



Amostragem Adaptativa em Redes De Sensores

AGOSTINHO ORLANDO FERREIRA FERRAZ

Outubro de 2015

AMOSTRAGEM ADAPTATIVA EM REDES DE SENSORES

Agostinho Orlando Ferreira Ferraz



Mestrado em Engenharia Eletrotécnica e de Computadores

Área de Especialização de Telecomunicações

Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

2015

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Eletrotécnica e de Computadores

Candidato: Agostinho Orlando Ferreira Ferraz, N° 1130197, 1130197@isep.ipp.pt

Orientação científica: Prof. Doutor Jorge Botelho Costa Mamede, jbm@isep.ipp.pt

Empresa: Instituto de Telecomunicações

Supervisão: Doutora Tânia Cláudia dos Santos Pinto Calçada, tcalcada@fe.up.pt



Mestrado em Engenharia Eletrotécnica e de Computadores

Área de Especialização de Telecomunicações

Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

30 de outubro de 2015

“A inovação distingue os líderes dos seguidores.”

Steve jobs

Agradecimentos

Serve o presente momento para homenagear e agradecer a todas as pessoas e instituições que contribuíram, direta ou indiretamente, para a elaboração e sucesso desta dissertação, evidenciando as seguintes:

Ao meu orientador, Professor Doutor Jorge Mamede, uma palavra de amizade e de profundo agradecimento pelo seu apoio contínuo e incansável, pela excelente orientação e conselhos prestados que contribuíram de uma forma positiva para o desenrolar deste trabalho.

À minha coorientadora, Doutora Tânia Calçada, pela disponibilidade, simpatia e informações necessárias para a realização deste projeto no Instituto de Telecomunicações.

Aos investigadores do Instituto de Telecomunicações nomeadamente, Pedro Cunha, Carlos Penichet, Tiago Lourenço e Yuniur Luis pela disponibilidade de recursos, colaboração e apoio prestados no desenvolvimento desta dissertação.

Aos meus pais e irmão pelo inestimável apoio familiar que contribuiu na minha educação e pelo vosso esforço diário para a realização da minha vida profissional e pessoal.

A todos os meus colegas e amigos, pelo apoio e amizade nas diversas batalhas encontradas ao longo destes dois anos de mestrado.

A toda a restante família pela ajuda e preocupação que sempre demonstraram, e em especial a minha prima Dina Ferraz, que sempre me ajudou no preenchimento dos vários requisitos existentes numa faculdade, a ti, o meu muito obrigado.

Aos pais da minha namorada, como segundos pais que são, que ajudaram em várias etapas da minha vida académica.

Em especial, à minha namorada Patrícia Leitão, que nos momentos mais difíceis soube ter compreensão e dedicação para me escutar e me guiar perante os mesmos. Um muito obrigado não chega só! Simplesmente tudo o que sou deve-se a ti.

Resumo

Uma nova área tecnológica está em crescente desenvolvimento. Esta área, denominada de internet das coisas, surge na necessidade de interligar vários objetos para uma melhoria a nível de serviços ou necessidades por parte dos utilizadores. Esta dissertação concentra-se numa área específica da tecnologia internet das coisas que é a sensorização. Esta rede de sensorização é implementada pelo projeto europeu denominado de *Future Cities* [1] onde se cria uma infraestrutura de investigação e validação de projetos e serviços inteligentes na cidade do Porto. O trabalho realizado nesta dissertação insere-se numa das plataformas existentes nessa rede de sensorização: a plataforma de sensores ambientais intitulada de *UrbanSense*. Estes sensores ambientais que estão incorporados em *Data Collect Unit* (DCU), também denominados por nós, medem variáveis ambientais tais como a temperatura, humidade, ozono e monóxido de carbono. No entanto, os nós têm recursos limitados em termos de energia, processamento e memória. Apesar das grandes evoluções a nível de armazenamento e de processamento, a nível energético, nomeadamente nas baterias, não existe ainda uma evolução tão notável, limitando a sua operacionalidade [2].

Esta tese foca-se, essencialmente, na melhoria do desempenho energético da rede de sensores *UrbanSense*. A principal contribuição é uma adaptação do protocolo de redes Ad Hoc OLSR (Optimized Link State Routing Protocol) para ser usado por nós alimentados a energia renovável, de forma a aumentar a vida útil dos nós da rede de sensorização. Com esta contribuição é possível obter um maior número de dados durante períodos de tempo mais longos, aproximadamente 10 horas relativamente às 7 horas anteriores, resultando numa maior recolha e envio dos mesmos com uma taxa superior, cerca de 500 KB/s. Existindo deste modo uma aproximação analítica dos vários parâmetros existentes na rede de sensorização. Contudo, o aumento do tempo de vida útil dos nós sensores com recurso à energia renovável, nomeadamente, energia solar, incrementa o seu peso e tamanho que limita a sua mobilidade. Com o referido acréscimo a determinar e a limitar a sua mobilidade exigindo, por isso, um planeamento prévio da sua localização. Numa primeira fase do trabalho analisou-se o consumo da DCU, visto serem estes a base na infraestrutura e comunicando entre si por WiFi ou 3G. Após uma análise dos protocolos de routing com

suporte para parametrização energética, a escolha recaiu sobre o protocolo OLSR devido à maturidade e compatibilidade com o sistema atual da DCU, pois apesar de existirem outros protocolos, a implementação dos mesmos, não se encontram disponível como *software* aberto.

Para a validação do trabalho realizado na presente dissertação, é realizado um ensaio prévio sem a energia renovável, para permitir caracterização de limitações do sistema. Com este ensaio, tornou-se possível verificar a compatibilidade entre os vários materiais e ajustamento de estratégias. Num segundo teste de validação é concretizado um ensaio real do sistema com 4 nós a comunicar, usando o protocolo com eficiência energética. O protocolo é avaliado em termos de aumento do tempo de vida útil do nó e da taxa de transferência.

O desenvolvimento da análise e da adaptação do protocolo de rede Ad Hoc oferece uma maior longevidade em termos de tempo de vida útil, comparando ao que existe durante o processamento de envio de dados. Apesar do tempo de longevidade ser inferior, quando o parâmetro energético se encontra por omissão com o fator 3, a realização da adaptação do sistema conforme a energia, oferece uma taxa de transferência maior num período mais longo. Este é um fator favorável para a abertura de novos serviços de envio de dados em tempo real ou envio de ficheiros com um tamanho mais elevado.

Palavras-Chave

OLSR, Protocolos de rede Ad Hoc, Eficiência energética, Energia renovável, Análise energética, Sensores ambientais, WiFi, Testbed.

Abstract

A new technology area is in increasing development, this area called Internet of things arises the need to link various objects to raising the level of services or needs from users. This dissertation focuses on a specific area of internet technology of the things that is sensing. This sensing network is implemented by the European project called Future Cities [1] where you create a research infrastructure and validation of smart projects and services in Porto. The work done in this thesis is part of an existing platforms that network sensing: the environmental sensor platform titled UrbanSense. These environmental sensors are incorporated into Collect Data Unit (DCU), also known by us, measure environmental variables such as temperature, humidity, ozone and carbon monoxide. However, we have limited resources in terms of power, processing and memory. Despite major developments in terms of storage and processing, the energy level, especially in batteries, there is still such a remarkable evolution, limiting their operation.

This thesis will focus on improving the energy performance of UrbanSense sensor network. The main contribution is an adaptation of networks Ad Hoc protocol OLSR (Optimized Link State Routing Protocol) to be used by us powered by renewable energy, in order to extend the life of the nodes of the sensing network. With this contribution is possible to get more data for longer periods of time, approximately 10 hours, resulting in increased collection and sending of the same with a rate greater than about 500 KB / s. There an analytical approach of the various parameters in the sensing network. However, increasing the lifetime of the sensor nodes using renewable energy, particularly solar energy increases its weight and size which limits its mobility. With that extra to determine and limit their mobility demanding, so advance planning on your location. In a first phase of the study analyzed the consumption of DCU, as these are the basis on infrastructure and communicating with each other by WiFi or 3G. After an analysis of routing protocols that support energy parameter, the choice fell on the OLSR protocol due to maturity and compatibility with the current system of DCU, because although there are other protocols, their implementation, if any, is not available as open source software.

For the validation work performed in this work is done, a test without renewable energy, for the characterization of system limitations. With this test, it became possible to verify the compatibility between the various materials and adjustment strategies. In a second validation test is implemented an actual test system with 4 nodes to communicate using the protocol with energy efficiency. The protocol is evaluated in terms of increasing the lifetime of the node and throughput.

The development of the analysis and adaptation of the Ad Hoc network protocol offers greater longevity in terms of lifetime compared when there is upstream processing. Despite the longevity of time is less than when the energy parameter is by default with the factor 3, the realization of the system adaptation as energy offers greater throughput over a longer period. This is a favorable factor for the opening of new real-time data push services or sending files with a higher size.

Keywords

OLSR, Ad Hoc protocol, Energy aware, Renewable energy, Testbed, Environmental sensors, Energy analysis, WiFi.

Índice

AGRADECIMENTOS	I
RESUMO	III
ABSTRACT	V
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABELAS	XI
ÍNDICE DE EXCERTOS DE CÓDIGO	XIII
ACRÓNIMOS	XV
1. INTRODUÇÃO	1
1.1. OBJETIVOS.....	2
1.2. ORGANIZAÇÃO DA DISSERTAÇÃO	2
2. PLATAFORMA URBANSENSE	5
2.1. ARQUITETURA GERAL DA URBANSENSE.....	6
2.2. ARQUITETURA DOS SENSORES	7
2.3. IMPLEMENTAÇÃO DAS DCU	8
2.4. RESUMO.....	9
3. TECNOLOGIAS	11
3.1. SOLUÇÕES ANÁLOGAS.....	11
3.2. SISTEMAS FOTOVOLTAICOS	12
3.3. BATERIA	14
3.4. COMUNICAÇÕES AD HOC	16
3.4.1 <i>Protocolos pró ativos</i>	17
3.4.2 <i>Protocolos reativos</i>	18
3.4.3 <i>Protocolos com eficiência energética</i>	19
3.5. RESUMO.....	20
4. PROTOCOLOS DE ROUTING AD HOC COM EFICIÊNCIA ENERGÉTICA	23
4.1. ESTRATÉGIAS PARA MINIMIZAR O CONSUMO ENERGÉTICO EM MANETS	23
4.2. AODV COM EFICIÊNCIA ENERGÉTICA.....	24
4.3. OLSR COM EFICIÊNCIA ENERGÉTICA.....	25
4.4. SELEÇÃO DO PROTOCOLO COM EFICIÊNCIA ENERGÉTICA	26

4.5.	OLSR COM <i>WILLINGNESS</i> DINÂMICO	27
4.6.	RESUMO	28
5.	IMPLEMENTAÇÃO DE REDE AD HOC COM GESTÃO ENERGÉTICA	31
5.1.	ANÁLISE DE CONSUMO DA DCU	31
5.1.1	<i>Sensor de Corrente</i>	32
5.1.2	<i>Análise do consumo energético</i>	33
5.2.	HARDWARE	39
5.2.1	<i>Implementação do MoPi</i>	40
5.2.2	<i>Instalação do sistema de energia solar</i>	42
5.3.	SOFTWARE DE REDES	43
5.3.1	<i>Instalação do protocolo OLSR</i>	43
5.3.2	<i>Configuração do protocolo OLSR</i>	44
5.4.	ENSAIO DO SISTEMA	45
5.5.	RESUMO	48
6.	ANÁLISE E DISCUSSÃO DOS RESULTADOS	51
6.1.	EXPERIMENTAÇÃO DO SISTEMA COMPLETO.....	52
6.2.	ANÁLISE DOS RESULTADOS	54
6.2.1	<i>Tráfego de pacotes recebidos e perdidos no OLSR com eficiência energética</i>	54
6.2.2	<i>Tempo de adaptação do OLSR com eficiência energética</i>	55
6.2.3	<i>Eficiência do OLSR com eficiência energética</i>	56
6.2.4	<i>Taxa de transferência do OLSR com eficiência energética</i>	59
6.3.	RESUMO	61
7.	CONCLUSÕES	63
7.1.	TRABALHO FUTURO	64
	REFERÊNCIAS DOCUMENTAIS	65
	ANEXO A. CÓDIGO DE ADAPTAÇÃO ENERGÉTICA	71

Índice de Figuras

Figura 1	Arquitetura da <i>UrbanSense</i>	6
Figura 2	Caracterização da DCU [1]	7
Figura 3	Instalação de uma DCU na cidade [1].....	8
Figura 4	Módulo fotovoltaico em paralelo [14]	14
Figura 5	Módulo fotovoltaico em série [14].....	14
Figura 6	Exemplo de execução do protocolo OLSR [21]	17
Figura 7	Disseminação da mensagem RREQ do protocolo AODV [14]	18
Figura 8	Transmissão da mensagem RREP do protocolo AODV [14].....	18
Figura 9	Mensagem RREQ do protocolo AODV [26]. (a) Mensagem RREQ original. (b) Mensagem RREQ com função energética.....	25
Figura 10	Mensagem HELLO do protocolo OLSR [19].....	26
Figura 11	Rede Ad Hoc com escolha de MPR com o parametro energético.....	28
Figura 12	Sensor de Corrente INA219 [39].	32
Figura 13	Arquitetura do sistema de medição de consumo.....	33
Figura 14	Consumo dos vários sensores presentes na DCU. (a) Consumo do sensor de Temperatura. (b) Consumo do sensor de Humidade. (c) Consumo do sensor de Precipitação. (d) Consumo do sensor da Velocidade do Vento. (e) Consumo do sensor de Direcção de Vento. (f) Consumo do sensor de Luminosidade. (g) Consumo do sensor de Particulas. (h) Consumo do sensor de CO. (i) Consumo do sensor de NO2. (j) Consumo do sensor de O3.	36

Figura 15	Consumo da placa WiFi.....	37
Figura 16	Consumo dos vários equipamentos existentes numa DCU.	38
Figura 17	Esquemática da interligação dos vários equipamentos.	39
Figura 18	Ilustração da estrutura do MoPi [42].	40
Figura 19	Curva de descarga da bateria <i>Lead-Acid</i> incluída no MoPi [15].	41
Figura 20	Esquema de montagem do sistema solar [43].....	42
Figura 21	Esquema de ensaio do cenário da testbed.....	45
Figura 22	Ilustração do pré-teste em laboratório.	46
Figura 23	Ilustração da testbed em fase experimental.	52
Figura 24	Esquemática da testbed.	53
Figura 25	Validação do funcionamento de rotas do protocolo OLSR.	53
Figura 26	Tráfego de pacotes dos vários nós em teste.....	54
Figura 27	Tempo de adaptação do protocolo em relação à vertente energética.	55
Figura 28	Nó com o parâmetro de <i>Willingness</i> =1.....	56
Figura 29	Nó com o parâmetro de <i>Willingness</i> =7.....	57
Figura 30	Nó com o parâmetro estático de <i>Willingness</i> =3.....	58
Figura 31	Comparação dos diferentes parâmetros de <i>Willingness</i>	58
Figura 32	Taxa de transferência entre parâmetro dinâmico e estático.....	59

Índice de Tabelas

Tabela 1	Comparação dos vários tipos de baterias existentes (adaptado de [15])	15
Tabela 2	Comparativos entre os vários Protocolos Ad Hoc	20
Tabela 3	Consumo de cada sensor presente na DCU e o seu consumo Total.	37
Tabela 4	Taxa de transferência com o parâmetro dinâmico e estático e o seu calculo aritmético.	60

Índice de Excertos de Código

Código 1	Recolha e escrita dos dados do INA219.....	34
Código 2	Obtenção do protocolo OLSR através do repositório apt-get.....	43
Código 3	Obtenção e instalação do protocolo OLSR.....	44
Código 4	Alterações efetuadas no ficheiro de configuração do OLSRD.....	44
Código 5	Excerto para realizar a adaptação energética no protocolo OLSR.	47

Acrónimos

ABR	–	Associativity-Based Routing
AODV	–	Ad Hoc On-Demand Distance Vector
APM	–	Advanced Power Management
BIOS	–	Basic Input/Output System
DCU	–	Data Collecting Units
DSDV	–	Destination Sequence Distance Vector
DSR	–	Dynamic Source Routing
EA	–	Energy Aware
FSR	–	Fisheye State Routing
GPIO	–	General Purpose Input/output
GND	–	Ground
LED	–	Light-Emitting Diodes
LMR	–	Lightweight Mobile Routing
MANET	–	Mobile Ad Hoc Network
MPR	–	MultiPoint Relays
OLSR	–	Optimized Link State Routing
PAMAS	–	Power-Aware Multipath Access Protocol
RERR	–	Route Error

- RREP – Route Reply
- RREQ – Route Request
- Rx – Receiver
- TC – Topology Control
- Tx – Transmitter
- WRP – Wireless Routing Protocol

1. INTRODUÇÃO

A popularização de tecnologias sem fios e sensores, aliada à requisição de novas aplicações demanda a criação de uma era omnipresente e inteligente denominada de Internet das coisas. A Internet das coisas é composta por um conjunto de tecnologias que fornecem conectividade em todos os momentos a objetos e máquinas. Este conceito é baseado no princípio que os mais diversos objetos, equipados com dispositivos eletrónicos, interagem e cooperam uns com os outros através da utilização de ligações sem fios para garantir uma comunicação ubíqua.

Com este conceito a ganhar mais importância e a ser cada vez mais expandido para as mais variadas áreas, é necessário procurar e inovar soluções para tornar cada vez mais autónomas estas redes de sensorização. Existem vários projetos com este conceito bem presente e com sistemas capazes de interagir entre si [1, 3, 4, 5]. Esta dissertação está incluída no âmbito de um projeto europeu Cidades do futuro [1] que propõe fazer da cidade do Porto um laboratório à escala urbana, onde vários investigadores e empresas podem testar produtos, inovações e serviços, de forma a obter a perceção de sustentabilidade da mobilidade, qualidade de vida e parametrização do meio ambiente.

A presente dissertação apresenta o trabalho na rede de sensorização, recaindo sobre melhorias de ajuste energético garantindo maiores larguras de banda durante períodos

longos. De modo, a haver um equilíbrio entre o tempo de vida útil do nó e a energia residual consumida.

No âmbito deste projeto existe uma rede de sensorização, denominada UrbanSense, que inclui diversos sensores como luminosidade, temperatura, humidade entre outros [6, 1]. Esta rede de sensorização tem como opção a criação de rede mesh, possibilitando uma forma de comunicarem entre si. Com esta rede, é possível estender a cobertura de rede em zonas onde não há redes infraestruturas.

Contudo, esta rede não tem autonomia energética, sendo um dos objetivos nesta dissertação perceber o custo energético de cada nó na rede e criar uma solução eficiente recorrendo a energia renovável. A presente dissertação apresenta o trabalho na rede de sensorização, tornando mais autónoma na comunicação e interação com os vários equipamentos presente nas redes. Existindo uma preocupação no tempo de vida útil de cada nó, havendo uma procura para o melhoramento do aproveitamento da bateria existente em cada nó consoante a utilização do nó.

1.1. OBJETIVOS

O objetivo principal desta dissertação é obter os vários consumos dos sensores existentes na rede e realizar uma adaptação das comunicações entre os nós de acordo com a energia existente em cada nó. Dada a complexidade intrínseca a este objetivo, surgiu a necessidade de o subdividir em múltiplas tarefas de realização mais simples, tais como:

- Estudo do consumo do nó da rede;
- A avaliação dos protocolos de comunicações Ad Hoc para a utilização na rede nós;
- Ajuste dos parâmetros dos protocolos de comunicações com impacto no consumo energético;
- Especificação, desenvolvimento e avaliação de um nó autónomo e sustentável;
- Aumento da largura de banda em conformidade com o tempo de vida útil do nó.

1.2. ORGANIZAÇÃO DA DISSERTAÇÃO

Este documento é composto por 7 capítulos. No capítulo 1 são expostas as motivações, objetivos e a organização do presente documento.

No capítulo seguinte, 2, é apresentado um contexto sobre o projeto europeu *Future Cities* e a rede de sensorização presente no mesmo, bem como o objetivo e a motivação para a realização da dissertação.

No capítulo 3 apresenta-se o estado de arte das diversas áreas existentes neste projeto como as várias soluções presentes, o estudo da bateria a utilizar na *testbed*, estudo do painel solar e a sua eficiência e por fim um estudo dos vários protocolos de rede Ad Hoc com a vertente energética. Sendo decidido quais os protocolos a serem estudados para a implementação da dissertação.

No capítulo 4 é realizado um estudo pormenorizado sobre os protocolos de rede Ad Hoc com vertente energética, de forma a perceber o seu funcionamento a nível de rede e a sua possível implementação. É selecionado o protocolo a considerar na presente dissertação e investigada a (s) melhoria (s) a realizar.

No capítulo 5 são apresentados os principais passos de implementação da *testbed* sendo dividido em duas áreas, sendo uma a nível energético e outra a nível de comunicações. A nível energético recorre-se a equipamentos de análise de consumos, como o INA219 e o MoPi. A nível de comunicações é demonstrado o desenvolvimento de vários excertos de código para o funcionamento protocolar da *testbed* e a limitação encontrada no protocolo de rede Ad Hoc num ensaio inicial.

No capítulo 6 são descritos os vários testes realizados: i) longevidade, ii) tempo de adaptação do sistema e iii) a taxa de transferência para a validação da adaptação realizada no protocolo de comunicações consoante a energia existente no nó.

No último capítulo, 7, são agrupadas as principais conclusões sobre o trabalho descrito neste documento e perspetivados futuros desenvolvimentos.

2. PLATAFORMA URBANSense

Num mundo cada vez mais dinâmico e em constante movimentação surge cada vez mais a necessidade de tornar inteligente a sua sustentabilidade. Assim sendo, a forma inteligente de o fazer, é utilizar a tecnologia existente e a sua inovação, de maneira a ser possível obter uma melhor qualidade de vida e ajudar nas possíveis decisões existentes em tempo real ou a longo prazo. Desta forma, existem métodos que ajudam a perceber as dificuldades e as melhorias possíveis de implementar numa cidade recorrendo a sensores ambientais, sensor de ruído entre muito outros.

Apesar de existir em muitas cidades projetos com a utilização de sensores ambientais, como no caso [6, 4, 3, 7], muitos sensores ambientais estão posicionados em locais estratégicos com os dados a serem guardados localmente, requerendo visitas periódicas para a sua recolha e possível manutenção. Isto numa primeira tentativa torna-se apetecível, mas não flexível nem rentável a longo prazo. As possíveis perdas existentes quando existe a manutenção dos sensores são o maior obstáculo para se tornar uma cidade mais inteligente. Assim, surge no âmbito de um projeto europeu denominado de *Future Cities* [1] implementado na cidade do Porto, a solução de sensorização *UrbanSense* [1] em que os vários sensores conseguem enviar, automaticamente, os seus dados através de redes WiFi ou 3G.

2.1. ARQUITETURA GERAL DA URBANSENSE

A rede de sensores denominada por *UrbanSense* [1] consiste em nós fixos, chamados de Data Collecting Units (DCUs). Cada DCU é composta por uma vasta gama de sensores, tais como sensores de temperatura, ruído, qualidade do ar, luminosidade e sensores de imagem que tem como objetivo analisar imagens recolhidas a fim de contar pessoas num determinado local. Estes sensores conseguem comunicar através de duas redes possíveis: numa primeira observação, tentam enviar dados através de uma rede sem fios, numa segunda tentativa transmitem através de uma conexão 3G. Os dados depois de enviados são recebidos no servidor da plataforma, onde são armazenados e posteriormente examinados. Para uma melhor compreensão da arquitetura existente no projeto *UrbanSense* a Figura 1 ilustra um diagrama em que é possível identificar os vários processos.

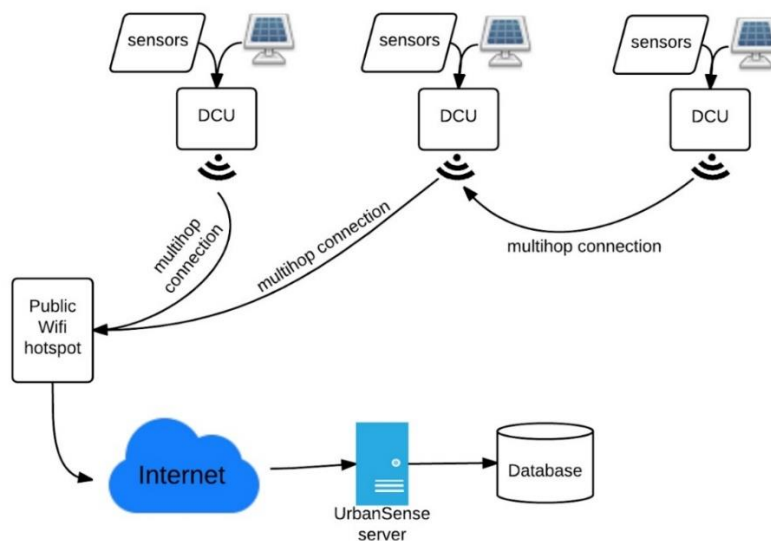


Figura 1 Arquitetura da *UrbanSense*

Como é possível observar na ilustração anterior, as informações sobre os vários sensores são guardadas localmente, sendo isto uma mais-valia no caso de haver uma falha de energia ou perda de conectividade. Assim, existem as unidades localizadas em locais pré definidos e com características relevantes para a cidade do Porto, em termos de processamento das imagens recolhidas para a contagem do número de pessoas, da temperatura, qualidade do ar presente e ruído.

A camada das comunicações é a mais favorável em relação à utilização de envio dos dados recolhidos pelos sensores, visto que muitas vezes conseguem conectar-se a redes sem fios públicas existentes pela cidade do Porto ou a utilização da rede 3G, em caso de não existir uma rede pública ou um nó perto para comunicar por *multihop*. Caso a sua utilização seja pelo WiFi, os dados são transmitidos diretamente para a *cloud* através da internet.

2.2. ARQUITETURA DOS SENSORES

A DCU é composta por três componentes principais como é demonstrado na Figura 2: parte de processamento e controlo, parte de comunicação e a parte de sensorização. A placa de sensores constitui-se pelos sensores ambientais conectados e a monitorização dos mesmos. Esta placa é ligada em separado, devido à necessidade de criação de um involucrio que permita que os sensores entrem em contato com o meio ambiente, mas ao mesmo tempo consigam proteger o circuito eletrónico.

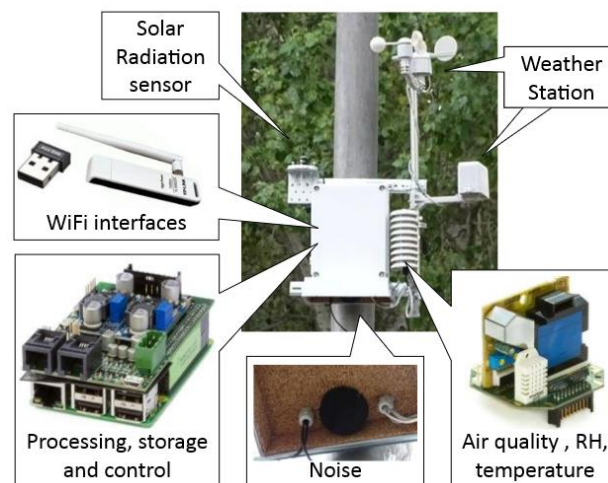


Figura 2 Caracterização da DCU [1]

A estação meteorológica é composta por um sensor de direção de vento, um anemómetro e um pluviómetro que executam, respetivamente, medições de velocidade do vento e chuva.

O invólucro abrange um sensor de temperatura e humidade relativa do ar e quatro sensores de qualidade do ar que medem partículas, ozono, dióxido de nitrogénio e monóxido de carbono. O DCU inclui também um harmonómetro que mede o ruído ambiente. É também possível verificar a existência de um sensor de radiação solar.

O controlo da DCU é feito na placa de processamento na qual existe um *Raspberry Pi* [8] (modelo B+) que assegura a recolha de dados para uma base de dados e executa os vários pedidos de medidas aos sensores. Neste dispositivo é possível armazenar alguns dos dados enquanto não é efetuando o envio para a cloud através dos métodos acima referidos.

Por último, a parte da comunicação é constituída por uma placa WiFi que realiza as comunicações com os *hotspots* públicos e uma placa 3G para as situações onde não existe comunicação por WiFi para o envio dos dados para o servidor. As comunicações via WiFi utilizam protocolos de rede Ad Hoc/Mesh.

2.3. IMPLEMENTAÇÃO DAS DCU

Este projeto já se encontra em vários pontos da cidade como é possível ver na Figura 3, contudo continua a ser necessária energia elétrica proveniente de um poste elétrico para alimentar cada DCU. Assim torna-se importante expandir este projeto para várias zonas em que não seja necessária corrente elétrica para haver uma maior dispersão de sensores e uma análise sensorial mais completa de múltiplos locais da cidade.

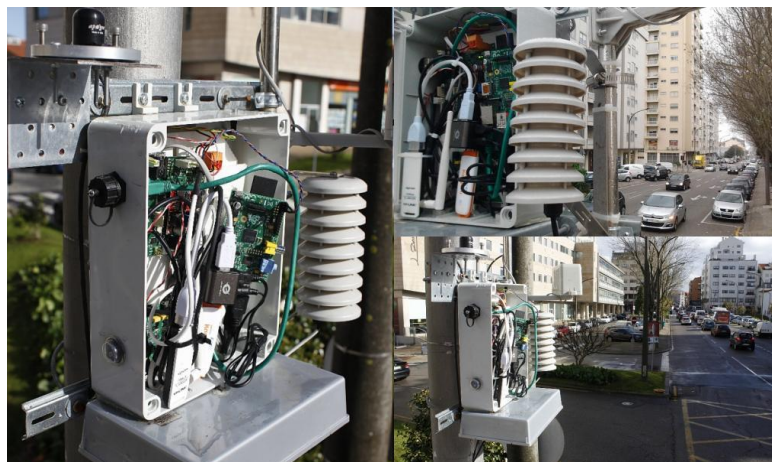


Figura 3 Instalação de uma DCU na cidade [1]

Para realizar esta expansão é necessário efetuar um levantamento do consumo energético dos vários sensores ambientais e das comunicações existentes, para posteriormente identificar e criar uma solução para um completo funcionamento do sistema recorrendo a fontes de energia alternativas, como por exemplo usando baterias. Numa primeira aproximação, foi realizada uma dissertação [9] em que os autores se propuseram a colocar as unidades de operação sem alimentação por ligação elétrica, de forma a assegurar as comunicações entre as várias DCU.

Criando a necessidade de catalogar os vários consumos dos sensores ambientais existentes e das comunicações realizadas na *UrbanSense*. A solução ideal é delineada pela utilização de baterias com recurso a energias renováveis, nomeadamente, energia solar. Este recurso renovável serve de carregamento energético da bateria durante o dia, e evitando a utilização de recursos da bateria nessa altura. Durante a noite o funcionamento das unidades é assegurado pelas baterias.

2.4. RESUMO

A arquitetura deste projeto é bastante complexa e por esta razão torna-se importante possuir o conhecimento sobre o seu consumo energético e de documentar cada sensor de forma a poder decidir qual a melhor via de abordar o problema da alimentação do sistema, quando o mesmo estiver em níveis baixos. O objetivo da dissertação consiste em arranjar uma solução para o problema existente neste projeto europeu, inquirindo a utilização de protocolos de rede Ad Hoc com a vertente energética juntamente com baterias e painel fotovoltaico.

Torna-se importante adquirir este conhecimento dos vários sensores e a forma como o projeto é inserido no seio da sociedade, para que a escolha dos vários testes a serem realizados, para obtenção do consumo, sejam os mais parecidos com o funcionamento típico do sistema em si.

3. TECNOLOGIAS

Várias instituições estão a melhorar a sua investigação em tecnologia procurando desenvolver sistemas com dimensões cada vez mais pequenas e com desempenho superior. Um conceito na área tecnológica tem surgido com possibilidades de implementação, internet das coisas [6, 3]. Este conceito recorre sobretudo à utilização da sensorização para recolher dados/informação e modulá-los de forma perceptível para o ser humano.

3.1. SOLUÇÕES ANÁLOGAS

Com a área de sensorização ambiental a ganhar cada vez mais relevância, existem diversos projetos emergentes no mesmo âmbito que o projeto Future Cities, numa tentativa de desenvolver e melhorar cada vez mais as tecnologias existentes e assim ampliar a gama de serviços e funcionalidades disponíveis.

O *SmartSantander* [6] é um projeto fundado pela União Europeia no sétimo programa de investigação (FP7). É realizado em Espanha e integra um centro de investigação para apoio em aplicações e serviços típicos de uma cidade inteligente. O projeto prevê a implementação de 20.000 sensores em Belgrado, Guildford, Lübeck e Santander, explorando uma grande variedade de tecnologias. A infraestrutura é definida pela utilização de redes sem fios nos diversos sensores espalhados.

O Observatório do ambiente urbano de Málaga criou um projeto denominado de CitySense [4], que permite a monitorização ambiental através dos *smartphones* dos utilizadores que tiverem a aplicação desenvolvida pelo observatório. A comunicação é realizada através de redes sem fios, tanto para envio dos dados como para sincronização. Com esta infraestrutura é possível obter dados sobre os níveis de CO, NO₂, ozono entre muitos outros.

Sense Smart City [3] é um projeto que realiza pesquisas sobre novas oportunidades de negócio para desenvolver e aumentar a inovação nas cidades urbanas, de forma a tornar as mesmas inteligentes proporcionando melhorias na qualidade de vida dos cidadãos. Este projeto dispõe de várias soluções utilizando sensores para monitorização da meteorologia, tráfego, energia, etc.

O Distributed Embedded System [10, 11, 12] (DES) desenvolveu uma implementação real com utilização de redes *multihop* híbridas. O DES assenta em duas redes: sem fios e sensorização. As duas redes dispõem de 15 nós equipados com tecnologia 802.11 a/b/g e uma placa denominada de MSB-A2 que incorpora sensores de temperatura e humidade. O DES-Testbed permite comparar estudos de simulação com experiências do mundo real. Para facilitar a especificação, execução e avaliação de experiências em cenários complexos.

VITAL [5] é um projeto que consiste no desenvolvimento de uma aplicação capaz de interagir em tempo real com várias aplicações e serviços implementados nas cidades inteligentes. Sendo uma integração e uma interação entre os vários serviços existentes e servindo como uma base para possíveis inovações, este projeto facilita à comunidade de desenvolvimento de aplicações reutilizar os seus produtos/aplicações em diferentes cidades, reduzindo o esforço e o tempo de desenvolvimento.

3.2. SISTEMAS FOTOVOLTAICOS

A obtenção de energia de uma forma mais limpa e rentável têm sido cada vez mais motivada com os vários estudos relacionados com o aquecimento global e o esgotamento, a curto prazo, das fontes de energia não renováveis, como por exemplo, o petróleo. Um grande avanço surgiu com a invenção das células fotovoltaicas que permitem a obtenção de energia através do sol. Nestas células, a radiação solar é convertida em eletricidade, explorando o efeito fotovoltaico dos materiais.

Cada vez mais, as células, módulos e sistemas fotovoltaicos são mais eficientes, mais ergonómicos e com um investimento de retorno consideravelmente rápido. A diversidade existente para as vastas áreas comerciais ou domésticas, permite que esta tecnologia consiga uma elevada expansão e reduções de custo, tanto a nível fabril como a nível de consumidor final.

Com esta evolução é possível escolher de forma adequada o tipo de células solares a adquirir conforme os requisitos de cada projeto. Os três principais tipos de células solares [13] são:

- Células monocristalinas de Silício [Si-m];
- Células policristalinas de Silício [Si-p];
- Células de Silício Amorfo [Si-a];

As células monocristalinas foram as primeiras células a serem criadas para a obtenção da energia solar. O seu rendimento é consideravelmente bom (aproximadamente 23% em laboratório), mas em contrapartida a sua técnica fabril é dispendiosa devido ao uso de material em estado puro e estrutura cristalina perfeita. Outra desvantagem é a utilização de técnicas complexas na sua produção [13, 14].

Numa segunda geração das células solares, surgiram as células policristalinas com a grande vantagem de terem um custo de produção inferior relativamente as células de primeira geração. Contudo, apesar da grande redução de custos, o seu rendimento elétrico também é reduzido, cerca de 18% em laboratório, devido à imperfeição do cristal a nível fabril [13].

Por último, as células de silício amorfo tem um processo fabril mais simples devido a estas células serem compostas por um suporte de vidro ou de uma outra matéria sintética, e com uma espessura muito fina possibilitando o seu uso como material de construção, retirando assim aproveitamento energético para edifícios. Contudo o seu rendimento energético é muito reduzido, aproximadamente 10% em laboratório [13, 14].

De forma a ser possível obter mais energia agrupam-se as células fotovoltaicas em serie ou paralelo, formando um módulo. A combinação das células em paralelo (Figura 4) é utilizada para alcançar uma corrente de aproximadamente de 3A e manter o nível de tensão. Isto deve-se ao facto de que ao criar o módulo em paralelo, soma-se as correntes de cada célula mas mantem-se a tensão [14].

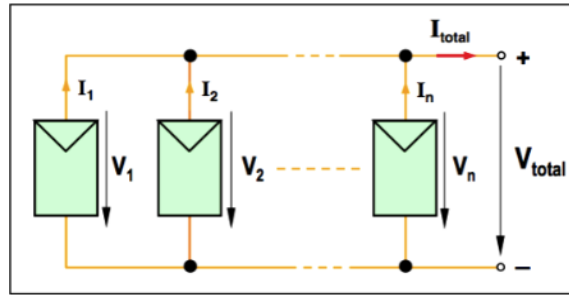


Figura 4 Módulo fotovoltaico em paralelo [14]

Como é possível observar na imagem anterior, cada célula i contribuiu com a sua corrente (I_i) resultando que, na saída do módulo, a corrente total (I_{total}) é igual ao somatório de todas as correntes das células fotovoltaicas que dele fazem parte. Já a tensão total é equivalente a tensão referida em cada célula.

A combinação em série (Figura 5) de células fotovoltaicas é a mais utilizada por conseguir criar uma tensão até 12V, o que possibilita o seu armazenamento recorrendo a uma bateria.

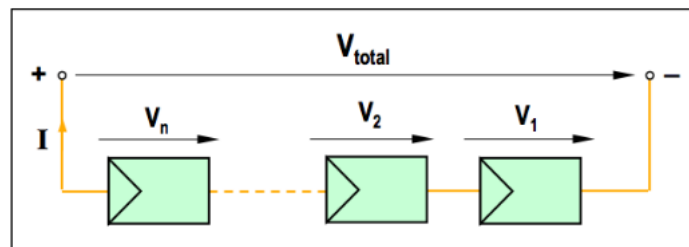


Figura 5 Módulo fotovoltaico em série [14]

Contudo, apesar de esta combinação ser a mais utilizada, a sua potência pode sofrer de quedas abruptas já que, como às células estão em série, se uma delas estiver em penumbra ou com defeito irá, comprometer a potência de saída módulo. Assim, para evitar isso é normalmente usado um sistema com um díodo de Schottky, de modo a conseguir que o módulo não fique dependente de uma ou várias células encobertas (por exemplo nuvens, folhagem das árvores) ou com defeito.

3.3. BATERIA

A energia obtida nas mais várias formas é assumido quase como um bem essencial no dia-a-dia. Com a necessidade de a armazenar a energia obtida ou excesso da mesma para uso futuro. Desta forma, foram criadas as baterias de forma a conseguir armazenar energia produzida em excesso para situações em que ocorra falha de energia. Portanto, a bateria não é mais que um

dispositivo que consegue converter as reações químicas em energia elétrica. Esta reação química denomina-se como oxidação-redução, processo que envolve transferência de elétrons de um material para outro através de um circuito elétrico.

A composição da bateria é conseguida através de várias células, cada com dois elétrodos mergulhados num eletrólito. Como os elétrodos são constituídos por materiais porosos diferentes do eletrólito, existindo reação cada um deles com o eletrólito. Caso a bateria seja recarregável a mesma realiza as reações eletroquímicas consoante a necessidade, oxidação caso esteja a ser descarregada (perda de elétrons) e redução caso esteja a ser carregada (ganho de elétrons) [15].

Existem vários tipos de baterias tornando possível uma escolha adequada consoante o tipo de utilização. Cada bateria é caracterizada pelo seu elemento químico constituinte, o que faz com que se altere as características da mesma em relação ao número de ciclos (descarga e carga), o tempo de armazenamento e limites de temperatura. Para uma melhor análise das várias características entre os diversos tipos de baterias, a Battery University [16] elaborou a Tabela 1 de comparações entre os vários tipos de baterias mais usadas.

Tabela 1 Comparação dos vários tipos de baterias existentes (adaptado de [16])

	NiCd	NiMH	Lead Acid	Li-ion	Li-ion polymer	Alkaline Recarregável
Densidade energética (Wh/Kg)	45-80	60-120	30-50	110-160	100-130	80 (inicial)
Resistência interna (mΩ)	100-200	200-300	<100	150-250	200-300	200-200
Ciclos de vida	1500	300-500	200-300	500-1000	300-500	50 ³
Tempo de Carregamento	1h	2-4 h	8-16h	2-4h	2-4h	2-3h
Auto descarga por mês	20%	30%	5%	10%	≈10%	0.3%
Voltagem da célula	1.25	1.25	2	3.6	3.6	1.5
Corrente de carga - Pico - Ideal	20C 1C	5C 0.5C ou inferior	5C 0.2C	>2C 1C ou inferior	>2C 1C ou inferior	0.5C 0.2C ou inferior

Temperatura de funcionamento (descarga apenas)	-40° a 60°C	-20° a 60°C	-20° a 60°C	-20° a 60°C	0° a 60°C	0° a 65°C
Manutenção	30-60 Dias	60-90 Dias	3-6 Meses	-----	-----	-----
Custo Típico (US \$)	\$50	\$60	\$25	\$100	\$100	\$5
Custo por ciclo (US \$)	\$0.04	\$0.12	\$0.10	\$0.14	\$0.29	\$0.10-0.50
Comercialização desde:	1950	1990	1970	1991	1999	1992

Com base na informação existente na tabela anterior, é possível ponderar a execução ou escolha do tipo de bateria consoante a necessidade do projeto em questão. Numa melhor percepção nota-se que as baterias mais recentes, como são o caso das de Li-ion, têm mais ciclos de carga do que, por exemplo, as de Lead Acid. Os ciclos de carga ou ciclos de vida representa o número de vezes que a bateria pode usufruir da capacidade total da bateria. O ciclo de carga diminui sempre que é concluído o carregamento, resultando ao longo do tempo numa diminuição da capacidade da bateria. A carga da bateria é efetuada em duas fases: uma fase rápida até os 80% e lenta dos 80% até à carga completa. Ao ultrapassar o número previsto de ciclos de carga da bateria, a mesma entra no estado de terminado ou crítico. Este estado consiste no défice de carregamento da bateria, não ultrapassando o limiar dos 80%. Com este défice, o descarregamento da bateria vai ser alterado, não indo de encontro com a curva característica de descarga [16]. Apesar de as baterias mais usadas em telemóveis, gps, etc., serem as Li-ion, não são as mais aconselháveis para sistemas de painéis fotovoltaicos por serem rapidamente danificadas com a temperatura, e sofrerem um desgaste rapidamente. Assim, por não terem essas limitações, as baterias de *Lead-Acid* têm uma grande vantagem para a utilização de armazenamento de energia solar.

3.4. COMUNICAÇÕES AD HOC

Redes Ad Hoc ou Mobile Ad Hoc Network (MANET) [17, 18] estão em expansão, visto serem redes móveis que oferecem enumeras vantagens relativamente a outras redes, como a tolerância a falhas de comunicação, instalação acessível, etc. Devido as suas vantagens, as redes Ad Hoc estão a ser cada vez mais utilizadas, resultando em melhoramentos globais. Existem três tipos de protocolos para esta rede, os pró-ativos, ou também designados por “*table-driven*”, os reativos ou também chamados por “*on-demand*” e os híbridos. Os pró-ativos mantêm a informação da rota nos diversos nós da rede, enquanto os reativos só descobrem as rotas quando é necessário, não existindo informação guardada no próprio nó com a rota para a comunicação

com outro nó. Já os híbridos são uma combinação dos dois métodos anteriores, sendo uma evolução que procura reaproveitar o melhor de cada um e uma tentativa de excluir algumas lacunas.

3.4.1 Protocolos pró ativos

O Optimized Link State Routing (OLSR; RFC 3626) [17, 19, 20] é o protocolo pró-ativo mais conhecido nas MANET's. As vantagens deste protocolo são o controlo constante do tráfego existente na rede e as rotas sempre disponíveis para uma possível comunicação ou retransmissão. Este protocolo utiliza mensagens HELLO que servem para deteção de nós vizinhos, já que todos os nós enviam, periodicamente, estas mensagens para atualizar a sua informação de rotas. Para que a rede não fique sobrecarregada, existe uma seleção de um MultiPoint Relaying (MPR) com funções de limitar o número de nós a retransmitirem pacotes, reduzir duplicações dos mesmos e permitir comunicação com a maioria dos nós. Este procedimento de seleção do MPR pode ser facilmente compreendido na seguinte ilustração Figura 6.

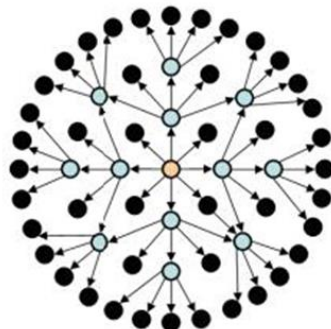


Figura 6 Exemplo de execução do protocolo OLSR [21]

Como é possível observar na Figura 6, os nós azuis são os nós MPR que os outros nós selecionaram, evitando assim enumeras mensagens a serem retransmitidas e havendo uma maior coordenação da rede. Para que esta seleção não seja aleatória, cada nó seleciona o seu MPR para que com esse nó, a informação consiga chegar ao maior número de nós possível.

Além deste protocolo existem outros nomeadamente Wireless Routing Protocol (WRP), Destination Sequence Distance Vector (DSDV) e Fisheye State Routing (FSR) [17]. Em que o conceito de armazenamento da informação da rota em cada nó mantem-se, apenas há mudanças no seu cabeçalho e mensagens protocolares.

3.4.2 Protocolos reativos

Os protocolos reativos têm um grande *overhead* em relação aos protocolos pró-ativos devido à necessidade de procura de rotas somente quando necessário. De modo a que o encapsulamento do seu cabeçalho realizado em cada iteração, até ao destino, terá toda a informação da rota percorrida, desde o emissor até ao recetor. Caso a rota seja elevada, este cabeçalho terá um tamanho superior que o mesmo, num processo idêntico, realizado por um protocolo pró-ativo. Existem vários protocolos reativos, contudo os mais utilizados são: Ad Hoc On-demand Vector Routing (AODV), Dynamic Source Routing (DSR), Associativity-Based Routing (ABR) e Lightweight Mobile Routing (LMR) [17] [14].

O protocolo AODV (RFC 3561) [22] é o protocolo mais conhecido e mais utilizado, devido à sua eficiência em relação ao protocolo DSR. Como referido anteriormente, apenas quando um nó precisa de enviar um pacote, é que executa a pesquisa da rota que o pacote deve seguir ao ser encaminhado. Para uma melhor perceção do procedimento as ilustrações Figura 7 e Figura 8 representam este processo.

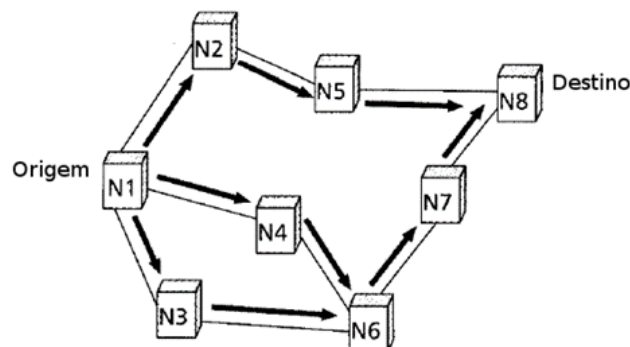


Figura 7 Disseminação da mensagem RREQ do protocolo AODV [14]

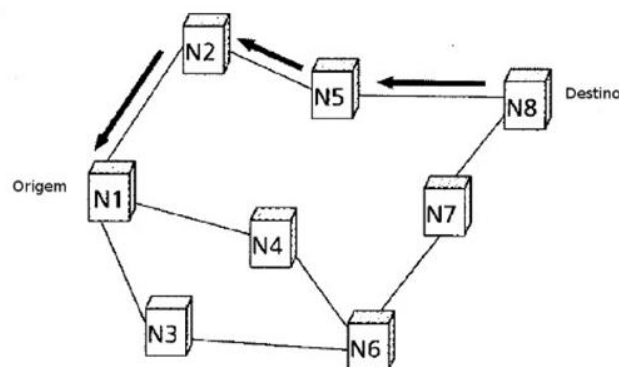


Figura 8 Transmissão da mensagem RREP do protocolo AODV [14]

Como é possível observar-se na Figura 7, o nó 1 que pretende enviar um pacote para o nó 8, solicita uma rota aos nós vizinhos até chegar ao nó destino. Este processo é denominado por *Route Request* (RREQ), e realiza-se por *broadcast* pelo nó de origem para os seus vizinhos de forma a perceber se os mesmos são ou não os destinatários do pacote. Caso não o sejam, reencaminham a informação para os seus vizinhos até encontrar o nó destino. No final deste processo existem várias rotas para o mesmo destinatário. Assim, nesta fase, o nó destinatário envia a mensagem *Routing Reply* (RREP, Figura 8). Esta mensagem é enviada para o emissor através do nó destino ou do último nó intermediário, por *unicast*. Este processo que consiste no primeiro a entregar o RREQ e reencaminhar a RREP, não só reduz o tempo de pesquisa de rota, como assume-se que o primeiro nó que entregou a mensagem RREQ seja o que possui uma rota menor.

Por fim, caso exista uma remoção de um nó é enviada uma mensagem denominada de *Route Error* (RERR) para todos os nós da rede a informar que um nó saiu da rede. Cada nó que receber a mensagem RERR remove da sua tabela de encaminhamento as rotas que contenham o nó que foi removido.

3.4.3 Protocolos com eficiência energética

Um protocolo com um parâmetro de eficiência energética tem tido uma notável relevância na investigação de protocolos de redes Ad Hoc. Com o objetivo de reduzir o consumo de energia na transmissão de pacotes entre a origem e o destino, a fim de evitar encaminhamento de pacotes através de nós com baixo consumo de energia residual, permitindo assim, a otimização das rotas de informações pela rede e evitando interferências e colisões [17, 23].

Inicialmente, a investigação sobre protocolos de encaminhamento energeticamente eficientes tinha como intenção de serem utilizados apenas em redes de sensores onde a energia limitada dos nós é uma restrição forte. Contudo, as exigências são diferentes nas redes Ad Hoc em que um só nó tem vulgarmente mais recursos de *hardware* e como tal, melhor desempenho que leva a um maior consumo. Assim deve ser necessário sustentar os recursos de cada nó na rede pois cada nó pode ser, em qualquer momento, origem ou destino de dados.

Existem várias formas de recriar ou modificar os vários protocolos de encaminhamento utilizando mecanismo de encaminhamento de informação de controlo, algoritmo com maior otimização na escolha dos nós. Este mecanismo encontra-se presente no protocolo OLSR em que a seleção do MPR pode ser variada de uma forma autónoma consoante a energia residual

em vez da cobertura existente pela vizinhança em 2 saltos. Estes critérios em simultâneo ou separados permitem aumentar a eficiência energética do protocolo OLSR [24].

Tabela 2 Comparativos entre os vários Protocolos Ad Hoc

Protocolo	Métrica	Tipo	Perdas de Transmissão	Eficiência Energética
OLSR [25]	Min. Hop	Pró-ativo	✓	✓
AODV [26]	Min. Hop	Reativo	✗	✓
DSR [27]	Min. Hop	Reativo	✗	✗
ABR [26]	Min. Hop	Reativo	✗	✗
LMR [24]	Min. Hop	Reativo	✓	✗
WRP [28]	Min. Hop	Pró-ativo	✓	✗
FSR [29]	Min. Hop	Pró-ativo	✓	✗
DSDV [25]	Min. Hop	Pró-ativo	✓	✗

Na Tabela 2 denota-se os vários protocolos estudados relativamente a métricas, tipo, perdas na transmissão (descobertas de rotas, atualização da estrutura da rede) e se possui vertente energética. Visto ser um requisito base a necessidade de um protocolo com vertente energética, apenas os protocolos OLSR e AODV refletem essa capacidade.

3.5. RESUMO

Neste capítulo são descritos os vários projetos existentes e com soluções semelhantes, que crescem exponencialmente tanto na oferta de empresas-utilizadores finais. Sendo possível averiguar uma proximidade dos projetos referidos em relação há suas metodologias, serviços oferecidos e equipamentos utilizados.

É apresentado um estudo sobre painéis fotovoltaicos que permite perceber a sua constituição, o seu funcionamento e o seu rendimento. Tendo estes fatores uma importância decisiva para a escolha do painel para este projeto. São apresentados vários materiais constituintes de bateria, sendo que, apesar de não existir inovações contínuas como visto em várias áreas, é o equipamento que é mais necessário ter em consideração devido ao seu nível de carga, ciclos de vida e a sua manutenção. Desta forma, é apresentado uma tabela de comparações entre os vários tipos de bateria, para uma melhor decisão na escolha da bateria.

Visto, ser necessário uma grande eficiência do painel para carregar a bateria que por sua vez alimenta a DCU, foi escolhido para o projeto o painel fotovoltaico constituído por células monocristalinas devido à sua grande eficiência. Com o painel fotovoltaico previamente escolhido, a solução recaiu sobre as bateria *Lead-Acid GEL* por causa da sua tensão de 12 V. Uma vez que apesar do material constituinte da bateria ser do primeiro tipo a ser implementado, garante menos manutenção em relação aos seus ciclos de vida. Esta melhoria de manutenção consiste no novo material eletrólito denominado de GEL, garantindo uma maior preservação da bateria e um alargamento do tempo de manutenção.

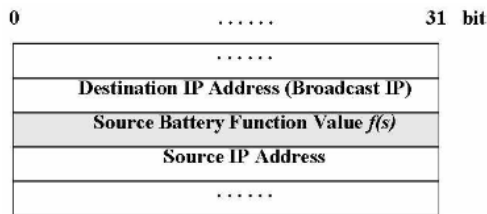
São ainda apresentados os vários protocolos de encaminhamento em redes Ad Hoc, como os pró-ativos (OLSR), reativos (AODV) e com eficiência energética. Sendo a principal função dos protocolos criar conectividade com diferentes modos de execução. São descritos os princípios de funcionamento dos protocolos pró-ativos e reativos. São expostas as características dos diferentes protocolos com recurso a uma tabela, com o intuito de apresentar e justificar a escolha de um estudo mais pormenorizado em relação à sua vertente energética. Neste caso apenas o OLSR e o AODV apresentavam esta característica. Sendo um fator de eliminação entre os diversos protocolos referidos.

4. PROTOCOLOS DE ROUTING AD HOC COM EFICIÊNCIA ENERGÉTICA

Um dos grandes problemas existentes nas redes Ad Hoc é o seu poder operacional ser limitado devido à dependência da energia existente nas baterias. Como o nó é móvel, o mesmo recorre a energia vinda de baterias que não é infinita e, uma dada altura, poderão surgir problemas da rede em que um nó, ou vários nós, não conseguem transmitir informações entre si por estarem sem energia. Assim, cada vez mais existe a intenção da redução do consumo dos vários nós numa MANET para prolongar o máximo de tempo possível o funcionamento da rede, adaptando esse funcionamento conforme as condições energéticas de cada nó na rede.

4.1. ESTRATÉGIAS PARA MINIMIZAR O CONSUMO ENERGÉTICO EM MANETS

Vários estudos têm sido realizados com o objetivo de minimizar o consumo energético nas MANETS como é o exemplo de [22, 30, 31], que utilizam uma versão do protocolo DSR que inclui o parâmetro energético para escolha de rota. Já [31] realiza uma adaptação do protocolo AODV para prolongar a vida da rede. Em relação a este fator, graças aos vários estudos é possível compreender os vários pontos fulcrais em que se pode abordar e as lacunas que se



(b)

Figura 9 Mensagem RREQ do protocolo AODV [26]. (a) Mensagem RREQ original. (b) Mensagem RREQ com função energética.

Na Figura 9 é demonstrado um novo parâmetro adicionado de *Source Battery Function Value* que representa o valor da bateria existente num determinado nó. Sendo esta informação divulgada na mensagem RREQ para os restantes nós. Já na mensagem RREP é enviado o custo da ligação para o nó vizinho, existindo uma estrutura criada localmente com os vários custo de ligação dos nós vizinhos. Há uma imensa variedade de algoritmos com eficiência energética para o protocolo AODV que substituem parâmetros nas mensagens ou apenas melhoram o seu modo de encaminhamento como o caso do Power-Aware Multipath Access Protocol (PAMAS) [24, 34]. Este é uma extensão com eficiência energética do protocolo AODV [35, 36]. Com isto, o protocolo PAMAS apenas realiza alterações no core do AODV, como um novo modelo de encaminhamento, para uma melhor utilização dos nós com mais energia residual, o mesmo desliga as ligações de rádios quando os nós não estão em uso, realizando um reaproveitamento de energia do próprio nó e aumento do tempo de vida da rede. Contudo, esta extensão não corrige totalmente o problema de existirem falhas ou atrasos de pacotes em situações de muito tráfego, mas demonstra benefícios significativos na tentativa de colmatar esta falha. Como o PAMAS é implementado no protocolo AODV, existe uma panóplia de implementações, que por sua vez pode ser utilizado em qualquer protocolo reativo.

4.3. OLSR COM EFICIÊNCIA ENERGÉTICA

Reaproveitando o protocolo OLSR, existe um parâmetro denominado de *Willingness*, traduzido para português significa disponibilidade, que é designado por um valor no intervalo de 0-7 em que a tabulação irá desde não disponível a sempre disponível, respetivamente [19, 35, 27].

Este parâmetro é incluído na mensagem *HELLO* do protocolo. Esta mensagem é ilustrada na Figura 10 em que é possível observar os vários parâmetros nela existentes, como o *Htime*

(campo em que é enviado o intervalo entre cada mensagem de envio), *link code* (especifica o estado do vizinho e o estado da ligação), entre muito outros.

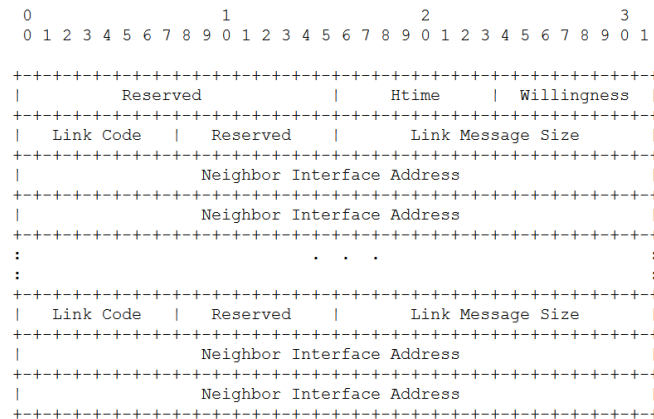


Figura 10 Mensagem HELLO do protocolo OLSR [19].

O parâmetro pode ser atribuído por omissão, com o valor 3, contudo é possível alterá-lo manualmente para um valor que se pretende. De uma forma autónoma é possível que este valor seja derivado de um processo de obtenção dependente da ligação existente com a corrente elétrica ou bateria. Caso o sistema se encontre ligado à corrente elétrica, o parâmetro é outorgado com o valor 7. Se estiver ligado a uma bateria, o mesmo solicitará um processo chamado de *Advanced Power Management* (APM) que foi desenvolvido pela Intel e pela Microsoft na década de 92, e que permitem obter as informações de energia comunicando com a *Basic Input Output System* (BIOS).

4.4. SELEÇÃO DO PROTOCOLO COM EFICIÊNCIA ENERGÉTICA

Identificadas as várias opções existentes dos protocolos selecionados, OLSR e AODV, investiga-se quais os procedimentos para executar e/ou implementar esta vertente energética. Contudo é neste aspeto que aparece uma limitação no protocolo AODV [35], visto que o mesmo não apresenta implementação real para o *Raspberry Pi* [8]. Apesar de haver inúmeras demonstrações nos mais variados simuladores de redes, tais como ns-2, ns-3 entre outros, não existe nenhuma implementação ou código fonte disponível para o *Raspberry Pi*.

Enquanto o protocolo OLSR apresenta no seu *site* oficial duas versões do protocolo, nomeadamente, OLSRDv1 e OLSRDv2. O primeiro é uma cópia exata do criador com várias versões com implementações antigas, estáveis e ainda uma versão beta. Como esta versão é a primeira obtém mais importância na procura e na resolução de erros e numa melhoria em termos

de algoritmia e performance. A segunda versão (OLSRv2) é implementada por investigadores, reescrevendo o protocolo OLSRDv1 para apenas exercer funções de rede principal, mas existindo possibilidade de incorporar várias adaptações, como é o caso da vertente energética. Visto esta versão ainda ser alfa, ainda existe muita incompatibilidade com diverso hardware, sendo este uma impossibilidade na execução do mesmo nesta dissertação. Sendo assim, a seleção do protocolo a ser explorado nesta dissertação é o OLSR (OLSRDv1).

4.5. OLSR COM *WILLINGNESS* DINÂMICO

Com a seleção do protocolo para a implementação nesta dissertação, é necessário perceber o princípio do seu funcionamento e a possibilidade da realização de melhorias. Assim, é necessário executar o protocolo e analisá-lo criticamente sobre o modo de operação do algoritmo em termos energéticos. Como é descrito em [25, 23] pelo seu criador, o *Willingness* tem vários critérios e apenas são estáticos, sendo escolhidos no início da execução do protocolo. Com esta informação é possível afirmar que os nós MPR são escolhidos diretamente consoante o seu número de vizinhos e o seu *Willingness*. Contudo como o *Willingness* é estático, por exemplo, um nó que no início tenha a bateria quase cheia e após um tempo fique quase vazia, apesar do seu estado, esse mesmo nó não deixava de ser um nó com funções de MPR. Este é problema do *Willingness* ser estático, e ser apenas um parâmetro em consideração no aparecimento do nó, e que, como se mostrará adiante nesta tese, desperdiça tempo de vida útil do nó.

Sendo assim, é proposto nesta dissertação a possibilidade de contornar este problema, e realizar uma melhoria no reaproveitamento da durabilidade do nó e na sua iteração nas comunicações. Assim, gerindo o valor do *Willingness* de forma dinâmica é criada a possibilidade de, consoante a energia restante num determinado nó, o mesmo poder ser escolhido pelos outros como MPR caso tenha a sua bateria acima de 50 % e de perder estas funções caso apareça um nó com maior percentagem de bateria. A Figura 11 ilustra um exemplo na escolha de um nó MPR consoante a energia. Supondo que o nó 2, por exemplo, tenha 50 % e o seu vizinho, nó 3, tenha 80 %. O nó 1 e o nó 4 apenas comunicam, entre si, pelos outros nós restantes como é possível verificar na imagem. Assim o nó 1 tem que escolher o nó 2 ou nó 3 como MPR para reencaminhar as suas mensagens para o nó 4. Numa fase inicial, será escolhido o nó 3 visto este ter mais bateria residual do que o nó 2. Contudo, ao longo do tempo o nó 3 vai perder energia ao passo que o nó 2 não. Assim, num determinado tempo o nó 2 terá mais bateria residual do que o nó 3.

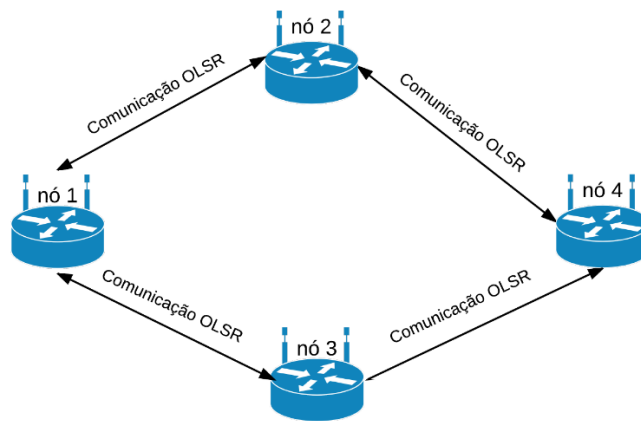


Figura 11 Rede Ad Hoc com escolha de MPR com o parametro energético.

Com o *Willingness* estático, o nó 3 vai permanecer como MPR até se desligar, sendo que nesta dissertação propõe-se uma forma de melhorar esta situação, tornando o *Willingness* dinâmico e realístico consoante a energia residual do nó, fazendo uma adaptação do seu *Willingness*. Para que na situação referenciada na imagem, em que o nó 2 depois de um intervalo de tempo permanece com mais energia que o nó 3, este seja selecionado como MPR em vez do nó 3. Desta forma, proporcionar o máximo aproveitamento a nível de envio de dados, uma vez que o nó escolhido como MPR é o nó que tem mais energia residual no intervalo de tempo. Contudo, o aumento das comunicações tem efeito na longevidade dos nós, havendo uma tentativa de retirar o máximo aproveitamento entre a energia e as comunicações.

4.6. RESUMO

Existem vários protocolos Ad Hoc que têm vindo a sofrer evoluções no que toca a melhorar a sua eficiência energética como o caso do AODV, DSR, DSDV entre muito outros [30, 36]. Contudo, neste projeto como a nível de processamento de *hardware* é utilizado o *Raspberry Pi*, ainda não existe implementação nem compatibilidade com o mesmo. De uma forma genérica, todas as melhorias consistem em conseguir prever o caminho mais curto rapidamente e com o parâmetro energético existente em cada nó. Anteriormente, foram descritos vários trabalhos de investigação e a sua realização de simulações dos vários protocolos com eficiência energética e as suas melhorias em termos de rendimento, tempo de vida dos nós e da rede [24].

A seleção dos vários protocolos foi subdividida em duas partes, numa primeira aproximação foram estudados os protocolos para perceber se incorporavam um parâmetro energético, e constatou-se que apenas dois protocolos tinham este parâmetro, eram eles o protocolo OLSR e o AODV. Visto haver apenas dois protocolos com parâmetro energético, foi realizado um

estudo sobre as várias alterações necessárias aos protocolos assim como à sua disponibilidade quanto ao código fonte. É neste aspeto que o protocolo AODV está limitado, não existindo código fonte disponível para uma implementação no seguimento do projeto descrito nesta dissertação.

Assim, o protocolo OLSR foi o escolhido para a implementação nesta dissertação e com a sua configuração e implementação definida, e com uma maturidade muito elevada [37, 32]. Nesta dissertação não se realizará uma evolução do protocolo, mas sim um reaproveitamento na sua implementação de forma a explorar as suas características de eficiência energética. Assim, será implementado num cenário de teste real o protocolo utilizando os vários recursos existentes, e resolvendo o problema do *Willingness* estático. Com o *Willingness* dinâmico pretende-se um aumento da longevidade da nossa rede e uma melhoria de *performance* das comunicações com uma escolha dos MPR consoante a energia residual dos nós.

5. IMPLEMENTAÇÃO DE REDE AD HOC COM GESTÃO ENERGÉTICA

Este capítulo descreve os componentes de uma implementação para teste de uma rede Ad Hoc energeticamente autónoma. Numa primeira fase é apresentada uma análise do consumo dos nós da rede. Depois são descritas as componentes de *hardware* e *software* incluídos no sistema. Na secção seguinte são apresentadas as diferentes fases do ensaio feito ao sistema.

5.1. ANÁLISE DE CONSUMO DA DCU

Com o avanço da tecnologia a grande parte da investigação tem recaído sobre uma vertente energética. Com esta vertente a ter a sua percentagem de encargo, é necessário caracterizar e dissecar o consumo do sistema criado.

Desta forma, existem várias hipóteses de caracterização do consumo recorrendo a ferramentas de medição como amperímetros, voltímetros ou sensores de correntes. As duas primeiras ferramentas são já muito conhecidas tanto a nível de investigação como a nível estudantil, tornando estas duas primeiras ferramentas a primeira operação de realização do consumo do sistema. Contudo, apesar de serem boas ferramentas, a maioria requer um esforço considerável

a nível físico, nomeadamente, montagem do circuito, cuidados com a corrente inserida na respetiva ferramenta, precisão do número de amostragem entre muito outros. Por outro lado, os sensores de corrente têm surgido como uma opção digital, com uma precisão muito boa nas medições e com versatilidade na medição de vários sistemas.

5.1.1 Sensor de Corrente

Neste trabalho é utilizado o sensor de corrente, visto ser uma ferramenta com características vantajosas em relação as opções anteriormente referidas, como por exemplo uma maior precisão nas medições. Desta forma, utilizou-se o sensor de corrente INA219 [38] para realizar a medição do consumo dos vários sensores incorporados na DCU. Este sensor é fornecido pela empresa Adafruit [39], especializada em componentes relativamente pequenos e no desenvolvimento de sistemas para internet das coisas. A Figura 12 ilustra o sensor que se utiliza neste projeto.

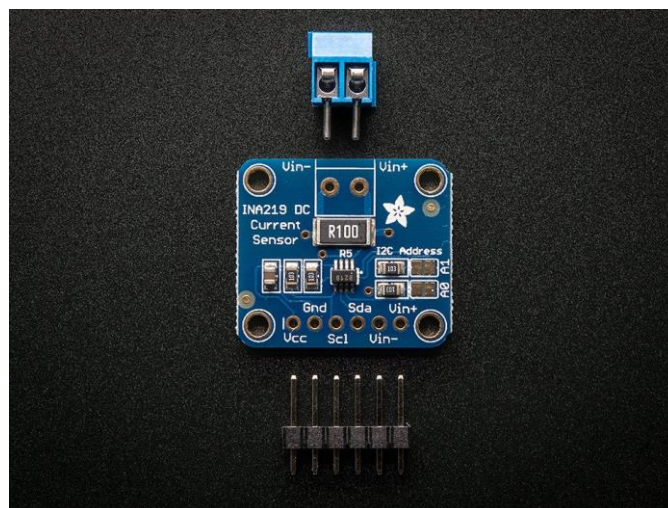


Figura 12 Sensor de Corrente INA219 [39].

Este sensor de corrente é capaz de detetar variações de corrente DC ± 3.2 A com uma resolução de ± 0.8 mA, o que permite obter um intervalo de medição grande, cerca de 400 valores. Além disso, este sensor é compatível com todos microcomputadores, por exemplo *Raspberry Pi*, *Micaz*, entre muitos outros. Conforme o manual fornecido pelo fabricante, este sensor tem uma precisão de amostragem de cerca de 0,01%, não interfere significativamente nas medições de consumo do sistema. A sua comunicação com os microcontroladores e microcomputadores é realizada pela interface I2C, usando as bibliotecas com funções pré-definidas para a realização destas comunicações e do acesso aos registos de voltagem, corrente e potência. Contudo, apesar da existência destas bibliotecas, a empresa que as desenvolveu apenas as disponibiliza para o

Arduino [39]. Apesar disso, existem dois autores que fornecem as mesmas bibliotecas para o *Raspberry Pi* e outros em python. Recorrendo a estas bibliotecas que são de livre acesso no github [7] [40] é possível efetuar experiências com o nosso sensor. Na Figura 13 é ilustrado o nosso circuito com o sensor e o nosso sistema [38].

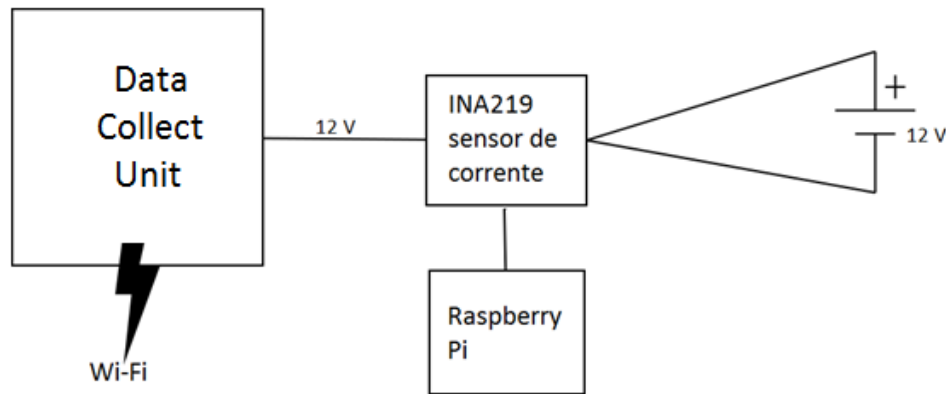


Figura 13 Arquitetura do sistema de medição de consumo.

Como é possível observar na figura anterior, é utilizado um *Raspberry Pi* ligado ao INA219 de modo a recolher os dados. Numa primeira aproximação, residia numa ligação do *Raspberry Pi*, que se encontra no interior do sistema, ao INA219. Contudo, isto interferia na obtenção dos vários consumos, que iam ser adulterados devido ao gasto energético existente para a obtenção das medições.

5.1.2 Análise do consumo energético

Com a ferramenta selecionada e descrita previamente é possível realizar as várias medições dos consumos. Para tal, é necessário escolher os vários testes predefinidos para cada opção de *software*, sendo que os testes escolhidos foram:

- Teste de *stress* para o WiFi;
- Teste de consumo de cada um dos sensores (DCU + sensor);
- Teste de consumo do sistema em modo inativo;
- Obtenção do consumo de ligação WiFi típicas.

Delineados os respetivos testes para a avaliação do consumo energético da DCU é necessário realizar a calibração do sensor de corrente, de forma inequívoca, visto para uma posterior análise dos respetivos dados [41]. É possível validar a calibração do sensor corrente recorrendo a cálculos aritméticos (teorema de Nyquist) ou com uma experiência de uma queda de tensão simples, na qual pode ser confirmada matematicamente ou com uma ferramenta analógica [41].

Efetuada a calibração do sensor de corrente são realizados os testes descritos anteriormente, com um intervalo significativo de medições, cerca de 100 amostras por segundo, para que seja possível posteriormente, analisar o comportamento do sistema em relação ao consumo existente. Em primeiro lugar, é necessário medir o consumo do sistema inativo para se obter um limiar de energia base do sistema. Com um limiar obtido, conhece-se o consumo mínimo do sistema.

Para a obtenção dos dados do sensor, recorre-se às bibliotecas já referenciadas, realizando um código para aquisição e armazenamento num ficheiro os dados em tempo real. Sendo efetuado à posteriori, as várias análises sobre a corrente, potência gasta e os seus erros relativos, como o desvio padrão e o intervalo de confiança de 95%. Assim, o Código 1 ilustra parte do código usado na obtenção dos dados do sensor e escrita dos mesmos no ficheiro.

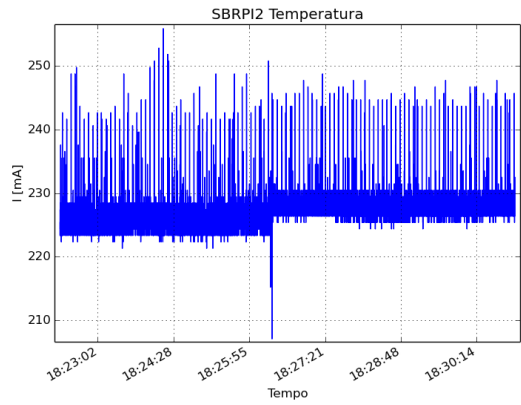
```
__author__ = 'Agostinho'
#!/usr/bin/python
import time,os,pwd,sys,datetime
from Subfact_ina219 import INA219

ina = INA219()
def timestamp():
    now = time.time()
    return now

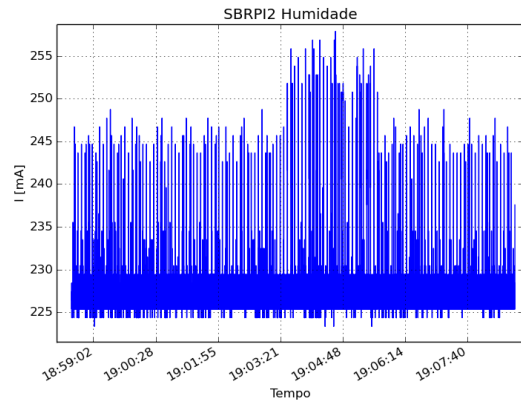
while True:
    volt = ina.getBusVoltage_V()+ina.getShuntVoltage_mV()/1000
    Watt = ina.getBusVoltage_V()*ina.getCurrent_mA()*0.001
    Arquivo =open("Sensor_Temperatura.txt","a")
    st = timestamp()
    arquivo.write( str(st)+";"+str(ina.getBusVoltage_V())+";"+
str(ina.getCurrent_mA())+";"+str(ina.getShuntVoltage_mV())+
";"+ str(volt)+ ";"+ str(watt)+"\n")
    arquivo.close()
```

Código 1 Recolha e escrita dos dados do INA219.

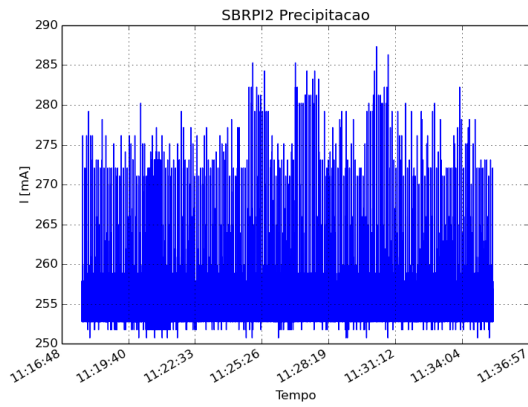
Com este código é possível realizar os vários testes com o mesmo método de obtenção do consumo do sistema. Cada medição de cada sensor é realizada com a DCU ligada em funcionamento normal e sem ligação WiFi, juntamente com o sensor pretendido para o teste. Em cada ensaio, apenas o sensor de que se pretende medir o consumo é que está ligado, os outros estão desligados. Apenas a obtenção do consumo da placa WiFi utiliza uma metodologia diferente, sendo os sensores todos desligados e apenas a placa WiFi ligada a executar o teste de *stress*. A seguinte imagem (Figura 14) ilustra os valores recolhidos dos vários sensores.



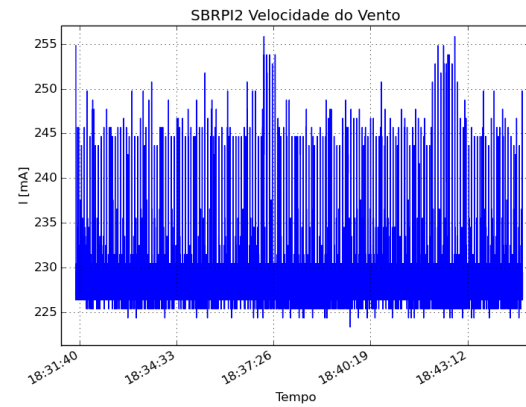
(a)



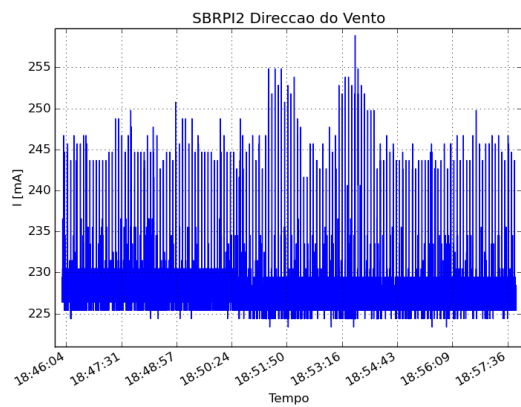
(b)



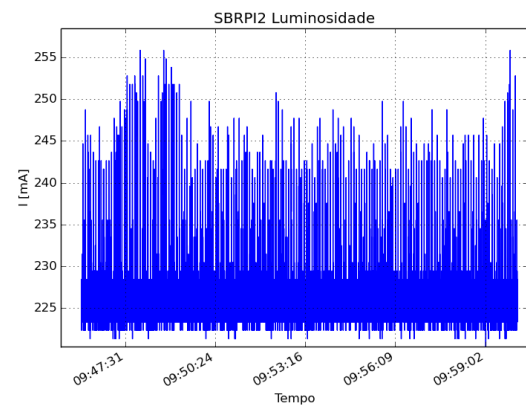
(c)



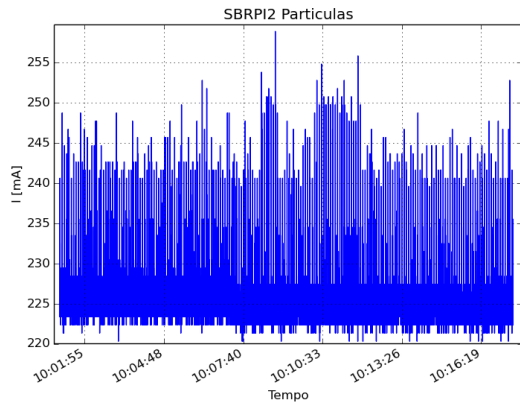
(d)



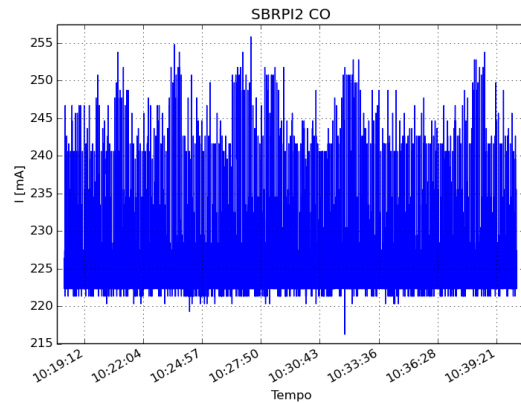
(e)



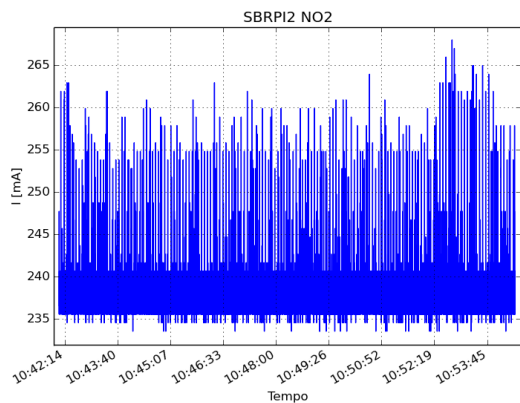
(f)



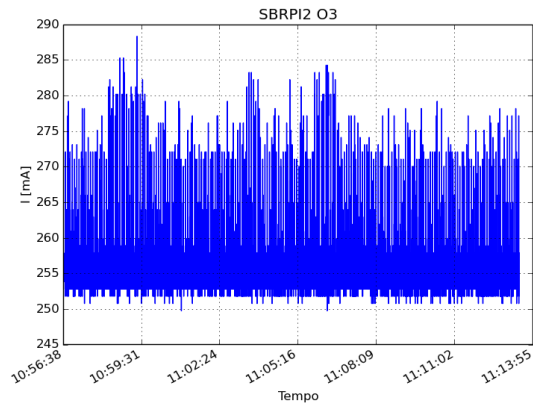
(g)



(h)



(i)



(j)

Figura 14 Consumo dos vários sensores presentes na DCU. (a) Consumo do sensor de Temperatura. (b) Consumo do sensor de Humidade. (c) Consumo do sensor de Precipitação. (d) Consumo do sensor da Velocidade do Vento. (e) Consumo do sensor de Direção de Vento. (f) Consumo do sensor de Luminosidade. (g) Consumo do sensor de Particulas. (h) Consumo do sensor de CO. (i) Consumo do sensor de NO2. (j) Consumo do sensor de O3.

As várias medições retiradas utilizando o sensor de corrente têm aproximadamente o mesmo intervalo de tempo, para obter amostras com uma grande certeza. Visto que estes sensores são de baixo preço e não existe um *datasheet* fornecido pelos vários fabricantes, tornou-se difícil perceber o seu comportamento ao longo do tempo. Assim são retiradas num intervalo de 1 segundo, aproximadamente 100 amostras. Com este número de amostras é possível obter um resultado fidedigno e respetiva validação do consumo do sensor medido. Com este número de amostras generosamente elevado é possível observar vários picos existentes em cada ilustração. Na Figura 14 (a) é possível verificar um pico em posição contrária ao normal. O mesmo foi

analisado para compreender o seu aparecimento. Efetuando análises em diferentes pontos da DCU, a anomalia do pico não se registava. Procedendo-se as trocas dos equipamentos, numa tentativa de excluir possíveis danificações ou imperfeições dos mesmos. Contudo, com estas execuções efetuadas, o pico contrário continuava a surgir. Nas restantes figuras, é possível, aproximadamente, reconhecer um intervalo de consumo médio de cada sensor. Sendo este intervalo identificado pela zona mais preenchida no gráfico.

Com o consumo dos sensores obtidos é necessário proceder à obtenção do consumo quando a DCU envia e/ou recebe dados. A Figura 15 apresenta o consumo da placa WiFi.

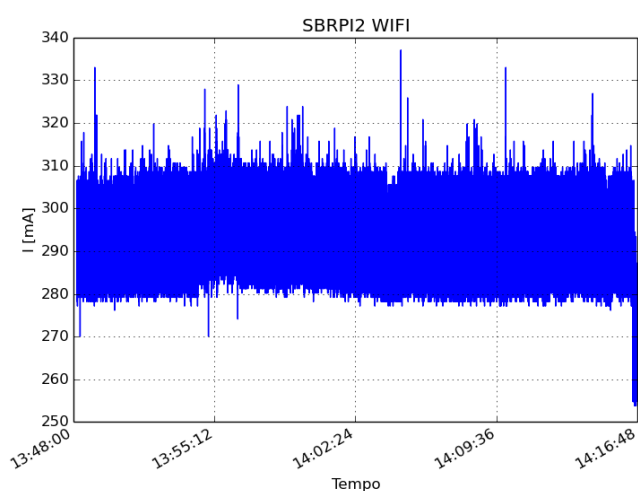


Figura 15 Consumo da placa WiFi.

Nota-se numa primeira comparação entre os gráficos dos sensores e o gráfico da placa WiFi, que esta última tem um consumo superior em relação aos sensores. Contudo, visto não ser completamente visível a média de consumo, o seu desvio padrão e o seu intervalo de confiança, a Tabela 3 apresenta as médias de consumo de cada sensor e da placa WiFi.

Tabela 3 Consumo de cada sensor presente na DCU e o seu consumo Total.

	Desvio Padrão [mA]	Média [mA]	Desvio Padrão [W]	Média [W]	Intervalo de Confiança [95%]			
					LOW [mA]	HIGH [mA]	LOW [W]	HIGH [W]
<i>Temperatura</i>	2,48	227,42	0,03	2,75	222,55	232,28	2,69	2,81
<i>Humidade</i>	2,58	227,45	0,03	2,75	222,40	232,50	2,69	2,81
<i>Precipitação</i>	2,51	256,04	0,03	3,09	251,12	260,95	3,03	3,15
<i>Velocidade vento</i>	2,22	228,30	0,03	2,76	223,95	232,66	2,71	2,81
<i>Direção vento</i>	2,34	227,60	0,03	2,75	223,01	232,18	2,70	2,81

<i>Luminosidade</i>	2,54	226,10	0,03	2,73	221,12	231,08	2,67	2,79
<i>Partículas</i>	2,43	225,46	0,03	2,73	220,70	230,22	2,67	2,78
<i>CO</i>	2,51	225,15	0,03	2,72	220,23	230,07	2,66	2,78
<i>NO2</i>	2,33	238,36	0,03	2,88	233,80	242,92	2,83	2,94
<i>O3</i>	2,47	255,62	0,03	3,09	250,77	260,46	3,03	3,15
<i>Placa WiFi</i>	8,51	295,09	0,10	3,56	278,42	311,76	3,36	3,76
<i>Sistema iddle</i>	2,27	216,37	0,03	2,59	211,93	220,82	2,64	2,75
<i>Sistema Completo</i>	7,60	328,69	0,09	3,97	313,79	343,59	3,79	4,15

Na Tabela 3 existe uma análise de consumos tanto em miliamperes como em Watt, bem como o desvio padrão e o intervalo de confiança das amostras recolhidas. Pode-se afirmar que estes dados estão corretos e devidamente validados, em coerência com as operações aritméticas. É possível observar também o consumo do sistema em *idle* e com o sistema completo em funcionamento. Neste último teste, são agrupados todos os sensores em funcionamento e a placa de WiFi. Sendo o consumo da DCU, no pior dos casos, de 4,15 W. Este resultado possibilita a criação de um sistema solar adaptado para o pior cenário.

A Figura 16 apresenta um histograma que mostra os vários consumos existentes na DCU, com as respetivas médias de consumo. Cada coluna representa o consumo apenas do respetivo sensor, sendo possível observar também o desvio padrão. No eixo do yy encontra-se a corrente que cada sensor consome e no eixo dos xx é identificado o sub-sistema que inclui os respectivos sensores, o sistema activo (com todos os sensores e WiFi ligado) e o sistema inactivo (com sensores e WiFi desligados).

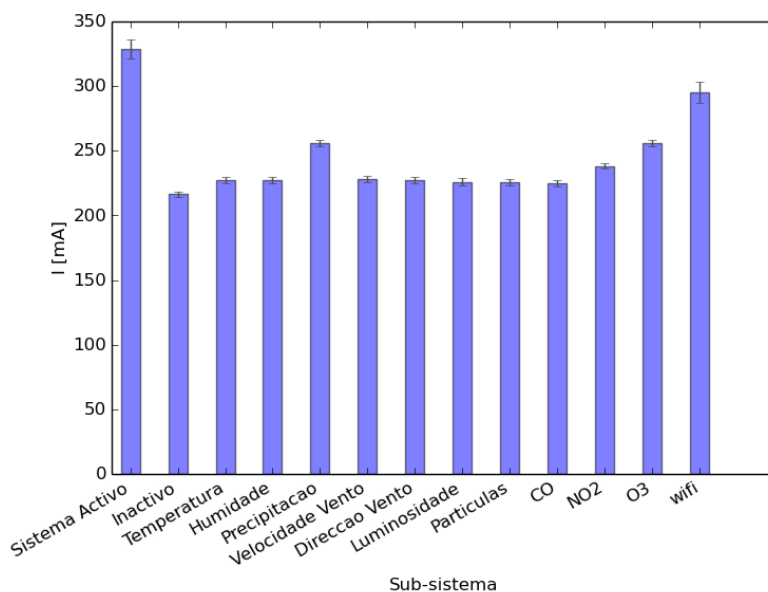


Figura 16 Consumo dos vários equipamentos existentes numa DCU.

Na ilustração (Figura 16) são exibidos os consumos dos vários sensores, respetivamente, sendo que os mesmos são inferiores ao consumo presente na placa WiFi. É possível notar também que o consumo existente no WiFi é muito próximo do consumo existente quando o sistema está completamente funcional, isto é, com todos os sensores em funcionamento e a placa WiFi a enviar e receber os dados para a base de dados.

Estes resultados têm uma importância relevante visto que estes sensores não possuem *datasheet* disponível. Descoberto o consumo do sistema em funcionamento completo, é necessário instruir uma estratégia orientada para as situações com défice de energia. A estratégia tem que ser direcionada para as comunicações, visto ser esta uma das principais componentes do consumo do sistema.

5.2. HARDWARE

Esta tese propõe usar energia solar como modo de tornar a plataforma UrbanSense energeticamente autónoma. Desta forma, foi necessário incluir no sistema novos elementos de *hardware*, tornando possível a criação da interligação entre o controlo da DCU, os respetivos sensores e os elementos do sistema solar. Na Figura 17 é ilustrada a interligação dos vários elementos do sistema.

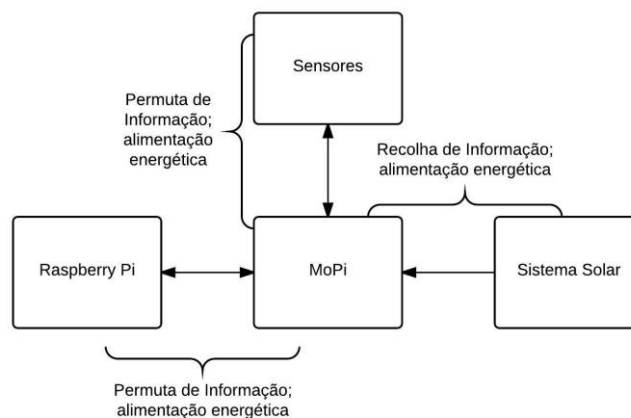


Figura 17 Esquematização da interligação dos vários equipamentos.

É visível na Figura 17, o esquema da interligação entre os vários equipamentos. O MoPi [42], tem a função de garantir de uma voltagem constante à entrada do Raspberry Pi que é um equipamento com elevada sensibilidade nas alterações de tensão. O MoPi pretende também colmatar falhas, nomeadamente, encerramentos precoces. E possibilita na mesma a permuta de

várias informações das quais, por exemplo, a aquisição dos dados dos sensores solicitado pelo *Raspberry Pi* e a energia residual num instante de tempo. Sendo esta a interligação e o acréscimo do *hardware* ao presente projeto. Adicionando o sistema solar constituído por um painel solar, uma bateria e um regulador de carga imprescindível para evitar danificar a bateria.

5.2.1 Implementação do MoPi

O MoPi [42] serve como ligação entre os sensores e a alimentação energética, de modo a ser possível recolher toda a informação do sistema solar e efetuar a permuta de informação entre os sensores e o *Raspberry Pi*. Sendo este um equipamento que permite evitar perdas de informação ou danificação dos mesmos devido ao seu regulador de comutação de 5V e microcontrolador existente. O microcontrolador implementa todas as funções relacionadas com o controlo do nível de tensão, com sinalização do mesmo através de *Light-Emitting Diodes* (LEDs) ou interfaces *General Purpose Input/Output* (GPIO) e comutação de ligado/desligado para o *Raspberry Pi*. Este pequeno circuito é adaptado a qualquer versão do *Raspberry Pi* e permite melhorar este minicomputador. O MoPi foi criado pelo Prof. Doutor Hamish Cunningham, num projeto de angariação de fundos que despertou o interesse de várias pessoas, sendo depois um projeto com relevo na universidade de Sheffield. Na Figura 18 é possível observar um esquema do MoPi.

Este pequeno circuito contem várias funcionalidades como [42]:

- Múltiplas entradas - pilhas normais, tomadas de energia, painéis solares;
- Permite um correto encerramento do *Raspberry Pi*;
- Possibilidade de monitorizar o *Raspberry Pi* 24 horas/7 dias;
- Notificações por LED, compatibilidade com Linux;
- Interface de programação desenvolvida em Python;
- Controlo remoto e ampla faixa de aceitação de voltagem.

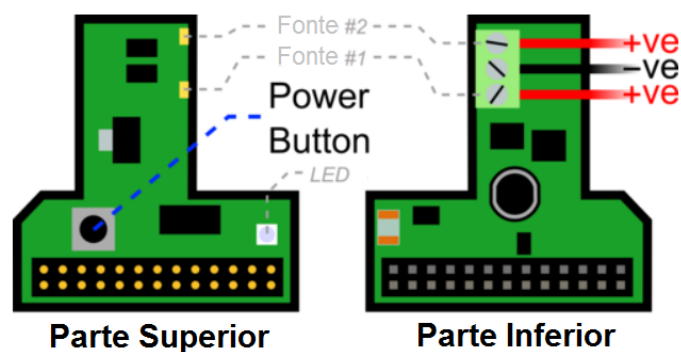


Figura 18 Ilustração da estrutura do MoPi [42].

Com este circuito diminuto é possível obter a energia atual das baterias, graças à interface de programação designada de *SimBaMon*, que consegue regular a entrada de energia fornecida ao *Raspberry Pi* e deste modo, fornecendo também a informação da bateria residual para o sustentar. Foi graças a este circuito que foi possível obter a energia residual, uma vez que o *Raspberry Pi* não possui tal funcionalidade. É também exequível, automaticamente, iniciar um encerramento “limpo” caso a energia residual existente na bateria esteja próximo do limite de descarregamento. O MoPi envia uma informação para trocar de bateria ou desligar o equipamento. Se o utilizador não executar nenhuma das opções referidas, e a bateria se encontrar no limiar crítico, é iniciado automaticamente um encerramento por escassez de energia. Esta funcionalidade é muito interessante e importante para evitar perdas de informação ou desgaste do equipamento.

A sua configuração e respetiva instalação da interface são bastante simples e documentadas, por parte do criador deste projeto. Tornando este acessório além de vantajoso, um elemento obrigatório para todos os usos seja usual, doméstico ou mesmo investigacional. Como já referido nos capítulos anteriores, as baterias utilizadas são constituídas por GEL. Na Figura 19 é possível visualizar as curvas de funcionamento das baterias *Lead-Acid* que o MoPi já inclui para a adaptação das baterias.

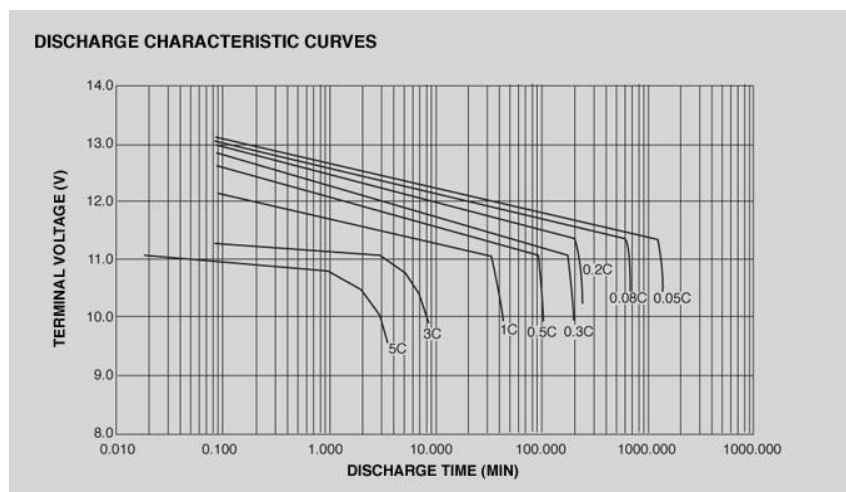


Figura 19 Curva de descarga da bateria *Lead-Acid* incluída no MoPi [15].

O MoPi já inclui, por omissão, as várias curvas do funcionamento de cada bateria. Realizando uma aproximação da voltagem mínima existente de cada constituinte, neste caso, *Lead-Acid*. Esta funcionalidade torna acessível a configuração do tipo de bateria, evitando vários problemas para a obtenção de uma amostragem fidedigna da medição sobre a tensão da bateria.

5.2.2 Instalação do sistema de energia solar

Como anteriormente investigado e referenciado os vários tipos de bateria, painéis fotovoltaicos e reguladores de carga, procedeu-se à instalação do sistema de energia solar consoante o consumo exposto. Pretende-se que este sistema não tenha falhas e que funcione continuamente. Para tal, consoante o consumo obtido anteriormente e o estudo efetuado sobre os vários recursos necessários, optou-se pelos seguintes [13]:

- Painel Solar Fotovoltaico Monocristalino [Suoyang] de 50 Wp a 12V;
- Bateria Solar [Dynoeurope] de 7Ah a 12V de GEL;
- Regulador de carga [Victron 100/15] 15^a a 12/24V.

Observando a exemplificação de uma montagem fornecida por um dos fabricantes (Figura 20), tornando-se claro a esquematização para realização do sistema. Evitando danificação dos vários equipamentos.

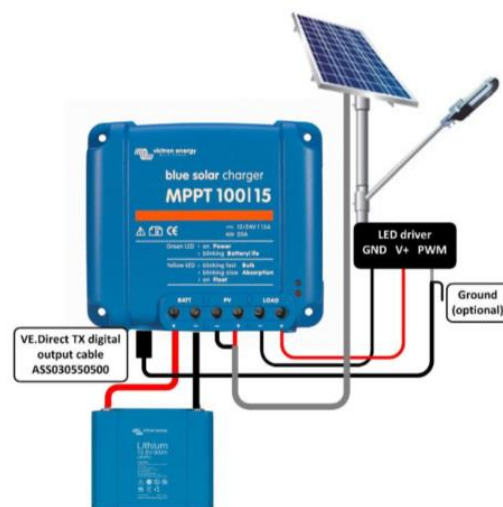


Figura 20 Esquema de montagem do sistema solar [43].

Os cabos próprios (Figura 20) fornecidos pelo fabricante na obtenção do equipamento solar, facilitou a sua montagem e evitando assim a necessidade de adquirir os mesmos. Foi necessário verificar as configurações existentes no regulador de carga, existindo *software* e manuais de ajuda fornecidos pelo fabricante do mesmo, tornando esta tarefa de configuração e adaptação dos vários parâmetros do regulador de carga muito mais inteligível.

Este regulador contém uma entrada *VE.direct* [43], que consiste em 4 pinos [3V3, TX, RX, GND] permitindo a recolha dos seguintes dados: i) corrente fornecida pelo painel solar, ii) corrente proveniente da bateria, iii) percentagem de carga da bateria e iv) funcionamento do modo de carga/descarga.

É por esta entrada de 4 pinos que também se inicia a configuração ou ajuste dos parâmetros, utilizando o cabo *VE.direct* com interface *Universal Serial Bus* (USB), efetuando a configuração do regulador. Para obtenção dos vários parâmetros de energia, recorre-se a um pequeno algoritmo realizado por vários autores, disponibilizado no repositório *Github* [43, 44, 45] com a utilização destes 4 pinos.

Com este trecho de código [43, 44, 45] é possível obter em tempo real os vários parâmetros de energia e realizar uma otimização do protocolo OLSR com eficiência energética. Se apenas fosse necessário obter os dados de uma forma visual, o fabricante disponibiliza um equipamento de monitorização através da entrada *VE.direct*. Assim, nesta dissertação é recriado o mesmo processo de recolha, usando o equipamento existente no projeto. De modo a haver uma forma direta de poupança e a criação de uma possibilidade na obtenção de dados para diversos serviços usufruírem.

5.3. SOFTWARE DE REDES

5.3.1 Instalação do protocolo OLSR

O protocolo OLSR foi implementado no *Raspberry Pi*, sendo possível executar esta implementação por duas formas (Código 2 e Código 3), recorrendo a repositórios de aplicações fidedignas onde estão versões estáveis. Contudo, nem sempre as versões estáveis são as mais recentes. É possível realizar a sua instalação recorrendo ao comando [21, 46, 37] [Código 2]:

```
Sudo apt-get install OLSRd -y
```

Código 2 Obtenção do protocolo OLSR através do repositório apt-get.

Com o comando executado como administrador, automaticamente, é efetuado a transferência dos ficheiros e a sua instalação na máquina. Porém, este processo não torna possível o acesso ao código fonte, sendo deste modo inacessíveis as alterações de forma a tornar o protocolo com eficiência energética. Com esta instalação apenas é possível configurar o protocolo com comandos básicos, sendo assim uma instalação e configuração de nível e acesso muito básico.

Uma outra forma de instalação que garante o acesso a todo o código fonte para análise e/ou alteração, é procedido com os seguintes comandos [37][Código 3]:

```
sudo apt-get update -y
sudo apt-get install build-essential -y
sudo apt-get install bison flex -y
wget http://www.OLSR.org/releases/0.5/OLSRd-0.5.5.tar.gz
tar xzvf OLSRd-0.5.5.tar.gz
cd OLSRd-0.5.5
sudo make
sudo make install
```

Código 3 Obtenção e instalação do protocolo OLSR.

Os três primeiros comandos servem para fazer a atualização das diversas bibliotecas presentes na máquina e a instalação de precedências para executar, configurar e implementar o protocolo. De seguida, inicia-se a transferência do código fonte do protocolo OLSR disponibilizado pelo criador do protocolo na página oficial. Com a transferência completa, é necessário extrair os vários ficheiros para uma pasta e realizar a sua compilação e instalação.

Com o processo finalizado é possível observar na pasta criada, a partir da sua extração, os vários ficheiros de código fonte do protocolo. Com este método de instalação do protocolo é possível aceder a ficheiros, cuja primeira opção não era possível obter.

5.3.2 Configuração do protocolo OLSR

A sua configuração é muito simples sendo apenas necessário definir a interface que se vai utilizar e que, neste caso, será uma placa de rede sem fios denominada por “wlan0”. Para tal, é necessário aceder ao ficheiro localizado por omissão na pasta “/etc/OLSRd/OLSRd.conf” e proceder a sua edição recorrendo a programas de edição, como o nano ou vim. Define-se os vários parâmetros de redes como o endereço de rede, o modo de comunicação de rede Mesh. Por omissão, o ficheiro já tem o parâmetro *Willigness* definido com o valor 3. No próximo enxerto de código é apresentado as alterações efetuadas no ficheiro OLSRd.conf [37, 21][Código 4]:

```
DebugLevel          1
IpVersion            4
LinkQualitylevel    0
MPRCoverage         1
Interface "wlan0"
```

Código 4 Alterações efetuadas no ficheiro de configuração do OLSRD.

Com a configuração do protocolo OLSR concluída é necessário executá-lo, sendo possível realizar a execução do mesmo de duas maneiras. Na primeira opção, é possível observar as várias configurações a serem executadas e as várias ações que o protocolo ativa ou processa, como encontrar nós vizinhos, ver o estado da ligação entre muitos outros. Já a outra opção de execução apenas executa o protocolo em segundo plano, não subsistindo informação sobre as tarefas executadas. Desta forma, é possível monitorizar através de outra aplicação os vários procedimentos do protocolo, que na segunda execução não ocorre, sendo apenas possível visualizar as várias atualizações existentes do protocolo com recurso a estas ferramentas. Nesta dissertação é executada a primeira opção do *daemon*, em que é necessário alterar o valor do parâmetro *debuglevel* para 1 no ficheiro de configuração. Se for necessário executar o *daemon* em segundo plano, modifica-se este parâmetro para 0. O *debuglevel* tem um intervalo de 0-9, sendo que com o aumento do parâmetro é acrescentado detalhes nos vários processos do protocolo.

5.4. ENSAIO DO SISTEMA

Numa criação do ensaio final, e devido à demora de entrega dos painéis solares, foi possível criar um cenário com uma implementação semelhante à *testbed* pretendida [47]. Recorrendo à Figura 21 é possível verificar o ensaio pretendido e o resultado esperado.

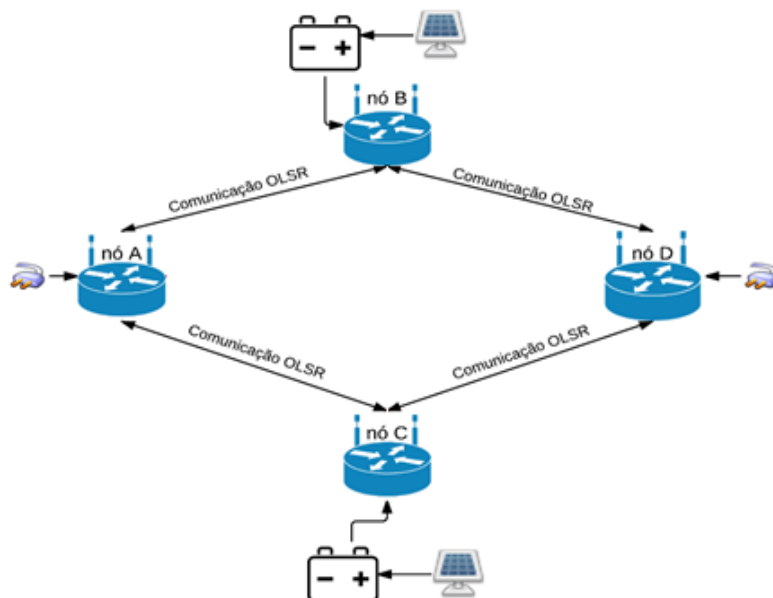


Figura 21 Esquema de ensaio do cenário da *testbed*.

Como é possível observar na Figura 21 existem dois nós (A e D) que são alimentados à corrente elétrica enquanto os nós do meio (B e C) apenas têm o sistema solar como alimentação de energia. Assim, numa primeira aproximação e considerando a limitação referida dos equipamentos, foi instruído um cenário experimental, de forma a obter o sistema funcional. De modo, a perceber as lacunas existentes aquando a chegada do sistema solar completo. Assim, realizaram-se testes em que os nós do meio apenas têm uma bateria *Lead-Acid* de 12V a 3.2 Ah. Estas baterias já existiam no Instituto de Telecomunicações [1] tendo sido as utilizadas para realizar pré-testes ao ensaio mencionado. Este pré-testes consistem em verificar a execução do protocolo e perceber possíveis erros nos *scripts* criados. Assim, compreender e colmatar possíveis lacunas, erros ou limitações do sistema. Desta forma, a Figura 22 ilustra este pré-teste para revelar a reação do ensaio e detetar possíveis falhas.

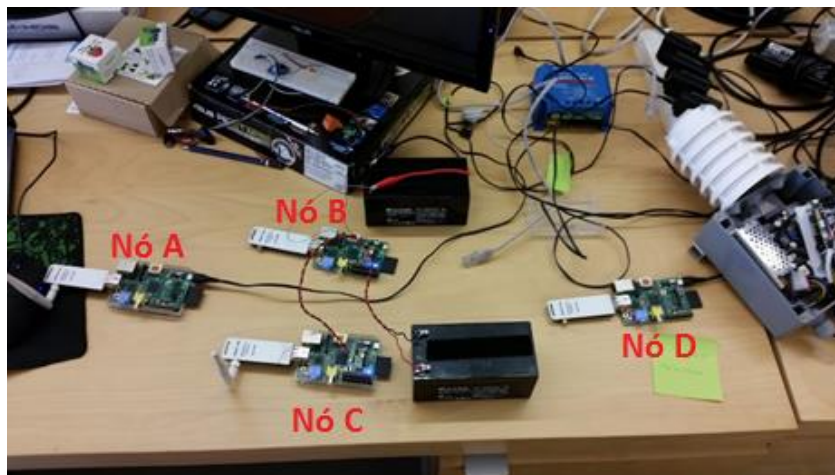


Figura 22 Ilustração do pré-teste em laboratório.

Assim, na ilustração Figura 22 não existem comunicações diretas entre o nó B e nó C, nem entre o nó A e nó D. Para o nó A realizar uma comunicação com o nó D terá que selecionar um entre os nós do meio (B e C), sendo um destes o seu MPR no encaminhamento da sua mensagem. Com este teste de ensaio, foi possível observar a escolha do MPR que o nó A efetua para realizar a comunicação com o nó D. Esta escolha foi feita através do parâmetro *Willingness*, que traduz a recetividade de um nó servir como MPR, consoante a sua disponibilidade energética nesse momento. Para que o parâmetro do *Willingness* esteja em concordância com a energia atual do nó, foi desenvolvido um script para a componente do MoPi, uma vez que ainda não existia o equipamento do sistema solar. Sendo possível obter a energia em voltagem da bateria, do equipamento referido. O próximo excerto de código

(Código 5) apresenta a obtenção da voltagem da bateria recorrendo ao MoPi, e realizando uma adaptação dessa voltagem para o parâmetro de *Willingness*.

```
__author__ = 'Agostinho'
import subprocess as sp

cmd= "sudo mopicli -v1"
p = sp.Popen(cmd,shell=True, stderr=sp.PIPE,
stdout=sp.PIPE)
pid=p.pid
stdout_value,stderr_value= p.communicate()
xlp=p.poll()
failed=p.returncode
a,b,c,bateriaV_real = stdout_value.split(" ")
print "Bateria voltage: %s" % bateriaV_real
teste= 0

if int(bateriaV_real) >= 12500:
    teste = 6
if int(bateriaV_real) >= 12000 and
int(bateriaV_real) < 12500:
    teste = 5
if int(bateriaV_real) >= 11500 and
int(bateriaV_real) < 12000:
    teste = 4
if int(bateriaV_real) >= 11000 and
int(bateriaV_real) < 11500:
    teste = 3
if int(bateriaV_real) >= 10500 and
int(bateriaV_real) < 11000:
    teste = 2
if int(bateriaV_real) >= 9500 and
int(bateriaV_real) < 10500:
    teste = 1

bateria=teste
```

Código 5 Excerto para realizar a adaptação energética no protocolo OLSR.

Como é possível observar no excerto acima (Código 5), existem intervalos delimitados de forma a tornar possível a adaptação da voltagem relativa ao parâmetro *Willingness*. Estes valores de transição são delimitados consoante a curva de descarregamento da bateria *Lead-Acid*, existindo assim uma tentativa de aproveitar ao máximo a disponibilidade energética da mesma. Numa primeira fase, obtém-se a voltagem da bateria através do MoPI efetuando uma manipulação dos argumentos para a variável denominada real, de modo a recolher o valor da voltagem. Realizando uma conversão do tipo de dados, de maneira a realizar uma comparação com os

vários parâmetros. Com a execução deste excerto de código obtém-se o parâmetro de *Willingness* para posteriormente se atualizar no protocolo OLSR [48, 2, 49].

Aquando da atualização do *Willingness* no protocolo, foi descoberta uma limitação da implementação usada. Nesta implementação, depois de se iniciar o *daemon* do protocolo, não é possível realizar a atualização do *Willingness* apesar de já integrar código para isso. Com esta limitação e, depois de vários testes para perceber se era possível adaptar o algoritmo, com a confirmação do autor a afirmar que o mesmo código já não era tão fiável, e por isso não executava essa função do protocolo como está especificado, e era pretendido. A solução encontrada, e também proposta pelo autor da implementação do OLSR, é ao iniciar o *daemon*, este ler o ficheiro de configuração, já anteriormente referido, para realizar uma atualização do parâmetro *Willingness*. Com este processo, rescreve-se o mesmo parâmetro atualizado no ficheiro e reinicia-se o *daemon*. Ao realizar esta operação para a atualização do *daemon* existe uma perda aproximada de 10 milissegundos.

Usufruindo do excerto de código já apresentado acima, é acrescentada a funcionalidade de verificar qual o valor do parâmetro do *Willingness* no ficheiro de configuração, e caso seja diferente do atual procede-se à alteração do mesmo e realiza-se o processo de reiniciar o *daemon*. O código total encontra-se no Anexo A.

5.5. RESUMO

Neste capítulo são descritas a arquitetura e implementação para a realização deste projeto. Nesta dissertação é proposto um modelo adaptativo das comunicações consoante a sua energia existente. Este capítulo é dividido em seis subcapítulos, no primeiro subcapítulo (dividido em duas secções) é apresentado o consumo energético da DCU. Em que é identificado a ferramenta selecionada, nomeadamente, INA219. Efetuando os vários testes de obtenção do consumo energéticos dos vários sensores, placa WiFi, e o sistema em *iddle* e completo. O segundo subcapítulo é descrito um módulo denominado de MoPi que serve, essencialmente, como microcontrolador e um regulador de carga. Este permite realizar várias funções como encerrar corretamente, criar uma possibilidade de ligar baterias e a respetiva monitorização da carga e do próprio sistema. Sendo um módulo com uma importância nesta dissertação. No terceiro subcapítulo é apresentado a implementação do sistema solar que são denotados os vários constituintes do sistema e a sua respetiva montagem. No quarto e no quinto subcapítulo é apresentado a instalação e configuração do protocolo de comunicação, nomeadamente, o

OLSR. Sendo demonstrado os vários passos e possibilidades de instalar e executar o *daemon*. No último subcapítulo é apresentado a arquitetura de montagem, e o *software* que será utilizado no sistema. Com isto, existiu a possibilidade de perceber os erros relativamente à montagem, e ajuste do código para obtenção da bateria e da manipulação em tempo real.

6. ANÁLISE E DISCUSSÃO DOS RESULTADOS

Neste capítulo é apresentada a avaliação do desempenho do modelo adaptativo proposto nesta dissertação, em relação às comunicações com eficiência energética. Os resultados foram obtidos em ambiente real e permitem analisar o desempenho do sistema no tempo de adaptação do protocolo OLSR, o número de pacotes existentes na rede, e quantificação dos pacotes recebidos ou perdidos consoante a adaptação energética da rede.

Para se avaliar o consumo de um nó com funções de MPR, foram efetuados três testes. Num cenário (1) o nó tinha como parâmetro de *Willingness* = 7, noutra cenário (2) tinha o parâmetro *Willingness* = 3 e por fim (3) com o parâmetro *Willingness* = 1. Ao realizar estes três cenários, é possível ter um cenário padrão de consumo quando o nó está em funções completas (*Willingness*=7), por omissão (*Willingness*=3) ou sem funções. Assim, obtém-se o número de horas que o nó MPR, dependendo do valor deste parâmetro, consegue realizar comunicações.

Os vários testes tiveram sempre o mesmo esquema de rede e o mesmo material apenas variando no meio ambiente, visto não ser possível controlar esta variável. Contudo, existiu sempre a tentativa de obter uma maior aproximação do mesmo entre os vários dias.

Foram efetuados vários testes de obtenção do tempo de adaptação do protocolo OLSR à energia existente, onde é indicado o tempo de adaptação de cada teste efetuado, a média e o intervalo de confiança de 95 %. É apresentado também o número de pacotes existentes na realização do comando *ping*, num espaço temporal de 300 segundos e recorrendo ao programa “*tshark*” [50]. Foram realizadas 5 vezes este teste, de forma a conseguir obter um desvio padrão e um intervalo de confiança a 95 % considerável.

6.1. EXPERIMENTAÇÃO DO SISTEMA COMPLETO

Com os recursos detalhados nos capítulos anteriores, é possível integrar todo o material selecionado e realizar a sua montagem, para uma experimentação do protocolo Ad Hoc com eficiência energética. A montagem realizou-se na cobertura do edifício I da Faculdade de Engenharia da Universidade do Porto [1]. A Figura 23 ilustra o sistema completo e a fase de experimentação dos vários testes.



Figura 23 Ilustração da testbed em fase experimental.

Na Figura 23 é possível observar os sistemas de experimentação a realizar os testes previamente identificados. Houve um problema que demorou um determinado período de tempo a perceber. Este problema de propagação consistia no facto de os dois nós da rede estarem ligados à corrente elétrica, pelo que era necessário colocar dentro de uma sala existente na cobertura. Contudo, a estrutura desta sala absorvia grande parte das ondas WiFi, não possibilitando o reconhecimento dos nós vizinhos. Esta limitação foi rapidamente

contornada, adicionando um antena WiFi com maior ganho em todos os nós da rede. A montagem seguiu o esquema descrito considerando já o sistema de painéis solares.

Numa primeira verificação do funcionamento do protocolo OLSR na escolha do nó para MPR é executado o comando *traceroute*. O mesmo permite visualizar o percurso que os pacotes efetuam até chegar ao destino [33]. Na Figura 25 é apresentada a rota que os pacotes do nó A realizam para comunicar com o nó D. Para tornar perceptível é apresentado novamente a esquematização do sistema (Figura 24 Esquematização da testbed.).

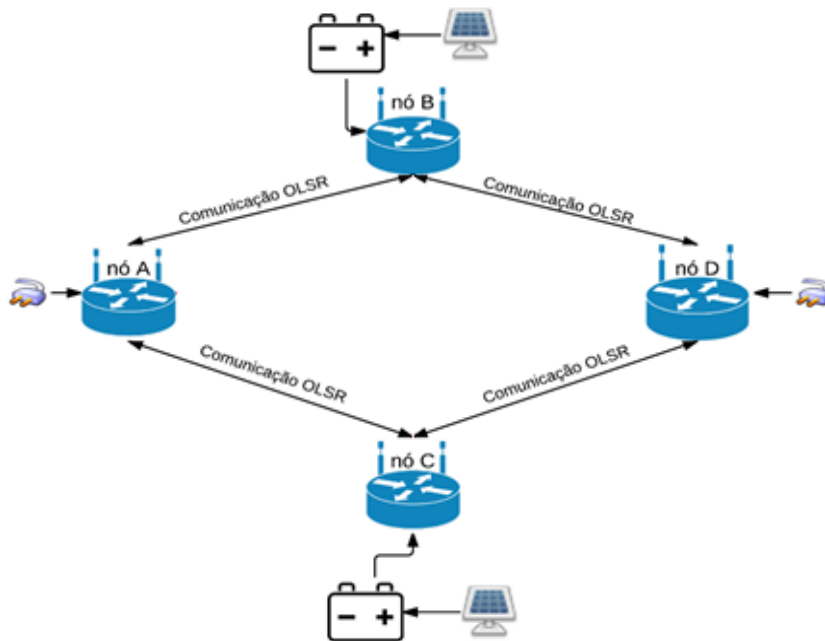


Figura 24 Esquematização da testbed.

```
pi@raspberrypi ~ $ traceroute 172.30.29.233
traceroute to 172.30.29.233 (172.30.29.233), 30 hops max, 60 byte packets
 1  wifi-interno          (172.30.6.144)  223.106 ms  224.631 ms  225.464 ms
 2  wifi-interno          (172.30.29.233) 232.533 ms  233.361 ms  234.617 ms
pi@raspberrypi ~ $
```

Figura 25 Validação do funcionamento de rotas do protocolo OLSR.

Como é possível observar na Figura 25, o percurso dos pacotes são direcionados para um nó da rede, neste caso o nó C. No momento em que se realizou este teste, era o nó C que tinha um nível superior de energia em relação ao nó B. Numa primeira aproximação, é possível verificar que o protocolo adaptado nesta dissertação encontra-se em correto funcionamento.

6.2. ANÁLISE DOS RESULTADOS

Para validar este protocolo no ambiente real, efetuaram-se vários testes de modo a verificar o número de pacotes recebidos com sucesso e o número de pacotes enviados num espaço de temporal controlado, de modo a perceber se a adaptação energética melhora ou deteriora o protocolo e a sua transmissão de dados.

6.2.1 Tráfego de pacotes recebidos e perdidos no OLSR com eficiência energética

Para a captura dos pacotes e análise dos mesmos utilizou-se o programa denominado “*tshark*” [50], com um tempo de captura de 300 segundos. Na Figura 26, é apresentado o histograma com os vários dados sobre o número de pacotes existentes nos vários nós.

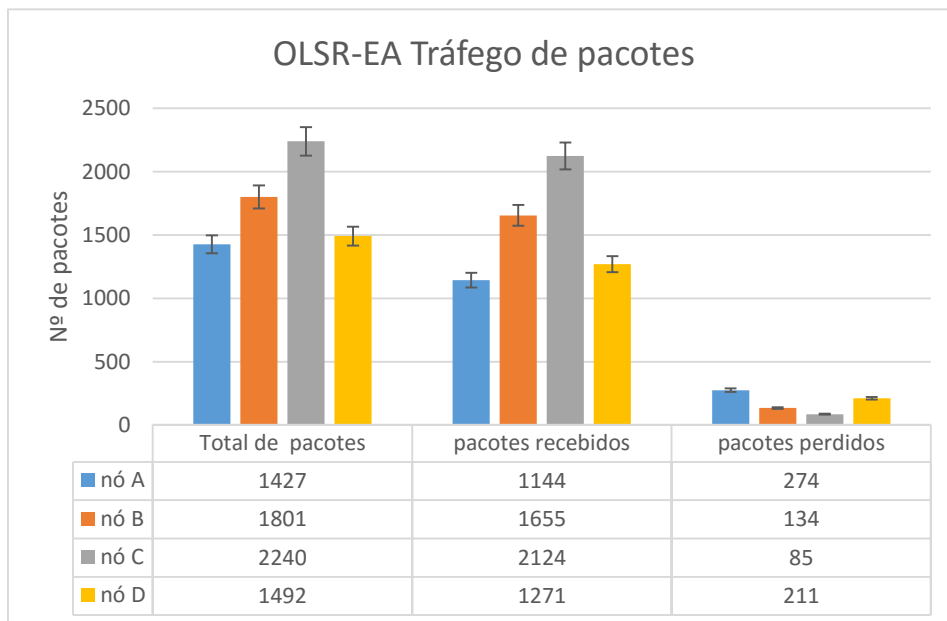


Figura 26 Tráfego de pacotes dos vários nós em teste.

É visível pela Figura 26 a existência de perdas nos vários nós, contudo esta perda não tem um peso considerável no resultado final. Assim, denota-se que este protocolo com eficiência energética em relação ao número de pacotes perdidos encontra-se dentro do plausível, entre os 5 % e 19 % [23, 25, 30]. Esta perda de pacotes pode ser justificada pelo tempo excedido do comando *ping* e/ou por sobrecarga de encaminhamento de pacotes *acknowledgment*. É possível observar que os nós B e C, intermediários dos restantes nós, realizam a progressão do tráfego dos outros nós consoante a sua energia. Havendo uma expressão quase nula nos pacotes perdidos nesses nós. Numa visão geral, existe continuamente na comunicação entre o emissor e o destinatário. Uma vez que entre o número de pacotes que são enviados e

recebidos, apenas existe uma diferença mínima de aproximadamente 12% [23, 25, 30], o que se pode concluir num correto funcionamento protocolo.

6.2.2 Tempo de adaptação do OLSR com eficiência energética

A realização deste teste tem uma relevância elevada, visto que sempre que numa rede de comunicação existe uma adaptação ou falha, o tempo de recuperação da mesma é importante. Desta forma, é necessário verificar se o protocolo usado nesta dissertação requer muito tempo para a adaptação à energia remanescente nas baterias. Foram efetuadas 20 medições, de forma a obter um intervalo de confiança e um desvio padrão com credibilidade. Neste teste são registados dois instantes de tempo, o primeiro regista o tempo de iniciação da mudança e o restante regista o momento em que o nó está novamente ativo na rede e a enviar dados. Realizando a diferença entre os dois tempos é obtido o tempo de adaptação do protocolo. A Figura 27 apresenta os tempos de adaptação medidos no protocolo OLSR com eficiência energética.

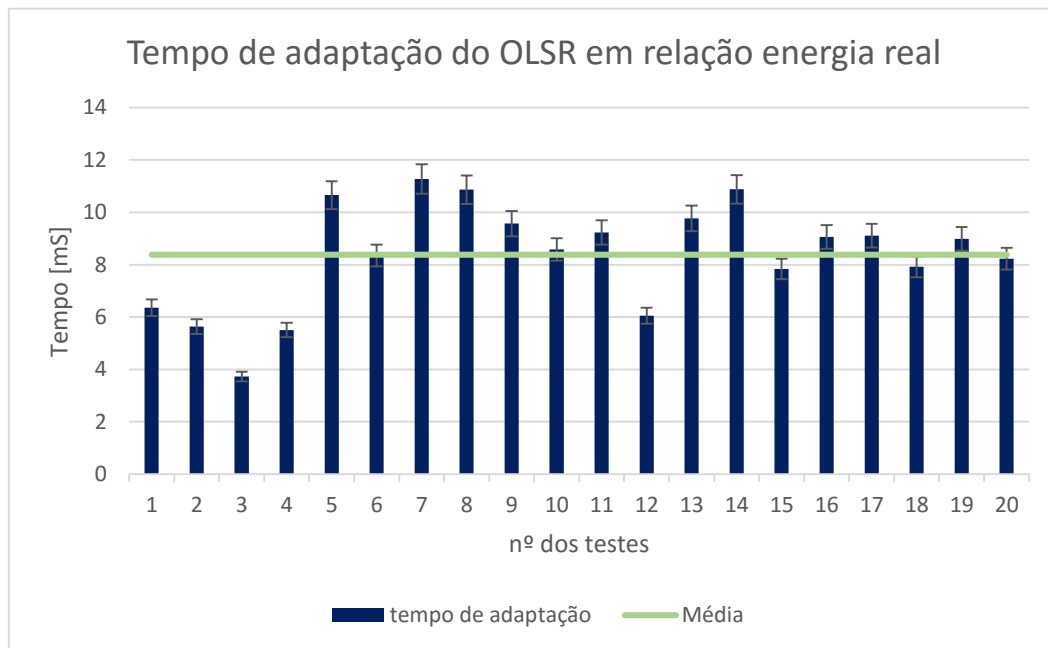


Figura 27 Tempo de adaptação do protocolo em relação à vertente energética.

É possível observar que nos primeiros testes existiu uma dispersão em relação aos restantes testes. Contudo, pode-se assumir que esta adaptação poderá dever-se ao Raspberry Pi ter poucos processos em execução em segundo plano. A outra hipótese poderá ser através da receção das mensagens periódicas, nomeadamente HELLO, muito perto do envio da mesma para os outros nós.

Pode-se afirmar também que a média do tempo de adaptação é de 8.37 milissegundos, o que releva ser muito insignificante no contexto da transferência da informação recolhida pelos sensores, e até mesmo, numa transmissão de ficheiros. Este teste confirma uma grande vantagem do presente protocolo, visto que o tempo de adaptação em relação à energia real era um ponto crucial para a avaliação do mesmo e um objetivo de melhoria do protocolo.

6.2.3 Eficiência do OLSR com eficiência energética

Concluído com grande apreciação os resultados e a sua análise dos testes obtidos sobre o protocolo, é necessário verificar a sua eficiência em relação ao protocolo Ad Hoc no que se refere à duração da bateria. Assim, procedeu-se à realização de dois testes de modo a obter um limite inferior e um superior para a longevidade operacional das comunicações dos nós. Neste caso, a duração da bateria quando o nó está com o parâmetro de *Willingness* no máximo (isto é, com o valor 7) e quando está no mínimo (isto é, com o valor 1). Desta forma, ao obterem-se estes dois resultados é possível verificar a duração mínima e máxima do nó consoante o seu *Willingness*. A Figura 28 e a Figura 29 apresentam os valores da carga da bateria do nó configurado, respetivamente, com o parâmetro de *Willingness* 7 e 1. É de referir que estes parâmetros de *Willingness* não foram alterados durante cada um dos testes.

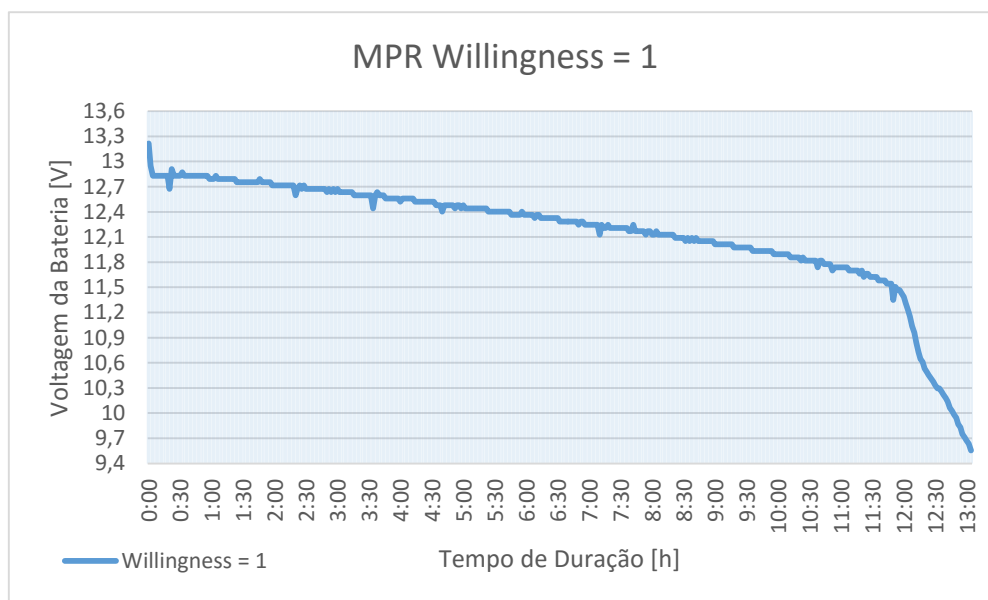


Figura 28 Nó com o parâmetro de *Willingness*=1.

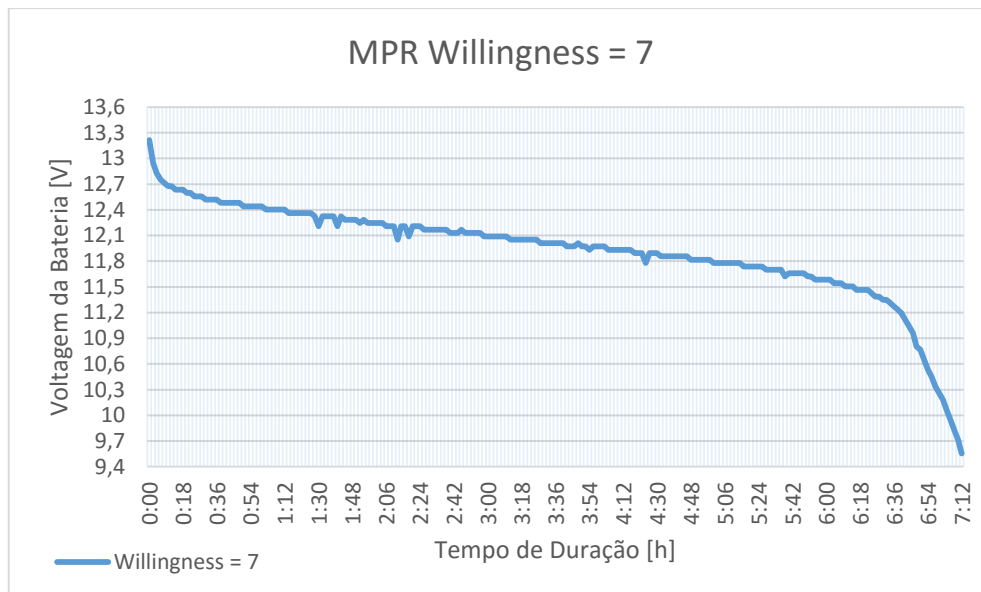


Figura 29 Nó com o parâmetro de *Willingness*=7.

Na Figura 29, o nó MPR com o parâmetro 7 é o que obtém menos duração de vida útil. Estes resultados vão de encontro com o esperado na teoria, visto que o parâmetro 7, o mesmo se encontra num estado permanentemente de MPR, sendo um nó principal e constantemente ativo nas comunicações entre os vários nós. Já o mesmo não acontece quando o parâmetro é 1, visto que este não participa nas comunicações da rede e apenas realiza o mínimo possível de transmissões com o seu MPR, aumentando o seu tempo de vida como é possível observar na Figura 28. Sabendo o número de horas mínimo e o número de horas máximo que o sistema suporta, são realizados os outros testes quando o OLSR está com o parâmetro *Willingness* = 3 estático e também com a adaptação proposta do protocolo presente nesta dissertação.

No teste da adaptação do protocolo é pretendido teoricamente que consiga uma melhor relação entre a comunicação e o seu tempo de vida útil. Com isto, espera-se que tenha um tempo de duração superior ao nó com o parâmetro 7 e mais perto da longevidade do parâmetro 1. Já quando o parâmetro encontra-se estático, é esperado um valor médio entre o parâmetro 7 e 1. A Figura 31 apresenta o tempo de vida útil do nó consoante o parâmetro estático de *Willingness*=3.

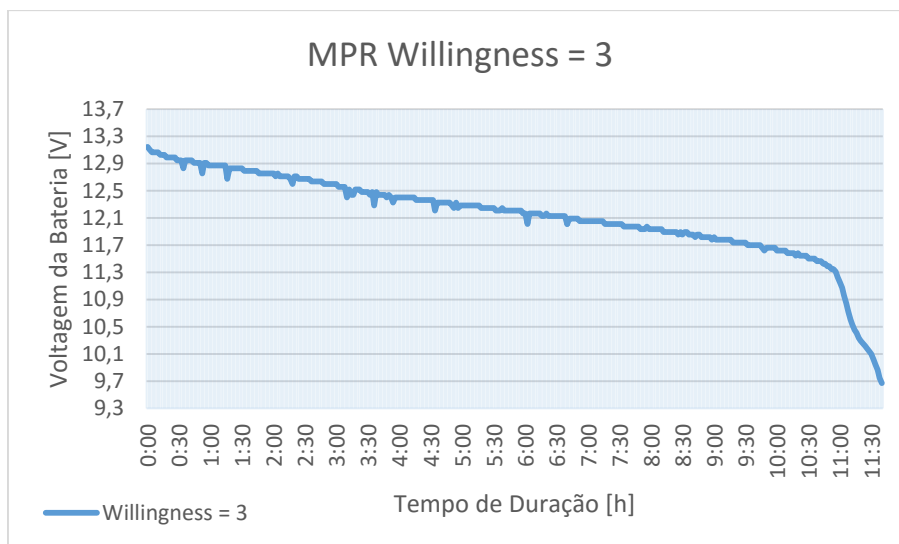


Figura 30 Nó com o parâmetro estático de *Willingness=3*.

É possível visualizar que com este parâmetro definido por omissão, o sistema consegue aguentar cerca de 11 horas e 40 minutos. Sendo este valor relativamente próximo do tempo de duração quando o parâmetro de *Willingness=1*. A Figura 31 apresenta o tempo de vida útil do nó quando o parâmetro de *Willingness* é adaptado dinamicamente.

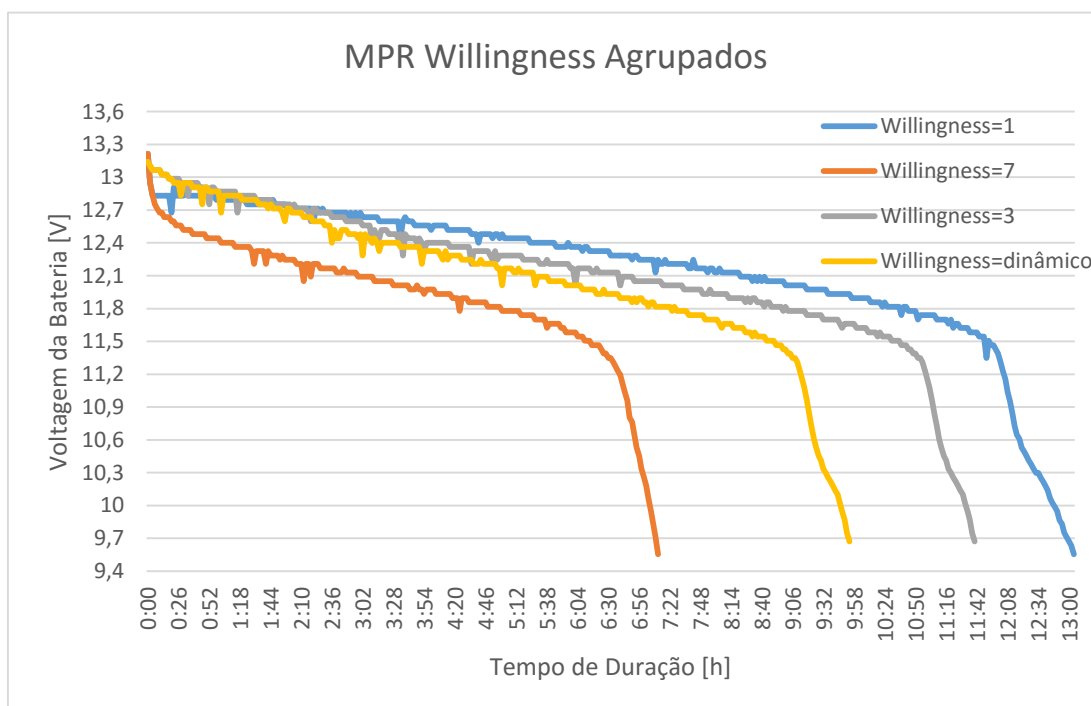


Figura 31 Comparação dos diferentes parâmetros de *Willingness*.

Na Figura 31, é ilustrado o protocolo descrito na presente dissertação (linha amarela) tendo um comportamento semelhante aos gráficos anteriores, havendo uma maior duração do

tempo de vida útil do nó a realizar comunicações, relativamente ao que permanece com o parâmetro de *Willingness* máximo e uma aproximação relativa ao tempo de duração quando este se encontra com o *Willingness* no mínimo. Contudo, existe uma diferença relativamente à experiência realizada com o parâmetro de *Willingness*=3, visto este ter menos tempo de duração comparando com o parâmetro estático.

6.2.4 Taxa de transferência do OLSR com eficiência energética

Desta forma, realiza-se uma análise à taxa de transferência entre o parâmetro estático (com o valor de 3) e a adaptação proposta (isto é, dinâmico), sendo enviado um ficheiro de um nó para o outro e realizada uma média da velocidade oferecida e o tempo de envio deste ficheiro. É selecionado um ficheiro com tamanho de 6460 KB cujo tamanho é aproximado ao de um ficheiro de áudio ou imagem com qualidade média. Para o envio deste ficheiro é utilizado o comando *Secure Copy (SCP)* do Linux. Sendo este comando já reconhecido, estável e com informações relativas à velocidade média de envio de um ficheiro e o seu tempo de duração. Deste modo, é repetido 15 vezes o processo de envio utilizando os parâmetros de *Willingness* dinâmico e estático. Na Figura 32 são apresentados os resultados das várias repetições das transferências e efetuada a respetiva análise.

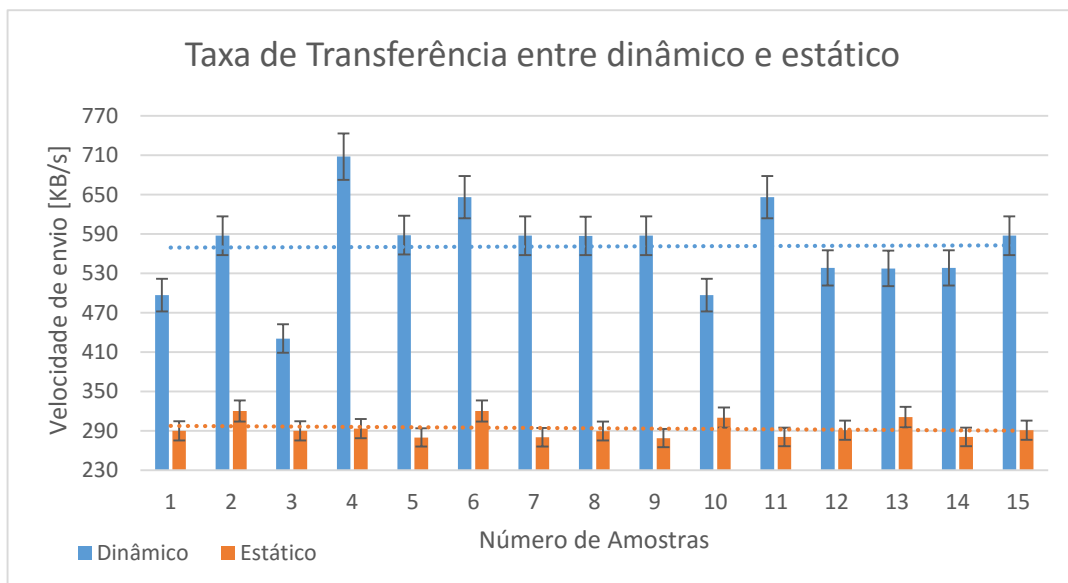


Figura 32 Taxa de transferência entre parâmetro dinâmico e estático

É possível averiguar na Figura 32 que o *Willingness* dinâmico consegue oferecer uma velocidade de transferência superior à conseguida usando o *Willingness* estático, sendo a velocidade média de envio de 570 KB/s e o tempo de envio de um ficheiro de

aproximadamente 6,6 MB, cerca de 11 segundos. Já o mesmo ficheiro demora o dobro do tempo (22 segundos), quando o parâmetro de *Willingness* é estático, dado que a velocidade de envio é inferior, cerca de 294 KB/s. Os valores apresentados são fiáveis, visto que o desvio padrão e o intervalo de confiança a 95% encontrarem-se num intervalo relativamente pequeno. Para uma interpretação mais detalhada dos valores da figura é apresentada na Tabela 4, a taxa de transferência entre o parâmetro dinâmico e estático, e os respetivos tempos de envio em cada teste.

Tabela 4 Taxa de transferência com o parâmetro dinâmico e estático e o seu calculo aritmético.

Willingness Dinâmico		Willingness Estático=3		Willingness Dinâmico		Willingness Estático=3	
KB/s	Segundos	KB/s	Segundos	Média			
KB/s	Segundos	KB/s	Segundos	KB/s	Segundos	KB/s	Segundos
496,9	00:13	290,1	00:23	570,86	00:11	293,89	00:22
587,3	00:11	320,2	00:20	Desvio Padrão			
430,7	00:15	290	00:23	68,40	00:01	14,55	00:01
707,8	00:09	293,6	00:22	Intervalo de Confiança (95%)			
588,2	00:11	280	00:24	Mínimo	Máximo	Mínimo	Máximo
646	00:10	320,3	00:20	532,98	608,74	285,84	301,95
587,3	00:11	280,4	00:24				
587,1	00:11	289,8	00:23				
587,3	00:11	278,9	00:25				
496,9	00:13	310,2	00:21				
646	00:10	280,8	00:24				
538,3	00:12	291,2	00:23				
537,5	00:12	311,2	00:21				
538,3	00:12	280,7	00:24				
587,3	00:11	291	00:23				

Este teste de processamento de envio de um ficheiro comprova o porquê de o tempo de duração ser inferior quando o parâmetro é dinâmico, em relação à longevidade do parâmetro estático. Quando o parâmetro é dinâmico, existe uma oferta superior na taxa de transferência em relação ao parâmetro estático. Como visto anteriormente, na análise de consumo, a placa de WiFi é o elemento que consome mais energia. Sendo deste modo possível validar e confirmar este efeito uma vez mais.

Apesar do tempo de duração ser inferior se o parâmetro for dinâmico, este consegue oferecer uma maior panóplia de serviços, uma vez que possui uma taxa de transferência alta. Podendo este ser utilizado em serviços de tempo real ou no envio de dados em situações oportunas. É possível afirmar que existe uma ponderação entre a longevidade e o envio de dados, oferecendo velocidades de envio durante um período maior.

6.3. RESUMO

Neste capítulo são descritos os vários testes realizados de forma a testar o protocolo OLSR e o protocolo presente na dissertação. Surge a necessidade de uma verificação do sistema funcional completo com a utilização de ferramentas de rede, onde é possível verificar se a rede Ad Hoc se encontrava criada e utilizá-la normalmente. De seguida, foram realizados os testes de avaliação da duração do tempo de vida útil do nó em relação à sua disponibilidade energética, em função dos valores extremos (7 e 1) atribuídos ao parâmetro de *Willingness*. Com o valor 7, o nó assumia-se sempre como MPR, estando envolvido nas comunicações e participando no reencaminhamento de dados. Com o valor 1, o nó não participa no reencaminhamento de comunicações, efetuando apenas comunicações de presença e/ou envio de dados quase obrigatórios.

Com o parâmetro estático *Willingness*=3, é possível ter mais tempo de duração mas em contrapartida perde na taxa de transferência. O presente protocolo, apesar de conferir uma duração do tempo de vida útil dos nós inferior à obtida com o *Willingness*=3, consegue oferecer uma taxa de transferência maior, cerca de 2 vezes superior, e possibilitando serviços de comunicação em tempo real. Assim, o protocolo presente consegue realizar uma homogeneização da energia existente do nó para a realização de comunicações. É necessário ter em conta, o custo associado da taxa de transferência elevada em relação ao tempo de duração. Uma vez que, cada serviço tem os seus requisitos, na presente dissertação era necessário o envio de dados de uma forma rápida, mas existindo sempre uma tentativa de prolongar a longevidade do nó.

7. CONCLUSÕES

Usualmente, os protocolos de rede são, preferencialmente, executados e implementados em simulações. Permitindo, deste modo, uma maior capacidade de progressão na algoritmia, maior número de amostras [17, 30]. Foi apresentado uma adaptativa em relação à energia existente, num sistema composto por vários sensores ambientais projetado pelo *Future Cities*. Tentando uniformizar os gastos energéticos, na recolha de dados fornecidos pelos sensores e envio dos mesmos recorrendo a ligações WiFi. O consumo descrito dos vários sensores de baixo custo que, até à data não eram conhecidos. Sendo esta informação de consumo dos mesmos, um reforço na motivação para a manipulação de um protocolo com eficiência energética.

Nesta dissertação, recorreu-se a uma implementação de uma *testbed* obtendo valores reais. Assim, havendo dificuldades de interligação dos vários materiais necessários para a implementação da mesma. O objetivo desta dissertação foi a realização de uma experimentação num protocolo de rede conhecido, com vários elementos presentes num sistema existente, sendo este e os vários referidos cumpridos com sucesso.

Foi projetado um sistema autónomo e com capacidade de execução de 24 horas por dia. Com o equipamento MoPi, a fornecer uma variedade de funcionalidades, tais como encerramento correto, monitorização da energia nas baterias e na comunicação com a plataforma do

Raspberry Pi e os sensores. Sendo tendo cumprido os objetivos, nomeadamente, realizar um estudo do consumo energético do nó da rede, avaliar os vários protocolos e ajustar os parâmetros de forma a ter impacto no consumo energético e aumento da largura de banda conforme a energia remanescente das baterias.

Os vários testes e a existência de um ensaio prévio para a verificação do sistema permite que, este se integre e acumule novas implementações. Sendo comprovado nesta dissertação, que o foco da mesma não foi só aumentar a longevidade de um nó da rede, mas também aumentar a largura de banda por um período superior. Deste modo, esta manipulação torna-se numa mais-valia para o alargamento da recolha e envio rápido de dados pela rede.

7.1. TRABALHO FUTURO

Permanecem ainda lacunas remanescentes no sistema, tais como a incompatibilidade dos vários materiais constituintes e o *software* com falhas de segurança, sendo estes os objetivos de melhoramento num trabalho futuro. Com a possibilidade de realizar uma otimização da possível arquitetura do protocolo e eficiência do tempo de vida útil, ocorrendo a possibilidade de uma reutilização de canais WiFi.

Uma linha de investigação possível, na continuação deste projeto, passa por estudar, implementar e analisar a implementação da 2^a versão do protocolo OLSR. Sendo que a mesma já realizará otimizações de rotas, podendo ser esta otimização aproveitada para acrescentar a funcionalidade energética. Deste modo, num futuro recente a mesma já se encontrará numa fase de maturação superior relativamente aquando a realização desta dissertação.

Referências Documentais

- [1] F. Cities, “Future Cities - porto living lab,” [Online]. Available: <http://futurecities.up.pt/site/>. [Acedido em 15 Jan 2015].
- [2] S. Yang, X. Yang, J. A. McCann, T. Zhang, G. Liu e Z. Liu, “Distributed Networking in Autonomic Solar Powered Wireless Sensor Networks,” *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS/SUPPLEMENT*, vol. 31, n° 12, pp. 750-761, Dez 2013.
- [3] S. S. City, “Sense Smart City,” Luleå University of Technology, 2010. [Online]. Available: <http://sensesmartcity.org/index.htm>. [Acedido em 2 Fev 2015].
- [4] CitySense, “Malaga CitySense,” Top digital, [Online]. Available: <http://citysense.es/>. [Acedido em 2 Fev 2015].
- [5] T. V. Consortium, “Vital - the Future of Smart Cities,” The VITAL Consortium, 2015. [Online]. Available: <http://vital-iot.eu/>. [Acedido em 4 Fev 2015].
- [6] smartsantander, “smartsantander - smart city,” [Online]. Available: <http://www.smartsantander.eu/>. [Acedido em 25 Jan 2015].
- [7] J. Shovic, “projectcuracao/projectcuracao,” projectcuracao, 14 Mar 2014. [Online]. Available: <https://github.com/projectcuracao/projectcuracao>. [Acedido em 22 Jan 2015].
- [8] R. Pi, “Raspberry Pi,” [Online]. Available: <https://www.raspberrypi.org/>. [Acedido em 20 Fev 2015].
- [9] B. d. S. M. Fernandes, “SENSOR INTEGRATION FOR SMART CITIES USING MULTI-HOP NETWORKS,” ISEP, PORTO, 2015.

- [10] F. U. B. -. DES-Workgroup, “DES-TESTBED,” Freie Universität Berlin, 2010. [Online]. Available: <http://www.des-testbed.net/>. [Acedido em 2 Fev 2015].
- [11] M. Baar, H. Will, B. Blywis, T. Hillebrandt, A. Liers, G. Wittenburg e J. Schiller, “The ScatterWeb MSB-A2 Platform for Wireless Sensor Networks,” Freie Universität Berlin, Setembro 2008. [Online]. Available: <ftp://ftp.inf.fu-berlin.de/pub/reports/tr-b-08-15.pdf>. [Acedido em 2 Fev 2015].
- [12] “Freifunk Berlin,” Freie Universität, [Online]. Available: <https://berlin.freifunk.net/>. [Acedido em 3 Fev 2015].
- [13] ccbs-energia, “ccbs-energia,” [Online]. Available: <http://www.ccbs-energia.pt/>. [Acedido em 2 Jun 2015].
- [14] T. F. M. M. Coelho, “Encaminhamento com QoS em redes móveis Ad hoc,” 2013. [Online]. Available: <http://hdl.handle.net/1822/29098>.
- [15] B. Lawson, “Electropaedia,” Woodbank Communications Ltd, [Online]. Available: <http://www.mpoweruk.com/leadacid.htm>. [Acedido em 14 Jun 2015].
- [16] I. Buchmann, “Battery University,” [Online]. Available: <http://batteryuniversity.com/>.
- [17] S. Corson e R. Macker., “Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations,” 1999. [Online]. [Acedido em 10 Mar 2015].
- [18] G. V. Brignoni, “Estudo de Protocolos de Roteamento em Redes Ad Hoc,” UNIVERSIDADE FEDERAL DE SANTA CATARINA, Florianópolis, 2005.
- [19] T. Clausen e P. Jacquet., “Optimized Link State Routing Protocol (OLSR),” 2003. [Online].
- [20] “OLSR,” [Online]. Available: http://www.olsr.org/mediawiki/index.php/Main_Page. [Acedido em 2 Abr 2015].

- [21] A. Tønnesen, “UniK olsrd plugin implementation HOWTO,” [Online]. Available: <http://www.olsr.org/docs/olsrd-plugin-howto.html>. [Acedido em 1 Mai 2015].
- [22] R. R. Gupta, M. K. Mishra e M. Shrivastava, “Power Saving Routing Protocol for Ad hoc Networks based on AODV,” *International Journal of Computer Applications*, vol. 85, n° 19, pp. 18-23, Jan 2014.
- [23] A. Kumaravel e M.Chandrasekaran, “A COMPLETE STUDY ON POWER AWARE ROUTING PROTOCOL FOR MOBILE ADHOC NETWORK,” *IOSR-JEEE*, vol. 3, n° 15, pp. 71-75, 2014.
- [24] B. S. O. a. N. Meghanathan, “A REVIEW OF THE ENERGY EFFICIENT AND SECURE MULTICAST ROUTING PROTOCOLS FOR MOBILE AD HOC NETWORKS,” *International journal on applications of graph theory in wireless ad hoc networks and sensor networks*, vol. 2, n° 2, pp. 1-15, Jun 2010.
- [25] F. D. Rango, M. Fotino e S. Marano, “EE-OLSR: ENERGY EFFICIENT OLSR ROUTING PROTOCOL FOR MOBILE AD-HOC NETWORKS,” *IEEE*, pp. 1-7, 2008.
- [26] T. Sharma e S. Singh, “COMPARATIVE STUDY OF MULTIPATH EXTENSIONS OF AODV,” *IJRET: International Journal of Research in Engineering and tecnologia*, vol. 2, n° 9, pp. 494-500, Set 2013.
- [27] S. Zarza Sanchez, R. Fernandez-Canti, J. Lazaro, I. Ortega Gomez e J. Navarro, “Monitoring and remote control of energy consumption by WiFi networks,” *IEEE*, vol. 11, n° 14, pp. 11-15, 2014.
- [28] K. Machado, D. Rosário, E. Cerqueira, A. A. F. Loureiro, A. Neto e J. N. d. Souza, “A Routing Protocol Based on Energy and Link Quality for Internet of Things Applications,” *Sensors*, vol. 13, n° 2, pp. 1942-1964, 2013.
- [29] X. Jing e M. J. Lee, “Energy-Aware Algorithms for AODV in Ad Hoc Networks,” *winlab.rutgers.edu*, vol. 1, n° 1, pp. 1-6, 2004.

- [30] J. Vazifehdan, R. Prasad e I. Niemegeers, “Energy-Efficient Reliable Routing Considering Residual Energy in Wireless Ad Hoc Networks,” *Mobile Computing, IEEE Transactions on*, vol. 13, n° 2, pp. 434-447, Feb 2014.
- [31] A. Mesleh, N. Al-Zabin, J. Abdel-Jalil, H. Abu-Asbeh, N. Al-Mutlaq, H. Abu-Matter e M. Al-Hunati, “AODV and DSR energy-aware routing algorithms: a comparative study,” *International Journal of Computer Science Issues*, vol. 9, n° 6, pp. 266-271, Nov 2012.
- [32] F. U. Berlin, “Optimized Link State Routing (OLSR),” [Online]. Available: <http://www.des-testbed.net/content/optimized-link-state-routing-olsr>. [Acedido em 23 Abr 2015].
- [33] F. Lee, “Routing in Mobile Ad Hoc Networks,” [Online]. Available: <http://cdn.intechweb.org/pdfs/12861.pdf>. [Acedido em 20 Mai 2015].
- [34] M. Maleki, K. Dantu e M. Pedram, “Power-aware Source Routing Protocol for Mobile Ad Hoc Networks,” *Univ. of Southern California*, vol. 1, n° 1, pp. 72-75, 2002.
- [35] M. Fotino e F. D. Rango, “Energy Issues and Energy aware Routing in Wireless Ad-hoc Networks,” *Marco Fotino and Floriano De Rango (2011). Energy Issues and Energy Aware Routing in Wireless Ad Hoc Networks, Mobile Ad-HocInTech*, pp. 1-17, 2011.
- [36] M. Radi, B. Dezfouli, K. A. Bakar e M. Lee, “Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges,” *Sensors*, vol. 12, n° 12, pp. 650-685, Jan 2012.
- [37] P. Fortuna, “Pedro Fortuna Websiste,” [Online]. Available: <http://telecom.inescporto.pt/~pfortuna/olsr.php>. [Acedido em 2 Abr 2015].
- [38] A. Industries, “adafruit/Adafruit_INA219,” Adafruit Industries, 25 Mai 2012. [Online]. Available: https://github.com/adafruit/Adafruit_INA219. [Acedido em 20 Jan 2015].

- [39] A. -. L. Ada, “Adafruit INA219 Current Sensor Breakout,” 4 Mai 2015. [Online]. Available: <https://learn.adafruit.com/adafruit-ina219-current-sensor-breakout>. [Acedido em 22 Feb 2015].
- [40] B. Wann, “Solar/battery/load power logging with Raspberry Pi and INA219,” 6 Abr 2014. [Online]. Available: <https://binaryfury.wann.net/2014/04/solarbatteryload-power-logging-with-raspberry-pi-and-ina219/>. [Acedido em 15 Mar 2015].
- [41] C. Wilson, “Understanding the INA219 I2C Current/Power Monitor,” 6 jul 2014. [Online]. Available: <http://cdwilson.us/articles/understanding-the-INA219/>. [Acedido em 22 Feb 2015].
- [42] Mopi, “MoPi: Hot-Swap Mobile Power for the Pi,” Hamish Cunningham and The University of Sheffield, 1995-2015. [Online]. Available: <https://pi.gate.ac.uk/pages/mopi.html>. [Acedido em 5 Abr 2015].
- [43] V. Energy, “Victron Energy Open Source,” Victron Energy, 31 Mar 2015. [Online]. Available: http://www.victronenergy.com/live/open_source:start. [Acedido em 20 Mai 2015].
- [44] J. Perez, “jepefe/bmvmonitor,” 11 Mar 2014. [Online]. Available: <https://github.com/jepefe/bmvmonitor>. [Acedido em 7 Jun 2015].
- [45] karioja, “karioja/vedirect,” 26 Set 2014. [Online]. Available: <https://github.com/karioja/vedirect>. [Acedido em 5 Jun 2015].
- [46] L. Strand, “Linux Optimized Link State Routing Protocol (OLSR) IPv6 HOWTO,” 23 04 2004. [Online]. Available: <http://www.tldp.org/HOWTO/OLSR-IPv6-HOWTO/index.html>. [Acedido em 12 Mai 2015].
- [47] E. A. Panaousis, G. Drew, G. P.Millar, T. A.Ramrekha e C. Politis, “A Test-bed Implementation for Securing OLSR in Mobile ad hoc networks,” *International Journal of Network Security & Its Applications*, vol. 2, nº 4, Oct 2010.

- [48] openwrt, “OLSR Mesh,” [Online]. Available: <http://wiki.openwrt.org/doc/howto/mesh.olsr>. [Acedido em 5 jun 2015].
- [49] A. González, R. Aquino, W. Mata, A. Ochoa, P. Saldaña e A. Edwards, “Open-WiSe: A Solar Powered Wireless Sensor Network Platform,” *Sensors*, vol. 12, nº 12, pp. 8204-8217, Jun 2012.
- [50] Wireshark, “Tshark - The Wireshark Network,” [Online]. Available: <https://www.wireshark.org/docs/man-pages/tshark.html>. [Acedido em 28 jun 2015].

Anexo A. Código de Adaptação Energética

```
#!/usr/bateriain/env python
# -*- coding: utf-8 -*-
__author__ = 'Agostinho'
import subbateriaprocess as sp
import time
import os, serial, json
import datetime

def bateriaate():
    cmd= "sudo mopicli -v1"
    p = sp.Popen(cmd,shell=True, stderr=sp.PIPE,
stdout=sp.PIPE)
    pid=p.pid
    stdout_value,stderr_value= p.communicate()
    xlp=p.poll()
    failed=p.returncode
    a,l,c,bateriaV_real = stdout_value.split(" ")
    print "bateria voltage: %s" %bateriaV_real
    return bateriaV_real

class vedirect:

    def __init__(self, serialport):
        self.serialport = serialport
        self.ser = serial.Serial(serialport, 19200,
timeout=10)
        self.header1 = '\r'
        self.header2 = '\n'
        self.delimiter = '\t'
        self.key = ''
        self.value = ''
        self.bateriaytes_sum = 0;
        self.state = self.WAIT_HEADER
        self.dict = {}

(WAIT_HEADER, IN_KEY, IN_VALUE, IN_CHECKSUM) = range(4)

def input(self, bateriayte):
    if self.state == self.WAIT_HEADER:
        self.bateriaytes_sum += ord(bateriayte)
        if bateriayte == self.header1:
            self.state = self.WAIT_HEADER
```

```

        elif bateriayte == self.header2:
            self.state = self.IN_KEY

            return None
        elif self.state == self.IN_KEY:
            self.bateriaytes_sum += ord(bateriayte)
            if bateriayte == self.delimiter:
                if (self.key == 'Checksum'):
                    self.state = self.IN_CHECKSUM
                else:
                    self.state = self.IN_VALUE
            else:
                self.key += bateriayte
            return None
        elif self.state == self.IN_VALUE:
            self.bateriaytes_sum += ord(bateriayte)
            if bateriayte == self.header1:
                self.state = self.WAIT_HEADER
                self.dict[self.key] = self.value;
                self.key = '';
                self.value = '';
            else:
                self.value += bateriayte
            return None
        elif self.state == self.IN_CHECKSUM:
            self.bateriaytes_sum += ord(bateriayte)
            self.key = ''
            self.value = ''
            self.state = self.WAIT_HEADER
            if (self.bateriaytes_sum % 256 == 0):
                self.bateriaytes_sum = 0
                return self.dict
            else:
                self.bateriaytes_sum = 0

    else:
        raise AssertionError()

def read_data(self):
    while True:
        bateriayte = self.ser.read(1)
        packet = self.input(bateriayte)

def read_data_single(self):
    while True:
        bateriayte = self.ser.read(1)
        packet = self.input(bateriayte)
        if (packet != None):
            return packet

```

```

def read_data_callbateriaack(self,
callbateriaackFunction):
    while True:
        bateriayte = self.ser.read(1)
        packet = self.input(bateriayte)
        if (packet != None):
            callbateriaackFunction(packet)

def print_data_callbateriaack(data):
    print data

ve = vedit('/dev/ttyUSBateria0')
ALL=(ve.read_data_single())
parsed_data = json.dumps(ALL)
parsed_json = json.loads(parsed_data)

load = (parsed_json["LOAD"])
bateria_corrente = (parsed_json["V"])
bateriaV_real = int(bateria_corrente)
WILL = 0

if int(bateriaV_real) >= 12500:
    WILL = 6
if int(bateriaV_real) >= 12000 and int(bateriaV_real) <
12500:
    WILL = 5
if int(bateriaV_real) >= 11500 and int(bateriaV_real) <
12000:
    WILL = 4
if int(bateriaV_real) >= 11000 and int(bateriaV_real) <
11500:
    WILL = 3
if int(bateriaV_real) >= 10500 and int(bateriaV_real)<11000:
    WILL = 2
if int(bateriaV_real) >= 9500 and int(bateriaV_real)
<10500:
    WILL = 1

bateria= WILL
arquivo = open("/etc/olsrd/olsrd.conf", "r")
linhas = arquivo.readlines()
arquivo.close()
linhas_a_escrever = ''
for linha in linhas:
    if "Willingness" in linha:
        f,valor_energia = linha.split("\t")
        if int(valor_energia) == bateria:
            print "Willingess not changed %i\n" %bateria
            verificar=bool(0)
        if int(valor_energia) != bateria:

```

```

        print "Willingess updated to %i\n" %bateria
        linhas_a_escrever += 'Willingness\t%i\n' %bateria
    verificar=bool(1)
    continue
    linhas_a_escrever += linha
arquivo = open("/etc/olsrd/olsrd.conf","w")
arquivo.write(linhas_a_escrever)
arquivo.close()
print "Proxima actualizacao daqui a 15 min \n"
if verificar is True:
    print "a actualizar willingness"
    os.system("sudo killall -SIGHUP olsrd")
    print "a levantar o daemon"
    os.system("sudo olsrd")
log=open("mpr_ea.txt","a")
log.write( str(bateriaV_real) + ";" + str(bateria) + ";" +
str(current_time) + ";" + str(valor_energia) + "\n")
log.close()
current_time = datetime.datetime.now().time()
print current_time

```