NH) metrics; in this case, both with high priority, used the fuzzy-based route classifier.

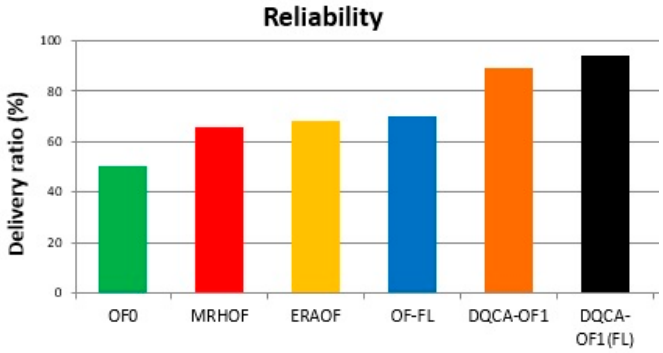


Figure 6. Delivery rate for the sent messages.

For reliability, between all the DQCA-OF applied to the classifier of routes based on fuzzy system, only the DQCA-OF1 (FL) was the more efficient when compared with the others. That is why only the DQCA-OF1 (FL) was considered for this analysis. The ETX union with *NH*, represented by DQCA-OF1 (FL), demonstrated to be relevant, since the application requiring reliability requires low ETX. In addition, the presence of the number of hops in this function allowed to increase the percentage of reliability, because the smaller the number of hops the greater the probability of delivery of the messages.

Figure 7 shows that, for applications that require priority in the quality of the delivery of data, the DQCA-OF2(FL) with EC = High achieved the best performance in the number of transmissions/retransmissions represented by ETX. The DQCA-OF2(FL) with *EC* = High got ETX = 31 when compared to the DQCA-OF2 (EC = High) functions, which achieved ETX = 46, DQCA-OF2 (*EC* = Low) with ETX = 96, OF-FL with ETX = 115, ERAOF with ETX = 125, MRHOF with ETX = 140 and OF0 with ETX = 180. This difference was achieved because the application required a high level of priority in the Energy consumed (*EC*) metric combined with the use of route classifier based on fuzzy system.

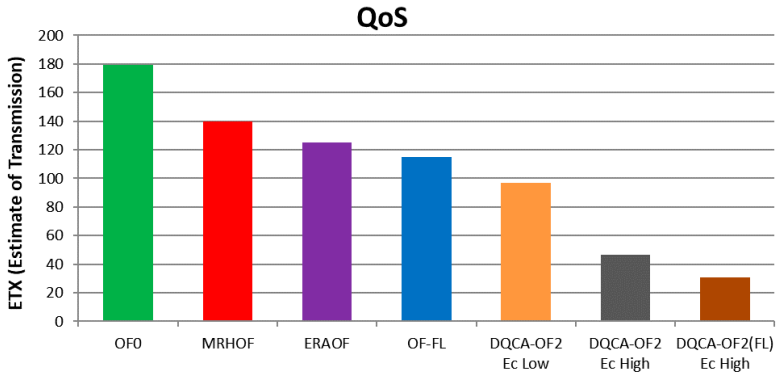


Figure 7. Quality of Service (QoS).

For the QoS, between all the DQCA-OF applied to the classifier of routes based on fuzzy system, only the DQCA-OF2 (FL) with Ec = high (high priority for energy consumed), was the more efficient when compared with the others. That is why only the DQCA-OF2 (FL) was considered for this analysis. The ETX junction with EC, represented by DQCA-OF2 (FL), proved to be important, since the application requiring QoS requires also low ETX. In addition, the presence of the metric energy consumed in this function, allows the increase of the network lifetime. These two benefits, increase the QoS of the network.

4.3. Statistical Data Analysis

To indicate the dispersion or variability of the data in terms relative to its mean value, it was used as a measure of dispersion the CVP (Pearson Variation Coefficient) with confidence degree of 95%. The CVP was used because it is a relative measure of variability and it is independent of the unit of measure used, where the observed data unit may be different and its value will not change. The CVP is given by the relation between the standard deviation and the mean referring to the data of the same sample.

$$\text{CVP} = \frac{\sigma}{\overline{X}} \cdot 100 \tag{17}$$

where:

- CVP is the Pearson Coefficient of Variation
- σ is the standard deviation of the series data
- $\overline{X}$ is the average of the series data

For the energy consumed metric, Table 3 shows that DQCA-OF4 presented lower value in the coefficient of variation with 2.15% when compared to DQCA-OF4 with the fuzzy system (8.64%), OF-FL (11.06%), ERAOF (13.89%), MRHOF (19.04%) and OF0 (29.92%). This shows that DQCA-OF4 data, in the metric energy consumed, are more homogeneous when compared to the data of the other objective functions, representing low dispersion around the mean.

Table 3. Statistical Analysis of Energy Consumption.

Objective Functions	Average (J)	Standard Deviation (J)	Coef. Var. Pearson
DQCA-OF4	174	3	2.15%
DQCA-OF4 (LF)	151	13	8.64%
OF-FL	145	16	11.06%
ERAOF	132	18	13.89%
MRHOF	126	24	19.04%
OF0	109	32	29.92%

For delay metric, Table 4 shows that DQCA-OF4 and DQCA-OF4(FL) presented similar dispersion represented by CVP. It is possible to observe that the four functions have similar stability. However, analyzing the mean, the DQCA-OF4(FL) was more efficient (see the value 11 in Table 4) when compared to the other objective functions. In this case, the lower the average, the lower the delay. The low standard deviation (shown in Tables 3 and 4) indicates that the data points tend to be close to the average, allowing variability to be expressed between the data.

Table 4. Statistical Analysis of Delay.

Objective Functions	Average (J)	Standard Deviation (J)	Coef. Var. Pearson
DQCA-OF4 (LF)	11	3	33.48%
DQCA-OF4	12	4	34.22%
OF-FL	15	5	33.62%
ERAOF	16	5	34.06%
MRHOF	17	6	33.81%
OF0	23	8	35.58%

5. Conclusion and Future Work

This paper proposes an approach to the dynamic routes selection for IoT based on context information from the applications. The approach consists the creation of four new Objective Functions (OFs). Each one of these objective functions is based on the fusion of metrics for 6LoWPAN environments, used by the RPL and described in RFC 6551. Each application may have requirements that simultaneously need up to three metrics (Expected Transmission Count (ETX) and/or Number of Hops (*NH*) and/or Energy Consumed (*EC*)). Additionally, the approach also accounts Route Classifier based on Fuzzy System, that estimating the degree of quality of the routes in IoT scenarios. The route classifier uses as input the values assigned to the metrics (present in the objective functions), ensuring consistency in decision making during the routing process. In general, it was observed that, for the scenarios evaluated, the proposed approach can increase the performance of the network to be used by an application of IoT. The approach proved to be able to increase the life time, to reduce the delay and to increase the reliability and the QoS.

It was shown that, using the proposed approach, the lifetime of the network increased and the energy consumption during the operation of the network reduced compared to the other proposals. The approach is more effective to use Objective Functions (OFs) dynamically (at run time) and to choose the path of the high quality by means of fuzzy System-Based Route Classifier. The route with the highest quality is selected as the preferred route. Thus, it is possible to observe that the proposed approach has substantial advantages in relation to the other proposals researched, mainly in the metric time of life, which obtained 168.72 J, allowing the network to function for 2100 s. In addition, provides high reliability by successfully delivering nearly 100% of data packets, low delay for data delivery and increase in QoS. An 30% improvement is attained in the network life time when using one of proposed objective function, keeping the devices alive for longer duration.

Based on the results presented, a specific objective function is recommended for each application. For application that requires a long lifetime, DQCA-OF4 is recommended. For application that requires a shorter delay, DQCA-OF4 (FL) is the more indicated. DQCA-OF1 (FL) is more recommended for applications that prioritize reliability. For application that requires QoS for data delivery, it is suggested to use DQCA-OF2 (FL).

The main contribution of this work, compared to the proposals available in the literature and considered in the study, includes an optimization of the RPL protocol to reduce the delay and to increase the reliability and QoS, guaranteeing also an increase in the network life-time. In this proposal, it is not considered the variation in the number of sensor nodes and RFID tags, the mobility of both the sink and the RFID reader and the network nodes.

Next steps include the deployment of this approach in other routing protocols for IoT, taking into account the scalability of objects that performs an IoT structure. It is also intended to carry out new studies to optimize the objective functions as well as the route classifier.

Acknowledgments: This work has been supported by National Funding from the FCT—*Fundação para a Ciência e a Tecnologia* through the UID/EEA/50008/2013 Project, by the Government of the Russian Federation, Grant 074-U01, by Finep, with resources from Funttel, Grant No. 01.14.0231.00, under the *Centro de Referência em Radiocomunicações*—CRR project of the *Instituto Nacional de Telecomunicações* (Inatel), Brazil; and by University Estacio of Sá (Teresina), Brazil, and by Brazilian National Council for Research and Development (CNPq) via Grant No. 309335/2017-5.

Author Contributions: All the authors contributed to conceive, implement, experiment, and validate the proposed approach; all the authors also contributed to wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Umamaheswari, S.; Negi, A. Internet of Things and RPL routing protocol: A study and evaluation. In Proceedings of the 2017 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 5–7 January 2017.
2. Miguel, M.; Jamhour, E.; Pellenz, M.; Penna, M. A Power Planning Algorithm Based on RPL for AMI Wireless Sensor Networks. *Sensors* **2017**, *17*, 679. [CrossRef] [PubMed]
3. Park, J.; Kim, K.; Kim, K. An Algorithm for Timely Transmission of Solicitation Messages in RPL for Energy-Efficient Node Mobility. *Sensors* **2017**, *17*, 899. [CrossRef] [PubMed]
4. Gokilapriya, V.; Bhuvaneswari, P.T.V. Analysis of RPL routing protocol on topology control mechanism. In Proceedings of the 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, India, 16–18 March 2017.
5. Alomari, A.; Phillips, W.; Aslam, N.; Comeau, F. Dynamic Fuzzy-Logic Based Path Planning for Mobility-Assisted Localization in Wireless Sensor Networks. *Sensors* **2017**, *17*, 1904. [CrossRef] [PubMed]
6. Pradeska, N.; Widyawan, W.; Najib, W.; Kusumawardani, S.S. Performance analysis of objective function MRHOF and OF0 in routing protocol RPL IPV6 over low power wireless personal area networks (6LoWPAN). In Proceedings of the 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, Indonesia, 5–6 October 2016.
7. Shakya, N.M.; Mani, M.; Crespi, N. SEEOF: Smart energy efficient objective function: Adapting RPL objective function to enable an IPv6 meshed topology solution for battery operated smart meters. In Proceedings of the 2017 Global Internet of Things Summit (GIoTS), Geneva, Switzerland, 6–9 June 2017.
8. Sousa, N.C.; Sobral, J.V.V.; Rodrigues, J.J.P.C.; Rabelo, R.A.L.; Solic, P. ERAOF: A New RPL Protocol Objective Function for Internet of Things Applications. In Proceedings of the 2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, Croatia, 12–14 July 2017.
9. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
10. Sobral, J.V.V.; Rabelo, R.A.L.; Oliveira, D.; Lima, J.C.; Araujo, H.S.; Filho, R.H. A Framework for Improving the Performance of IoT Applications. In Proceedings of the 14th International Conference on Wireless Networks, Las Vegas, NV, USA, 27–30 July 2015.
11. Baldauf, M.; Dustdar, S.; Rosenberg, F. A survey on context-aware systems. *Int. J. Ad Hoc Ubiquitous Comput.* **2007**, *2*, 263–285. [CrossRef]
12. Makris, P.; Skoutas, D.; Skianis, C. A Survey on Context-Aware Mobile and Wireless Networking: On Networking and Computing Environments' Integration. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 362–386. [CrossRef]
13. Perera, C.; Zaslavsky, A.; Christen, P.; Georgakopoulos, D. Context Aware Computing for The Internet of Things: A Survey. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 414–454. [CrossRef]
14. Sharkawy, B.; Khattab, A.; Elsayed, K.M.F. Fault-tolerant RPL through context awareness. In Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, Korea, 6–8 March 2014.

15. Couto, D.; Aguayo, D.; Bicket, J.; Morris, R. A high-throughput path metric for multi-hop wireless routing. *Wirel. Netw.* **2005**, *11*, 419–434. [CrossRef]
16. Vasseur, J.; Kim, M.; Pister, K.; Dejean, N.; Barthel, D. *Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks*; RFC 6551; IETF: Fremont, CA, USA, 2012.
17. Winter, T.; Thubert, P.; Brandt, A.; Hui, J.; Kelsey, R.; Levis, P.; Pister, K.; Struik, R.; Vasseur, J.A.R. Rpl: Ipv6 Routing Protocol for Low-Power and Lossy Networks, RFC 6550. 2012. Available online: http://tools.ietf.org/html/rfc6550 (accessed on 24 January 2018).
18. Jayaraman, P.P.; Haghighi, P.D. SA-A-WSN: Situation-aware adaptation approach for energy conservation in wireless sensor network. In Proceedings of the 2013 IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne, Australia, 2–5 April 2013; pp. 7–12.
19. Gaddour, O.; Koubaa, A.; Baccour, N.; Abid, M. OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol. In Proceedings of the 2014 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), Hammamet, Tunisia, 12–16 May 2014; pp. 367–372.
20. Chen, Y.; Chanet, J.; Hou, K.; Shi, H.; Sousa, G. A Scalable Context-Aware Objective Function (SCAOF) of Routing Protocol for Agricultural Low-Power and Lossy Networks (RPAL). *Sensors* **2015**, *15*, 19507–19540. [CrossRef] [PubMed]
21. Abbasbandy, S.; Amirfakhrian, M. The nearest trapezoidal form of a generalized left right fuzzy number. *Int. J. Approx. Reason.* **2006**, *43*, 166–178. [CrossRef]
22. Driankov, D.; Hellendoorn, H.; Reinfrank, M. *An Introduction to Fuzzy Control*; Springer: Berlin/Heidelberg, Germany, 1993; Volume XV, 316p.
23. Shaw, I.S.; Simões, M.G. *Controle e Modelagem Fuzzy*; Editora Edgard Blücher Ltda: São Paulo, Brazil, 1999.
24. Jang, J.S.R.; Sun, C.T.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1997; pp. 1482–1484.
25. Reznik, L. *Fuzzy Controllers Handbook*, 1st ed.; Newnes: Oxford, UK, 1997; p. 240.
26. Takagi, T.; Sugeno, M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man Cybern.* **1985**, *15*, 116–132. [CrossRef]
27. Osterlind, F.; Dunkels, A.; Eriksson, J.; Finne, N.; Voigt, T. Cross-Level Sensor Network Simulation with COOJA. In Proceedings of the 2006 31st IEEE Conference on Local Computer Networks, Tampa, FL, USA, 14–16 November 2006.
28. Dunkels, A.; Osterlind, F.; Tsiftes, N.; He, Z. Software-based on-line energy estimation for sensor nodes. In Proceedings of the 4th Workshop on Embedded Networked Sensors (Emnets 2007), Cork, Ireland, 25–26 June 2007; pp. 28–32.
29. Welbourne, E.; Battle, L.; Cole, G.; Gould, K.; Rector, K.; Raymer, S.; Balazinska, M.; Borriello, G. Building the Internet of Things Using RFID: The RFID Ecosystem Experience. *IEEE Internet Comput.* **2009**, *13*, 48–55. [CrossRef]