

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA – PATOS DE MINAS
ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÕES

AMIR BERNARDO BRAGA

**ANÁLISE DE DESEMPENHO DOS PROTOCOLOS CTP E
RPL EM AMBIENTE MÓVEL**

Patos de Minas - MG
2018

AMIR BERNARDO BRAGA

ANÁLISE DE DESEMPENHO DOS PROTOCOLOS CTP E RPL EM AMBIENTE MÓVEL

Trabalho de conclusão 2 apresentado à banca examinadora como requisito parcial para a aprovação na disciplina de TCC2 da graduação em Engenharia Eletrônica e de Telecomunicações, da Faculdade de Engenharia Elétrica, da Universidade Federal de Uberlândia, Campus Patos de Minas.

Orientador: Prof. Me. Alexander Bento Melo

AMIR BERNARDO BRAGA

ANÁLISE DE DESEMPENHO DOS PROTOCOLOS CTP E RPL EM AMBIENTE MÓVEL

Trabalho de conclusão 2 apresentado à banca examinadora como requisito parcial para a aprovação na disciplina de TCC2 da graduação em Engenharia Eletrônica e de Telecomunicações, da Faculdade de Engenharia Elétrica, da Universidade Federal de Uberlândia, Campus Patos de Minas.

Orientador: Prof. Me. Alexander Bento Melo

Patos de Minas, 11 de dezembro de 2018

COMISSÃO EXAMINADORA:

Prof. Me. Alexander Bento Melo
Universidade Federal de Uberlândia
Orientador

Prof. Dr. Júlio Cezar Coelho
Universidade Federal de Uberlândia
Examinador

Prof. Dr. André Luiz Aguiar da Costa
Universidade Federal de Uberlândia
Examinador

AGRADECIMENTOS

Primeiramente, agradeço a Deus pela vida, e pelas oportunidades concebidas.

Agradeço ao Prof. Me. Alexander Bento Melo, que me auxiliou no desenvolvimento deste projeto, sem o qual, não seria possível a execução do mesmo.

Agradeço aos professores da Universidade Federal de Uberlândia, que me proporcionou acesso ao conhecimento.

Agradeço aos familiares, que em todos os momentos, deram o total apoio, moral e psicológico para concluir este objetivo.

Agradeço meus inúmeros colegas e amigos, tanto os conhecidos na universidade, quanto os conhecidos de outrora.

Agradeço por fim, à instituição Universidade Federal de Uberlândia, sem a qual, nenhum objetivo poderia ser concluído.

Resumo

Com o avanço da Internet das Coisas, novos dispositivos serão capazes de se conectarem à internet e com isso os desafios que surgem para permitir que a rede absorva estes novos dispositivos é enorme, a Internet atual ainda não suporta de forma ampla e bem sedimentada a implantação da Internet das Coisas. Existem vários problemas tecnológicos nas áreas de segurança, mobilidade, qualidade de serviço e principalmente escalabilidade. Atualmente as empresas trabalham no desenvolvimento de protocolos para atender as novas aplicações e o novo perfil de tráfego que a rede irá ter, uma área que tem sido muito estudada e que terá um impacto muito profundo na questão da viabilidade da Internet das Coisas é a que trata do desenvolvimento e adequação de novos protocolos de roteamento, para ser ter ideia se olharmos somente para um rede de sensores sem fio, onde a disponibilidade de energia é baixa, o protocolo de roteamento precisa levar esta questão em consideração, para não tornar inviável a aplicação. Um exemplo de protocolo que evoluiu, não somente mais também em função do aumento da demanda de endereçamento, foi o protocolo IP, o IPv4 tem um suporte para aproximadamente 4 bilhões (2^{32}), de endereços, enquanto o IPv6 possui um número astronômico de endereços (2^{128}). A Internet das coisas, não se limita a objetos estáticos, sendo composta também por dispositivos móveis, como por exemplo sensores que estarão instalados em carros, acompanhando doentes em suas vidas diárias, em diversas colheitadeiras etc.

O objetivo deste trabalho é estudar o desempenho dos protocolos CTP e RPL, desenvolvidos para a utilização em redes de sensores sem fio, onde os requisitos relacionados ao consumo de energia são bem restritivos. Para permitir um estudo amplo serão desenvolvidas simulações utilizando-se o simulador de redes COOJA, instalado no sistema operacional CONTIKI. As simulações levarão em conta dispositivos móveis e fixos de forma e comparar o desempenho dos protocolos nos dois casos.

Palavras chaves: Redes de baixa potência; CTP; RPL; Internet Das Coisas.

Abstract

As the advancement of the Internet of Things, new devices will be able to connect to the internet and with this the challenges that arise to allow the network to absorb these new devices is huge. The current Internet still does not support in a wide and well sedimented way the Internet of Things deployment. There are several technological problems in the areas of security, mobility, quality of service and mainly scalability. Currently, companies works on the development of protocols to meet the new applications and the new traffic profile that the network will have, an area that has been much studied and will have a very deep impact on the Internet of Things viability is the new routing protocols development and adaptation, to have an idea if we only look at a wireless sensor network, where the energy availability is low, the routing protocol needs to take this issue into consideration, in order to make it a viable application. An example of a protocol that has evolved, not only but most because the increased demand for addressing, was the IP protocol, IPv4 supports approximately 4 billion (2^{32}) addresses, while IPv6 has an astronomic address numbers (2^{128}). The Internet of things is not limited to static objects, it is also made up for mobile devices, such as sensors that will be installed in cars, accompanying patients in their daily lives, in several harvesters, etc.

The objective of this monography is to study the CTP and RPL protocols performance, developed for wireless sensor networks where the requirements related to the energy consumption are very constrained. To fully study and understand the protocols, simulations will be taken using a network simulator named COOJA that is installed in the operational system CONTIKI. Simulations will be made using static devices and mobile devices, and the performance will be compared on both cases.

Keywords: Low Power and Lossy Networks; CTP; RPL; Internet of Things; IOT.

LISTA DE FIGURAS

Figura 1 - Rede RPL com 2 DODAGs e duas instâncias	18
Figura 2 - Seleção de parentes no CTP.....	20
Figura 3 - Cenário de teste com um nó móvel.....	24
Figura 4 - Cenário de teste com dois nós móveis	25
Figura 5 - Cenário com nós aleatórios.....	26
Figura 6 - Configuração 1º Cenário.....	28
Figura 7 - Configuração 2º Cenário.....	37
Figura 8 - Configuração 3º Cenário.....	45

LISTA DE GRAFICOS

Gráfico 1 - Saltos 1º Cenário	29
Gráfico 2 - Energia Consumida 1º Cenário RPL.....	30
Gráfico 3 - Pacotes Recebidos 1º Cenário RPL	31
Gráfico 4 - Intervalo de sinalização 1º Cenário RPL	32
Gráfico 5 - Energia Consumida 1º Cenário CTP	34
Gráfico 6 - Pacotes Recebidos 1º Cenário CTP	35
Gráfico 7 - Intervalo de Sinalização 1º Cenário CTP	36
Gráfico 8 - Saltos 2º Cenário	38
Gráfico 9 - Energia Consumida 2º Cenário RPL.....	39
Gráfico 10 - Pacotes Recebidos 2º Cenário RPL	40
Gráfico 11 - Intervalo de Sinalização 2º Cenário RPL.....	41
Gráfico 12 - Energia Consumida 2º Cenário CTP	42
Gráfico 13 - Pacotes Recebidos 2º Cenário CTP	43
Gráfico 14 - Intervalo de Sinalização 2º Cenário CTP	44
Gráfico 15 - Saltos 3º Cenário	46
Gráfico 16 - Energia Consumida 3º Cenário RPL.....	47
Gráfico 17 - Pacotes Recebidos 3º Cenário RPL	48
Gráfico 18 - Intervalo de Sinalização 3º Cenário RPL.....	49
Gráfico 19 - Energia Consumida 3º Cenário CTP.....	51
Gráfico 20 - Pacotes Recebidos 3º Cenário CTP	52
Gráfico 21 - Intervalo de Sinalização 3º Cenário CTP	53

LISTA DE QUADROS

Quadro 1 - Quadro de Dados 1º Cenário RPL.....	33
Quadro 2 - Quadro de Dados 1º Cenário CTP.....	36
Quadro 3 - Quadro de Dados 2º Cenário RPL.....	41
Quadro 4 - Quadro de Dados 2º Cenário CTP.....	45
Quadro 5 - Quadro de Dados 3º Cenário RPL.....	50
Quadro 6 - Quadro de Dados 3º Cenário CTP.....	54

LISTA DE ABREVIATURAS E SIGLAS

DAG - Gráficos Acíclicos diretos (*Direct Acyclic Graphs*)

DAO – Anuncio do destino do objeto (*Destination Advertisement Object*)

DIO – Informações do objeto DODAG (*DODAG Information Object*)

DIS – Solicitação de informação DODAG (*DODAG Information Solicitation*)

DODAG – Gráficos Acíclicos com Destino Orientado (*Destination-Oriented Acyclic Graph*)

ETX – Valor de transmissão esperada (*Expected transmission Count*)

GPS – Sistema de posição global (*Global Position System*)

ICMPv6 –Protocolo de mensagem de controle versão 6 (*Internet Control Message Protocol version 6*)

IETF - Força Tarefa da Engenharia da Internet - (*Internet Engineering Task Force*).

IOT - Internet Das Coisas - (*Internet of Things*).

LLNs - Redes de Baixa potência e com Perdas (*Low Power and Lossy Networks*).

NF - Nós Fixos.

NM - Nós Móveis

P2MP – Ponto-Multiponto (*point to multipoint*)

P2P – Ponto a Ponto (*point to point*)

PRR- Taxa de Recepção de Pacotes (*Packet Reception Rate*)

RPL – Protocolo de Roteamento Com Perdas (*Routing Protocol for Low Power and Lossy Network*)

WSN - Redes de Sensores Sem Fio - (*Wireless Sensor Networking*)

SUMÁRIO

CAPÍTULO 1	13
1.1. INTRODUÇÃO	13
1.2. TEMA DO PROJETO	14
1.3. PROBLEMATIZAÇÃO	14
1.4. HIPÓTESES	15
1.5. OBJETIVOS	15
1.5.1. Objetivos Gerais	15
1.5.2. Objetivos Específicos	16
1.6. JUSTIFICATIVAS	16
1.7. CONSIDERAÇÕES FINAIS	16
CAPÍTULO 2	17
2.1. CONTIKI	17
2.2. COOJA	17
2.3. RPL	18
2.4. CTP	19
2.5. MOBILIDADE	21
2.6. CONSIDERAÇÕES FINAIS	23
CAPÍTULO 3	24
3.1. CENÁRIOS DE TESTES	24
3.2. RECURSOS NECESSÁRIOS	27
3.3. CONSIDERAÇÕES FINAIS	27
CAPÍTULO 4	28
4.1. ANÁLISE DOS EXPERIMENTOS	28
CAPÍTULO 5	55
5.1. CONSIDERAÇÕES FINAIS	55
5.2. ESTUDOS FUTUROS	55
BIBLIOGRAFIA	57

CAPÍTULO 1

1.1. INTRODUÇÃO

A Internet das Coisas (do inglês *Internet of Things* (IoT)) surgiu juntamente com os avanços em diversas áreas, tais como: microeletrônica; sistemas embarcados; sensoriamento e comunicação. A IoT está sendo amplamente debatida e estudada, devido à sua abrangência nas mais diversas áreas de atividades, como, plantio, indústrias ou veículos [1]. Pode-se dizer que devido à limitação da Internet IPv4, a Internet das Coisas utilizando o IPv6, é uma extensão da mesma, que proporciona aos objetos utilizados no cotidiano a capacidade de se conectarem à Internet, desde que os mesmos possuam os hardwares necessários para se fazer a comunicação com a rede. Essa capacidade de conexão com a internet adquirida pelos objetos, viabilizará seu controle remotamente, podendo assim, provê algum tipo de serviço mesmo que o dispositivo esteja em um local completamente distante de quem o controla. Tais objetos, fazem parte das redes de sensores sem fio WSN (*Wireless Sensor Networking*). Como os dispositivos possuem baixa capacidade de hardware, foi necessário a implementação de um novo tipo de protocolo de roteamento, denominado RPL. [1]

RPL é um protocolo de roteamento que utiliza o IPv6 para redes LLNs (*Low Power and Lossy Networks*) ou redes de baixa potência e com perdas, que utiliza uma rota otimizada para transmitir o tráfego de dados dos nós para um nó central também denominado de nó raiz por meio do mecanismo *distance vector* [1]. Este protocolo foi proposto pelo IETF ROLL e foi projetado como uma solução para dispositivos com baixa potência e baixo custo comunicações serem executados em redes de grande escala. O RPL possui características importantes como utilizar mecanismos que facilitam o encaminhamento de dados e minimiza a complexidade do roteamento, o mesmo é recomendado para diminuir o requisito de memória e cabeçalhos de sinalização de roteamento [2]. Por ser desenvolvido para este novo ambiente o protocolo RPL é considerado um padrão de roteamento para IoT. Seu objetivo é otimizar o esquema de roteamento para um padrão em que todos os pacotes são enviados para um conjunto de roteadores de borda conectados à internet. [3]

Existem outros protocolos, como o protocolo CTP (*Collection Tree Protocol*) que é um

protocolo de coleta de dados bem conhecido no âmbito de redes de sensores sem fio devido ao fato de sua eficiência energética, além de possuir uma alta PRR (*Packet Reception Rate*). [4]

1.2. TEMA DO PROJETO

Análise de desempenho dos protocolos CTP e RPL em ambiente móvel.

1.3. PROBLEMATIZAÇÃO

Devido a varias limitações das redes sem fio, os modelos de protocolos para as redes de sensores sem fio são bastante desafiadores. Existem muitos problemas que podem limitar o funcionamento do roteamento na rede, como:

- A capacidade limitada de energia: Como os sensores são alimentados com bateria, existem uma limitação da capacidade de energia;
- Localização dos sensores: Um desafio dos protocolos de roteamento é administrar a localização dos sensores. Para isso, muitos protocolos assumem que os sensores são equipados com o Sistema de Posição Global (*GPS*), ou utilizem alguma técnica de localização para informar suas respectivas posições;
- Limitados recursos de hardware: Os sensores possuem capacidade limitada de armazenamento e de processamento, assim, possuindo capacidade computacional limitada;
- Características ambientais não confiáveis: uma rede de sensores usualmente opera em um ambiente não confiável. A topologia da rede, que é definida pelo sensor e pelas comunicações entre sensores, mudam frequentemente devido a adição de algum novo sensor ou quando um sensor é excluído;
- Escalabilidade: os protocolos de roteamento devem ser capazes de acomodar novos sensores, e também, sensores adicionados podem não ter a mesma capacidade em termos de energia, processamento e comunicação, do que os já existentes. Então a comunicação entre os sensores pode não ser simétricas, ou seja, um sensor pode não ter capacidade de ter comunicação bidirecional com a

mesma capacidade. E isso tem que ser levado em conta nos protocolos de roteamento. [5]

Alguns protocolos que tentam sanar estes problemas são o protocolo RPL e também o CTP.

O protocolo CTP é bastante eficiente em ambientes estáticos, entretanto, quando se trata de ambientes móveis, seu desempenho tende a decair, em relação ao estático. [6]

Por sua vez, utilização do protocolo RPL vem se tornando cada vez mais popular, passando a ser o mais utilizado quando se trata de redes de baixa potência com perdas. Para aqueles cenários em que contempla somente nós fixos o IETF (*Internet Engineering Task Force*), grupo que produz documentos técnicos relevantes de alta qualidade para projetos e gerenciamento da internet, definem especificações a serem seguidas, mas quando se trata de nós móveis com dispositivos móveis, não existe ainda uma definição concreta, então como não há um padrão definido, não distinguimos nós móveis de estáticos, podendo assim gerar um menor desempenho na rede. [1]

1.4. HIPÓTESES

Com base no que foi dito acima os protocolos RPL e CTP tem vantagens e falhas dependendo dos cenários nos quais são utilizados, sendo assim analisar o desempenho em cenários distintos ajudará a entender quando utilizar um ou outro protocolo.

1.5. OBJETIVOS

1.5.1. Objetivos Gerais

O objetivo desta pesquisa é avaliar o desempenho dos protocolos de roteamento CTP e RPL em cenários distintos, com mobilidade(nós móveis) e sem mobilidade(nós fixos), de forma a determinar o comportamento e desempenho dos mesmos.

1.5.2. Objetivos Específicos

- Entender o funcionamento dos protocolos RPL e CTP;
- Estudar o simulador Contiki/Cooja;
- Definir os cenários de simulações;
- Definir as variáveis de desempenho a serem estudadas.

1.6. JUSTIFICATIVAS

Como não há um padrão definido em relação aos nós móveis [1], é necessário testar várias configurações, em diferentes cenários, afim de obter um resultado demonstrando qual dos protocolos, CTP ou RPL, proporciona um melhor desempenho.

1.7. CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentado uma introdução às características dos protocolos RPL e CTP e também os objetivos que levaram a desenvolver esta pesquisa. O próximo capítulo analisa o método de funcionamento dos protocolos.

CAPÍTULO 2

2.1. CONTIKI

Contiki é um sistema operacional de código aberto para a Internet das Coisas, sendo usado para conectar micro controladores de baixa potencia à internet. O Contiki provê uma grande estrutura para à comunicação pela internet de dispositivos de baixa potencia, suportando os padrões IPv4 e IPv6, além dos mais recentes padrões para baixas potências, como: 6LoWPAN, RPL e CoAP. Com o Contiki, o desenvolvimento de aplicações é rápido e fácil, podendo ser desenvolvidas em C, além do mais, é possível utilizar o simulador de rede sem fio COOJA. Algumas características importantes do Contiki são:

- Alocação de memória: Contiki é desenvolvido para pequenos sistemas, tendo poucos KiloBytes de memorias disponíveis. Portanto, o Contiki é altamente eficiente no que se trata de uso de memoria;
- Rede IP completa: O Contiki possui compatibilidade com uma ampla variedade de protocolos como, UDP, TCP e os mais recentes RPL e CoAP;
- Potência: Contiki foi produzido para operar em sistemas de baixíssimas potências, como sistemas que podem funcionar por anos com uma simples bateria, além de possuir mecanismos para estimar o consumo de potência. [7]

2.2. COOJA

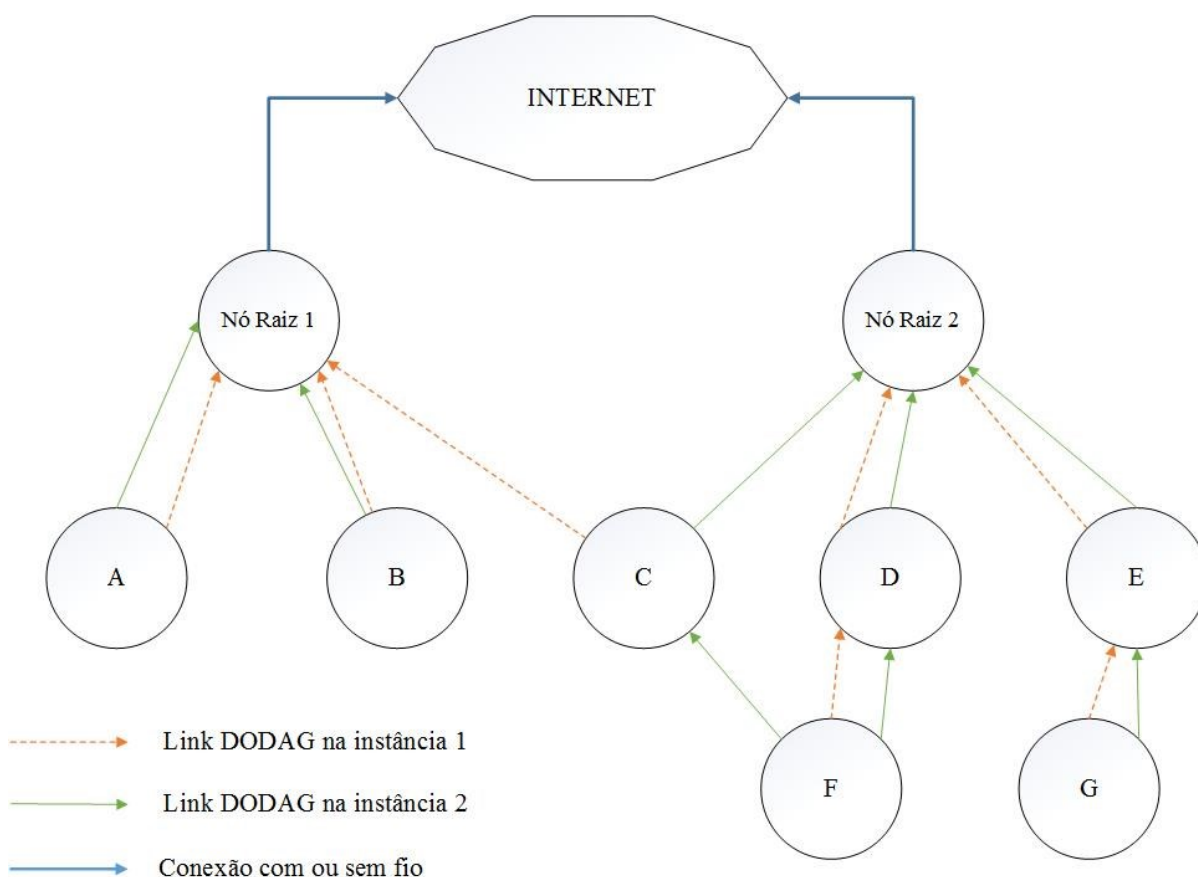
COOJA (*Contiki OS Java*) é um simulador de redes sem fio baseado em Java. A principal função do COOJA é a capacidade de expansão, no qual interfaces e plug-ins são utilizados. Uma interface representa uma propriedade de um nó sensor ou dispositivo, como a posição ou a transmissão de rádio. Um plug-in é utilizado para interagir com uma simulação, por exemplo para controlar a velocidade ou verificar o tráfego de rede entre os nós simulados. Um outro exemplo de plug-in é o de mobilidade, da qual simula um sensor em movimento. Para se conectar ao Contiki, o Cooja utiliza a Interface Nativa do Java (*JNI*), permitindo assim que

uma simulação de uma aplicação ocorra em um sistema real Contiki. Utilizando essa estrutura, qualquer simulação de aplicação pode funcionar em um sensor real. [8]

2.3. RPL

O RPL é baseado no conceito topológico de grafos acíclicos dirigidos, DAG (*Direct Acyclic Graphs*). O RPL foi otimizado para coletar dados de vários sensores (*convergecast traffic*), criando um DODAG (*Destination-Oriented Acyclic Graph*) no roteador de borda ou nó raiz, que pode funcionar como uma porta para a internet. A rede pode conter múltiplas instâncias do RPL, e estas podem conter vários DODAG, sendo um para cada nó raiz. Vale salientar que em uma instância específica, um nó só pode pertencer a um DODAG [3], como na Figura 1.

Figura 1 - Rede RPL com 2 DODAGs e duas instâncias



FONTE: [3]

Com o objetivo de manter e atualizar as rotas, o RPL utiliza quatro tipos de mensagens de controle utilizando o ICMPv6, são elas:

- DIO: *DODAG Information Object*: Esse tipo de mensagem de controle contém informações sobre o DODAG, como, o identificador do DODAG, a instância em que se encontra o DODAG (*RPLInstanceID*) ou as métricas utilizadas para computar a rota. As mensagens são enviadas de modo *broadcast* através da rede para construir e manter o DODAG. O Nó raiz é o único nó que pode começar a disseminação desta mensagem de controle. O controle do intervalo de envio de um DIO, é determinado pelo algoritmo *Trickle*;
- DIS: *DODAG Information Solicitation*: Um nó pode enviar essa mensagem de forma *unicast* ou *multicast* para seus vizinhos, o objetivo dessa mensagem é solicitar informações de configuração. Em resposta, será recebido um pacote DIO;
- DAO: *Destination Advertisement Object*: Esses tipos de mensagens são opcionais e são usadas somente quando rotas descendentes são necessárias, por exemplo em aplicações que requer tráfego ponto-multiponto (*P2MP*) ou ponto a ponto (*P2P*). As mensagens DAO enviadas de modo *unicast* aos vizinhos selecionados ou ao nó raiz. O DAO é a única mensagem de controle que pode ser reconhecida pelo destinatário;
- DAO-ACK: *Destination Advertisement Object Acknowledgment*: Esse pacote indica se os vizinhos que receberam o DAO querem agir como um salto (*hop*) na rota descendente, para o nó que está enviando mensagem. [2] [3]

2.4. CTP

O protocolo CTP é um protocolo de coleção baseado em árvore do qual o principal objetivo é determinar o melhor caminho de seus sensores, chamados de nós fontes, para o nó de coleta de informação, que é denominado nó raiz e é colocado no início da árvore. Se existirem múltiplas rotas na rede, a informação é roteada para a rota com menor custo/distancia.

No começo da sessão da rede, os nós raízes enviam alerta de identificação deles mesmo na rede, e, os nós fontes utilizam esses alertas para escolher o próximo salto em direção ao nó raiz baseado em um roteamento gradiente.

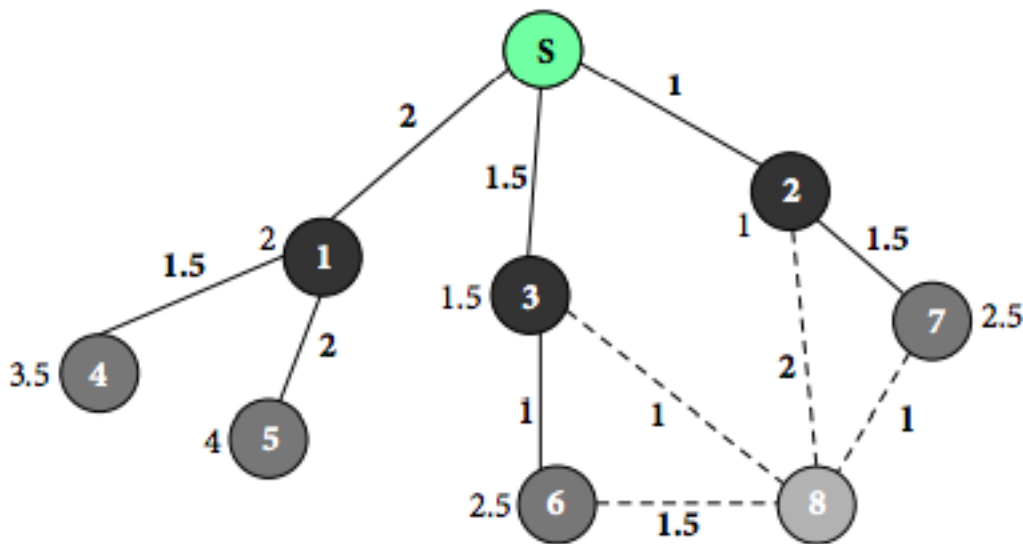
O protocolo CTP utiliza o valor da taxa de transmissão esperada (*Expected*

Transmission) como seu roteamento gradiente. ETX é um indicador de qualidade do link, sendo que para cada link, o valor do ETX é calculado como o número de transmissões que se leva para que um nó envie um pacote *unicast* para seu vizinho e seja recebido com sucesso.

Quando um nó raiz envia um requerimento de informação, cada nó atualiza o número de saltos e o custo total no requerimento, antes de passar os dados adiante. Cada nó escolhe o vizinho com o menor valor ETX como seu próximo salto (parente) na coleção da árvore. [9]

Na figura 2, mostra a seleção de parentes do nó 8. O nó 8 é vizinho e esta em comunicação com os nós 2,3,6 e 7. Pela figura, tem-se que $ETX_2 = 1$, $ETX_3 = 1.5$, ETX_6 e $ETX_7 = 2.5$. Como o nó escolhe sempre o menor valor de ETX como seu próximo parente, o nó 8 escolhera o nó 2 como seu parente.

Figura 2 - Seleção de parentes no CTP



FONTE: [10]

Para este estudo não será levado em conta o peso do link, apenas do ETX. Para efeito de esclarecimento, caso o valor do link fosse levado em conta, ao analisar o nó 8, o melhor parente a ser escolhido seria o nó 3.

2.5. MOBILIDADE

Cada DODAG possui determinados pais, possuindo aquele preferido, bem como uma lista com aqueles possíveis substitutos, para caso o alcance do sinal esteja fora de monitoramento. Como o alcance para descoberta de novos sensores compostos na rede são relativamente baixo, devido a baixa potencia dos mesmos, é muito comum que alguns nós não se encontrem no alcance necessário, o que ocasiona a perda do seu pai preferido. Caso isso ocorra, o nó espera uma mensagem para descobrir uma instância RPL e dissemina então um novo Rank que é um parâmetro que define a posição de um nó em respeito ao nó raiz, tendo em vista um novo DODAG. [1] [11]

Nem sempre um novo Rank resolverá o problema do alcance pois poderá receber um Rank maior que o já utilizado anteriormente, e também temos que considerar este tempo de troca, ou seja, a espera para um novo. Este tempo em muitas das vezes poderá ser grande, gerando uma perda de conexão, e como consequência a perda de pacotes transmitidos e recebidos. Um dos fatores para determinar o tempo de espera é o uso do algoritmo *trickle timer*. [1]

Quando as informações de roteamento forem transmitidas podem acontecer de um nó móvel interferir na rota, fazendo com que o caminho não seja finalizado corretamente. Também poderá acontecer de a rede entrar em loop com a criação de novas posições de nós. O DODAG possui certa instabilidade e nós móveis apresentam um menor desempenho, bem como consomem uma maior energia, pois seu rádio deverá estar ligado sempre para que haja uma atualização e recebimento de mensagens. Visando solucionar os problemas acima mencionados, foram propostas algumas soluções importantes, as quais veremos a seguir. [1]

A mobilidade é uma das principais inconsistências no RPL. Existem dois modelos principais que abordam a mobilidade: o algoritmo *trickle* e a transmissão de pacotes ICMPv6, controlada pelo protocolo ND (*Neighbor Discovery*). [11]

ALGORITMO TRICKLE: Protocolos de coleção tradicionais em redes de baixa potencia, tipicamente envia mensagens de controle *broadcast* em um intervalo fixo de tempo. Um intervalo pequeno de tempo requer mais largura de banda e também mais energia. Um intervalo de tempo mais largo, usa menos largura de banda e também menos energia, mas, erros de topologia pode ocorrer devido a incapacidade de lidar com redes dinâmicas.

A ideia básica do algoritmo *trickle* (RFC 6206) é propagar pequenos sinais (*beacons*) se houver mudança na rota de informação. O algoritmo utiliza uma estratégia mirando na rápida recuperação e rápido processamento. Enquanto um pacote DIS é enviado periodicamente do nó raiz até o primeiro nó parente ser selecionado, o *trickle timer* é usado para agendar as transmissões DIO. Esse tempo, permite que os intervalos DIOs aumente exponencialmente quando as condições de redes são estáveis, e diminui rapidamente ao mínimo, quando ocorre mudanças notáveis nas condições de rede.

O tempo periódico t do *trickle* é limitado pelos intervalos $[I_{\min}, I_{\max}]$, onde $I_{\min}$ é o menor valor do intervalo definido em milissegundos em um valor de base 2 e $I_{\max}$ definido como $I_{\min}$ multiplicado por 2 elevado a capacidade de $I_{\min}$ tem de dobrar seu valor. A formula pode ser vista na equação (1)

$$I_{\max} = I_{\min} * 2^{I_{\text{dobrar}}} \quad (1)$$

O algoritmo *trickle* também é capaz de atualizar a topologia global em um pequeno período de tempo. Um nó que detecta uma inconsistência em uma mensagem DIO, altera o t para o valor mínimo e atualiza a arvore. Se o DODAG continuar consistente, t é dobrado toda vez que ocorre uma transmissão DIO, ate atingir $I_{\max}$.

Quando a rede é estável, o algoritmo *trickle timer* gradualmente converge para o intervalo máximo. Em mobilidade, esse largo intervalo resulta em uma baixa resposta da rede. Quando uma inconsistência na rede é detectada, o período do DIO de todos os nós na rede decresce exponencialmente, afetando o tempo de processamento.

Protocolo ND (*Neighbor Discovery*): O RPL pode utilizar a técnica de descobrimento de vizinhos do IPv6 para detectar mudanças de ambiente. O protocolo ND permite que os nós detectem vizinhos que não são alcançáveis e que descubram novos vizinhos. Esse protocolo utiliza quatro mensagens de controles ICMPv6:

- Solicitação de vizinho (*Neighbor Solicitation*): determina o link da camada de endereço de um vizinho, e verifica se o vizinho ainda esta alcançável;
- Sinalização de vizinho (*Neighbor Advertisement*): responde à mensagem de solicitação do vizinho e também é enviado periodicamente para anunciar mudanças no link;

- Solicitação de rota (*Router Solicitation*): requer de um nó hospedeiro (móvel) informações até o nó raiz;
- Sinalização de rota (*Router Advertisement*): Uma rota envia periodicamente e como resposta à mensagem de solicitação de rota, informações do link e os parâmetros da internet. [11]

A construção da rede do protocolo CTP é parecida com a do RPL. O CTP estende o uso do algoritmo *trickle* enviando mensagens de controle em uma taxa que depende de quão dinâmica a rede é. Em suma, quando uma rota esta vazia, um conjunto de nós na rede sinaliza eles mesmo como nós raízes da rede. Posteriormente, os nós formam um conjunto de rotas ate estes nós. No CTP cada nó seleciona um parente como próximo salto, e este parente esta mais próximo do nó raiz, do que o próprio nó. [12]

Para fazer a manutenção da topologia, o algoritmo *trickle* aumenta a frequência de sinalização quando uma inconsistência é detectada, e decresce exponencialmente quando a topologia esta estabilizada. Este método, assegura uma resposta rápida às mudanças e minimiza o processamento de controle de trafico. Uma parte importante do algoritmo é a detecção de inconsistências e o CTP utiliza pacotes de informação para detectar estas inconsistências, o que garante que a detecção será feita em tempo certo, economizando recursos. [13]

Quando a topologia é criada, a rede disponibiliza o ETX de cada sensor, tendo isso como parâmetro, o nó móvel faz a escolha do próximo salto baseado no que obtiver o menor custo de ETX como seu próximo parente a ser enviado informações.

2.6. CONSIDERAÇÕES FINAIS

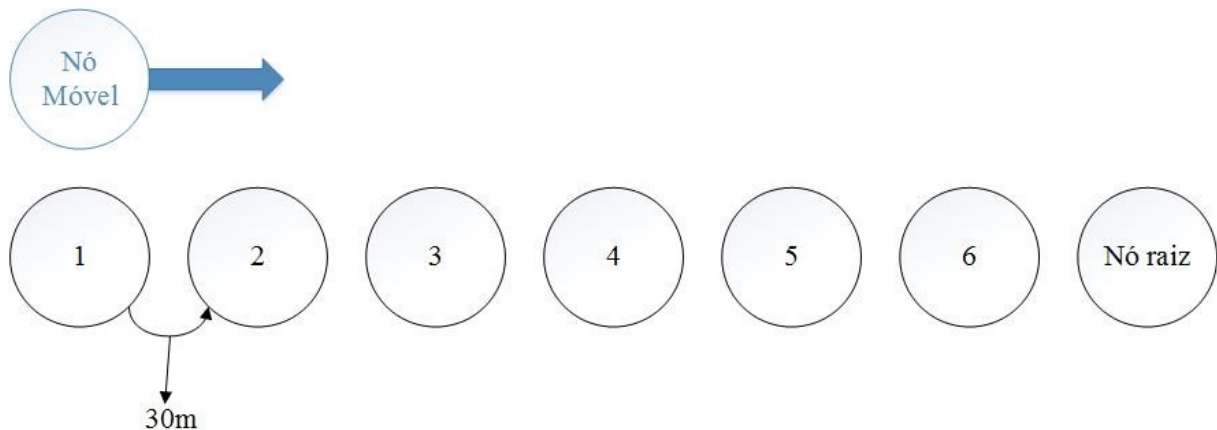
Neste capítulo, foi apresentado os mecanismos de funcionamento dos protocolos CTP e RPL, bem como quando é feita a manutenção e atualização de uma rota. No próximo capítulo será apresentado os cenários de testes, afim de verificar o desempenho dos protocolos citados, em um ambiente móvel.

CAPÍTULO 3

3.1. CENÁRIOS DE TESTES

Nos cenários a seguir será avaliado o desempenho quanto ao consumo de energia, a quantidade de pacotes entregues e também ao ETX, utilizando o simulador COOJA, que faz parte do sistema operacional de código aberto Contiki [6] para realizar os testes. Os testes serão realizados utilizando um ou dois nós móveis (NM) percorrendo 7 ou 10 nós fixos. No primeiro e segundo cenário os nós fixos(NFs) foram alinhados em intervalos de 30 metros, e o NM percorre os mesmos a uma velocidade de 1 metro por segundo até o nó raiz que se encontra na extremidade direita, como representado na Figura 3 e Figura 4. A velocidade de 1 metro por segundo foi escolhida para representar uma pessoa em posse de um dispositivo móvel se locomovendo normalmente. [1]

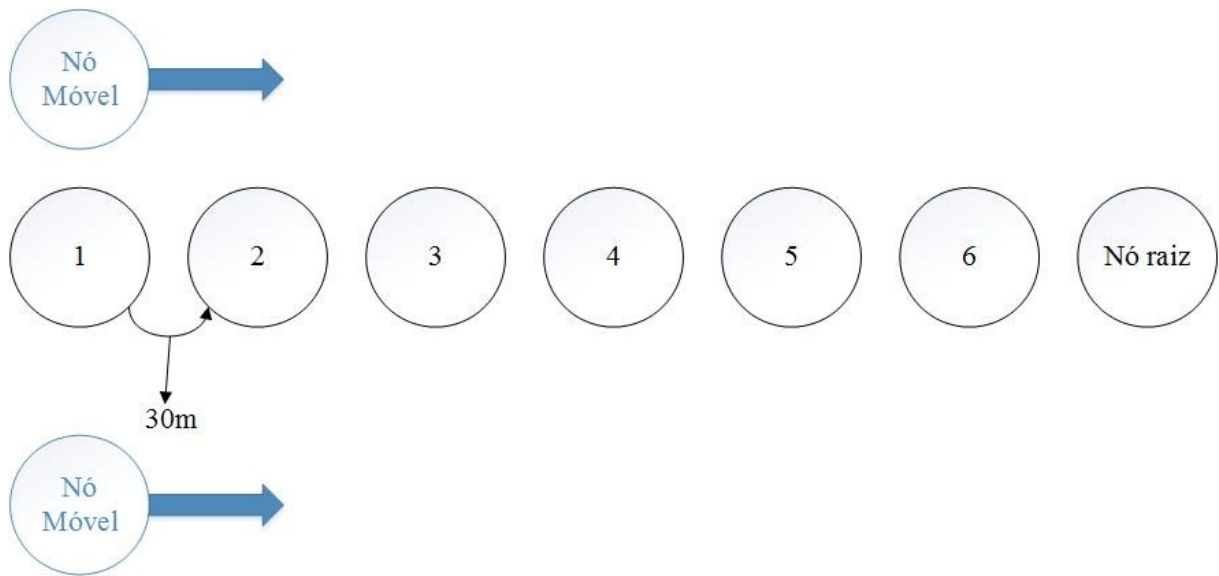
Figura 3 - Cenário de teste com um nó móvel



FONTE: Próprio Autor

Para o cenário com dois nós móveis, a distribuição dos nós ficou a característica da Figura 4.

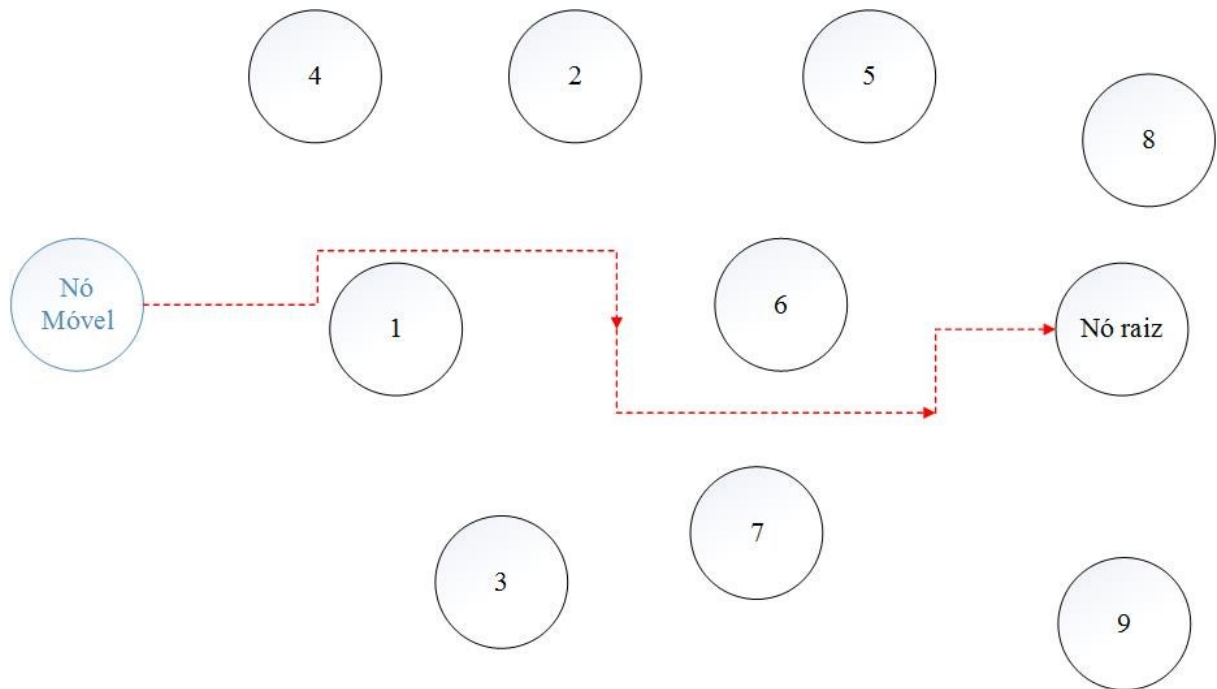
Figura 4 - Cenário de teste com dois nós móveis



FONTE: Próprio Autor

E por fim, o ultimo cenário de testes será realizado com 10 nós fixos contendo um nó raiz, e um móvel, espalhados aleatoriamente, sendo que o nó móvel percorrerá o caminho até o nó raiz, como mostrado na Figura 5.

Figura 5 - Cenário com nós aleatórios



FONTE: Próprio Autor

Os nós fixos foram configurados de modo que os mesmos não apresentem perdas de transmissão, ou seja, todos os pacotes transmitidos chegarão ao destino [1]. Tal configuração foi necessária para avaliar apenas o efeito que os nós móveis terão sobre a rede. Com isso, se retirado os nós móveis não haverá perda de pacotes e, assim, o ETX crescerá de modo constante a medida que os nós fixos estão mais distantes do nó raiz. Uma característica importante é que o nó raiz mantém o seu rádio ligado o tempo todo, o intuito por trás disto é que este nó não possua qualquer restrição de energia, podendo assim, estar conectado à uma rede de energia elétrica. [1]

Estes cenários foram escolhidos de modo a simular uma situação real, como por exemplo, uma indústria composta de 6 fornos enfileirados, enviando informações um ao outro, e o nó móvel, assemelha-se a uma pessoa caminhando entre estes fornos, trocando informações como quantidade consumida de energia, temperatura, tempo de funcionamento, etc. com os mesmos. Um outro cenário plausível, seria a de um hospital, cuja as pessoas acamadas estivessem sendo monitoradas por sensores, e o nó móvel seria uma pessoa que por ventura queira sair do quarto. Quando essa pessoa sai do quarto, e vai para outro recinto, ela vai escolhendo, durante o seu percurso, o quarto com quem ela troca informações. E este trabalho

consiste exatamente em saber a quantidade de energia, e de informações enviadas e perdidas, ocorrerá durante o percurso.

3.2. RECURSOS NECESSÁRIOS

Para a realização desta pesquisa, será necessário a utilização de um computador que contenha o VirtualBox ou semelhante, do sistema operacional baseado em Linux Contiki, além dos plug-ins necessários para simular o cenário de mobilidade.

3.3. CONSIDERAÇÕES FINAIS

Nesse capítulo foi mostrado os cenários de testes que serão realizados utilizando o simulador Contiki/Cooja e também os recursos necessários para realizar a simulação. A análise dos experimentos realizados será discutida no capítulo 4.

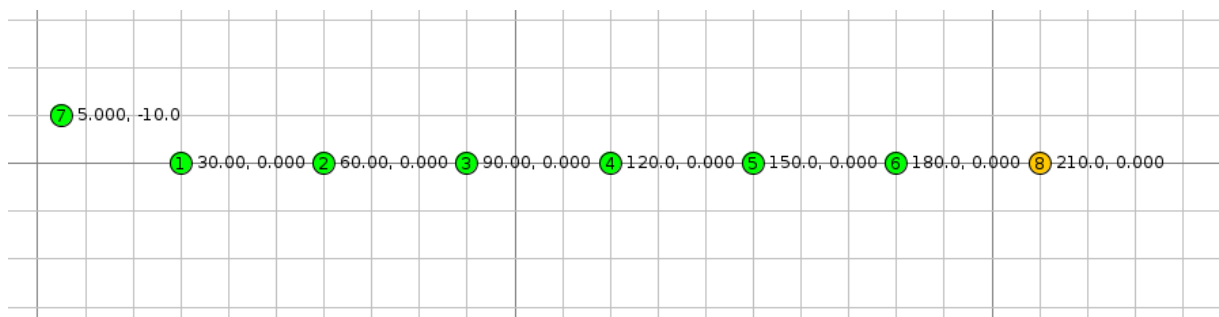
CAPÍTULO 4

4.1. ANÁLISE DOS EXPERIMENTOS

Neste capítulo será apresentado os resultados obtidos dos cenários descritos no capítulo 3. Para tal, foi utilizado o simulador Contiki/Cooja para fazer os experimentos.

4.1.1. Cenário 1

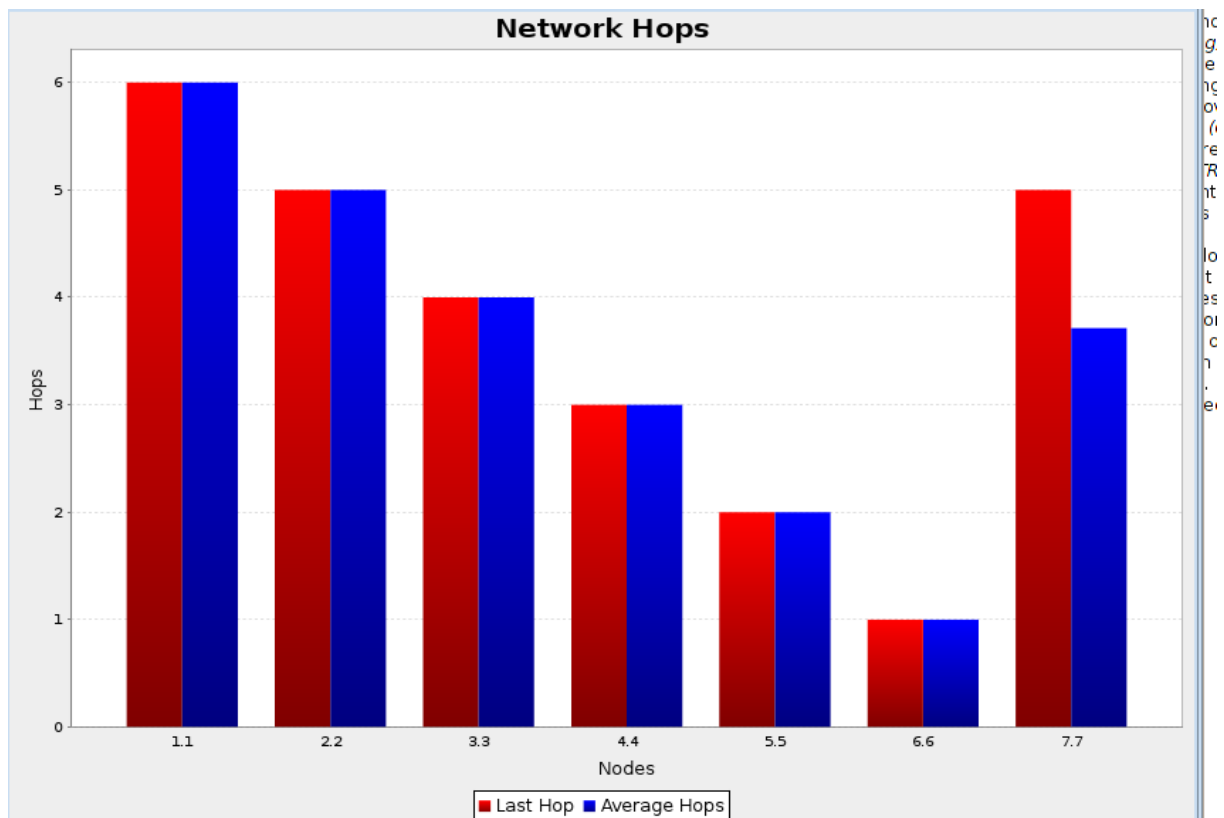
Figura 6 - Configuração 1º Cenário



FONTE: Contiki/Cooja

Conforme visto na Figura 6, os nós foram alinhados com uma distancia de 30 metros entre eles. O nó 8, representado pela cor amarela, é o nó raiz, e, o nó 7 é o nó móvel. Os números descritos na imagem são as posições dos nós em relação ao eixo x e y do plano cartesiano. O nó móvel teve seu inicio na posição (0, -10), com uma velocidade de 1 metro/s, percorreu o caminho ate a posição (180,-10). Vale ressaltar que o eixo y é fixo em -10, afim de obter uma melhor visualização do caminho percorrido. Este cenário foi testado por 15 minutos, e foram coletados dados quanto ao consumo de energia, taxa de pacote, ETX e também, a quantidade de saltos necessários para chegar ao nó raiz.

Gráfico 1 - Saltos 1º Cenário

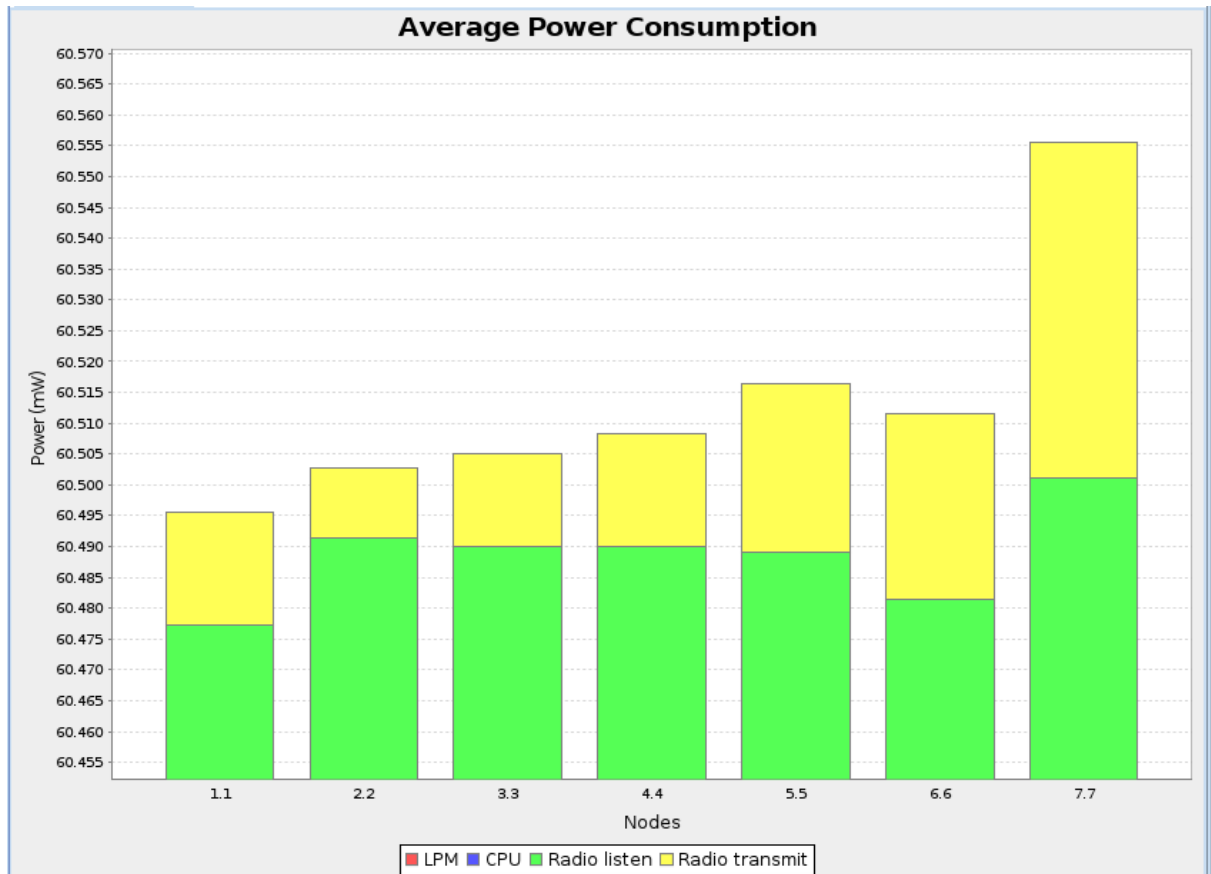


FONTE: Contiki/Cooja

O Gráfico 1 mostra a quantidade de saltos necessários para o pacote chegar ao nó raiz. A medida que os nós vão se aproximando do nó raiz, a quantidade de saltos diminui. O nó 1 necessita de 6 saltos, o nó 2 de 5 saltos, o nó 3 de 4 saltos, o nó 4 de 3 saltos, o nó 5 de 2 e o nó 6 de apenas 1 salto para chegar ao nó raiz. O nó 7, que é o nó móvel, necessita de 5 saltos, isso se da ao fato de que, o radio ou potencia do nó móvel, mesmo na posição inicial de (0,-10) consegue alcançar o nó 2, assim, a quantidade de saltos para o nó móvel alcançar o nó raiz começa do nó 2.

RPL

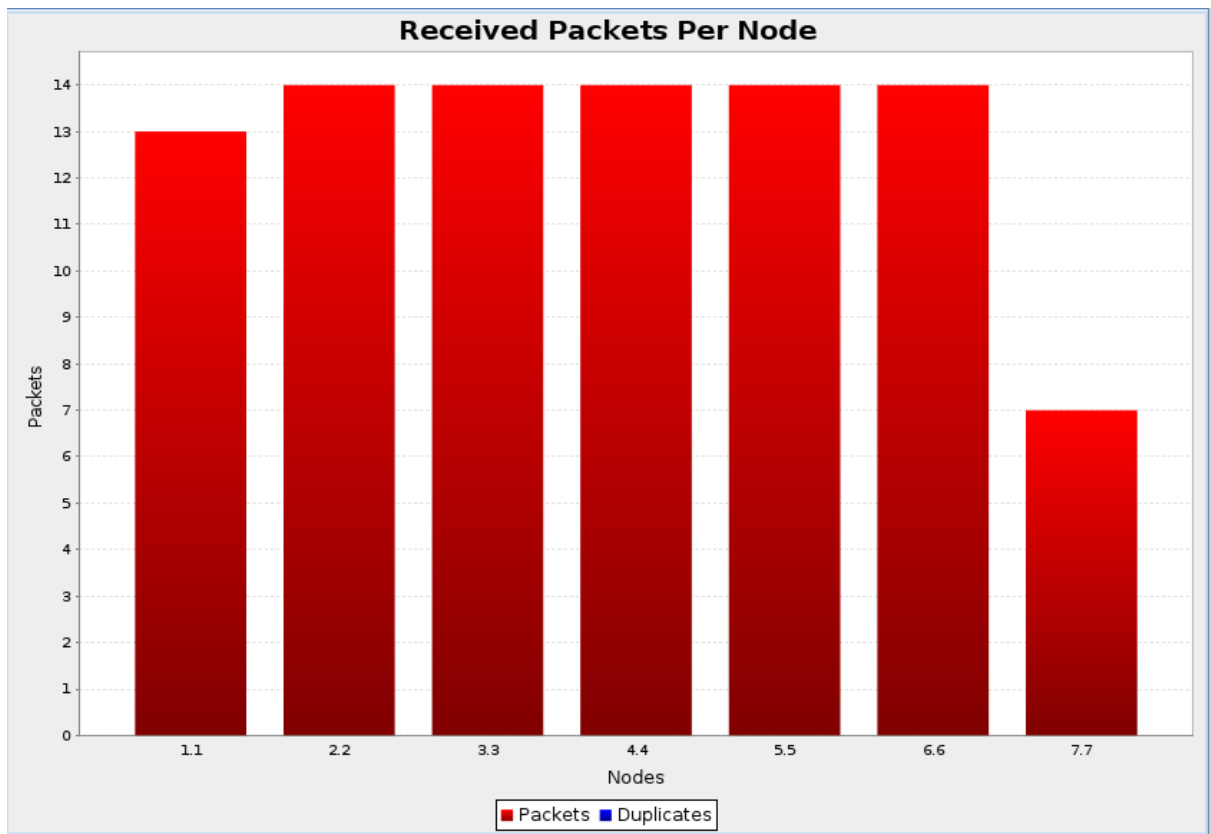
Gráfico 2 - Energia Consumida 1º Cenário RPL



FONTE: Contiki/Cooja

O Gráfico 2 mostra o consumo de energia de cada nó, levando em consideração o tempo que o radio esta “escutando” representado pelo verde, e o tempo que o radio esta “transmitindo”, representado pela cor amarela, percebe-se que o nó 7, que é o nó móvel, é o que mais consome energia.

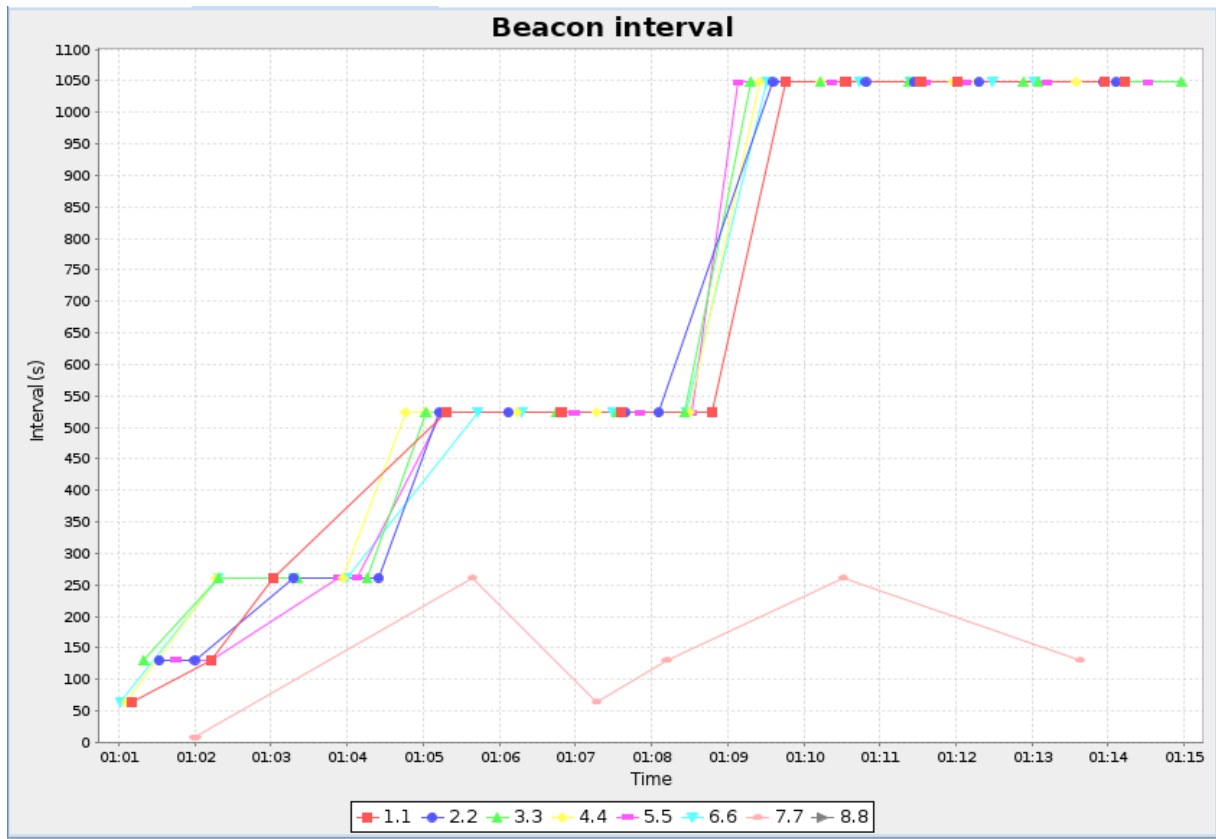
Gráfico 3 - Pacotes Recebidos 1º Cenário RPL



FONTE: Contiki/Cooja

Por meio do Gráfico 3, é possível perceber a quantidade de pacotes que o nó raiz recebeu de cada nó. Devido as constantes trocas de parentes, percebe-se que a menor quantidade de pacotes recebidos pelo nó raiz é oriunda do nó móvel.

Gráfico 4 - Intervalo de sinalização 1º Cenário RPL



FONTE: Contiki/Cooja

Um aspecto interessante é a análise do intervalo de sinalização descrito no Gráfico 4. De acordo com o algoritmo *trickle*, uma rede estável tende a convergir para um intervalo máximo de sinalização, ou seja, o tempo de um *Beacon* tende a ser maior. Quando a rede está no seu começo, percebe-se que o tempo de intervalo de sinalização é menor e, à medida que a rede vai ficando estável, o tempo passa a ficar constante. O nó móvel 7, é a linha “rosa” no gráfico. Como para o nó móvel o cenário fica variando, para mais perto ou para mais longe do nó raiz, caracteriza-se então um nó instável e, por conseguinte, seu tempo de sinalização também varia.

Quadro 1 - Quadro de Dados 1º Cenário RPL

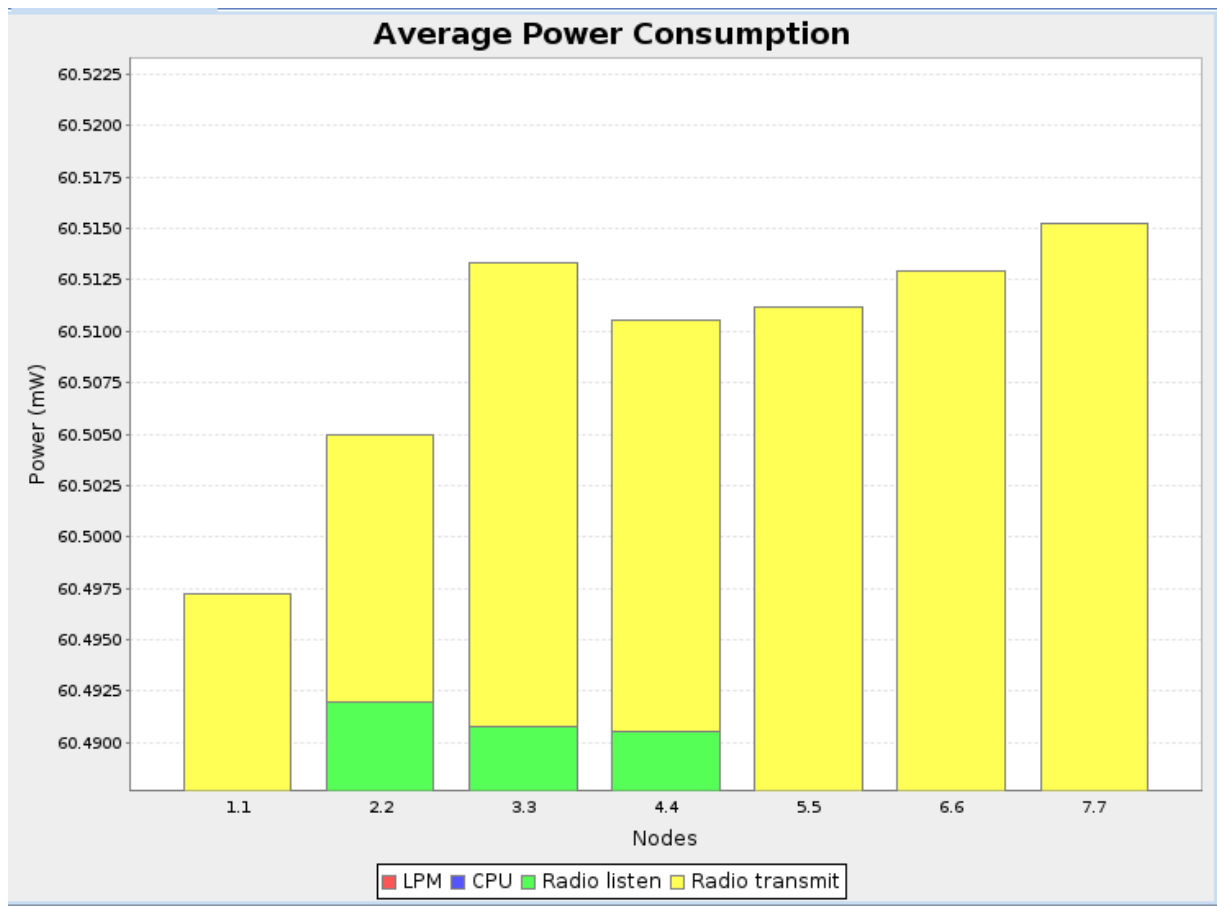
Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	13	0	1	6.000	975.846	48.385	0	11 min, 20 sec
2.2	14	0	0	5.000	773.071	40.241	0	10 min, 55 sec
3.3	14	0	0	4.000	641.929	32.098	0	11 min, 04 sec
4.4	14	0	0	3.000	513.214	24.027	0	11 min, 18 sec
5.5	14	0	0	2.000	384.000	16.000	0	10 min, 55 sec
6.6	14	0	0	1.000	256.000	8.000	0	10 min, 59 sec
7.7	7	0	6	3.714	846.000	29.714	2	2 min, 03 sec
8.8	0	0	0	0.000	0.000	0.000	0	
Avg	12.857	0.000	1.000	3.531	627.151	28.352	0.286	9 min, 48 sec

FONTE: Contiki/Cooja

O Quadro 1 mostra os dados do cenário 1. A menor quantidade de pacotes recebidos é oriunda do nó móvel, sendo recebidos 7 pacotes. Em geral, a media de pacotes recebidos para este cenário foi de 12,8 pacotes durante os 15 minutos decorridos do experimento. Nesta tabela, é perceptível também o ETX de cada nó, sendo que o nó mais afastado do nó raiz, o nó 1, é o que possui maior ETX, e a medida que vai se aproximando do nó raiz, o ETX dos nós vão diminuindo. O nó móvel, que é instável, possui uma ETX de 29.7, próximo a media total de 28.3.

CTP

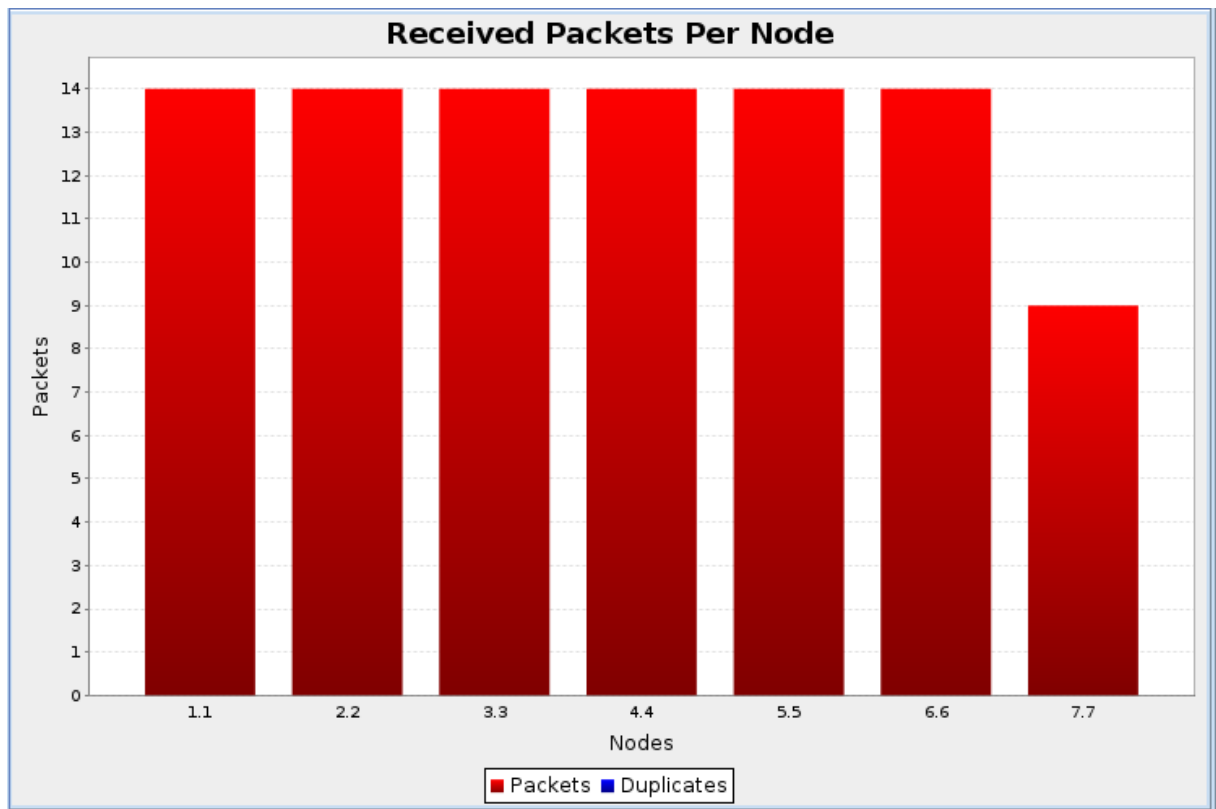
Gráfico 5 - Energia Consumida 1º Cenário CTP



FONTE: Contiki/Cooja

Para a simulação do primeiro cenário utilizando o protocolo CTP, verificamos por meio do Gráfico 5 que o consumo de energia se dá um pouco maior em relação aos nós fixos, mas ainda assim, o nó móvel é o que apresenta maior consumo de energia.

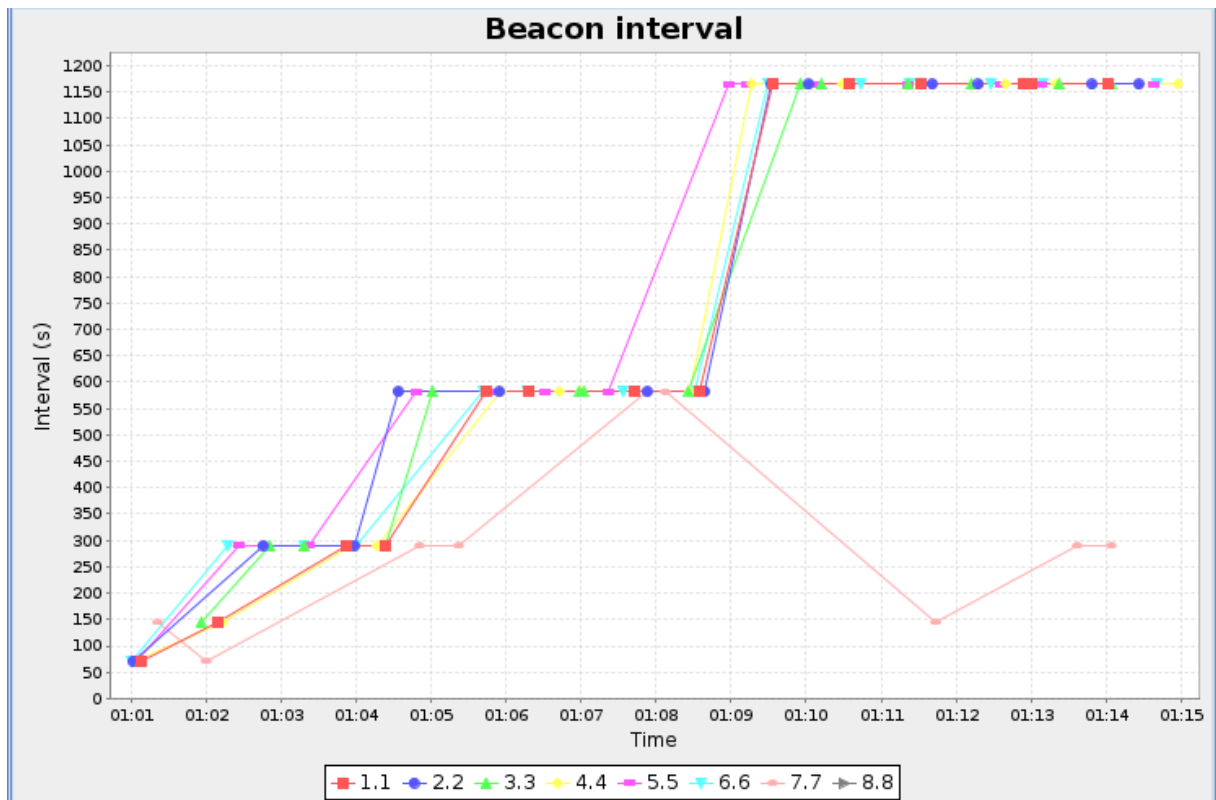
Gráfico 6 - Pacotes Recebidos 1º Cenário CTP



FONTE: Contiki/Cooja

O Gráfico 6 nos mostra a quantidade de pacotes recebidos no nó raiz, provindos do nó móvel. Verifica-se que houve uma recepção de 9 pacotes.

Gráfico 7 - Intervalo de Sinalização 1º Cenário CTP



FONTE: Contiki/Cooja

O intervalo de sinalização para o primeiro cenário do protocolo CTP pode ser vista no Gráfico 7. Verificamos que, o nó móvel representada pela cor “rosa”, apresenta uma forma mais contínua, ou seja, tende a manter-se mais na estabilidade.

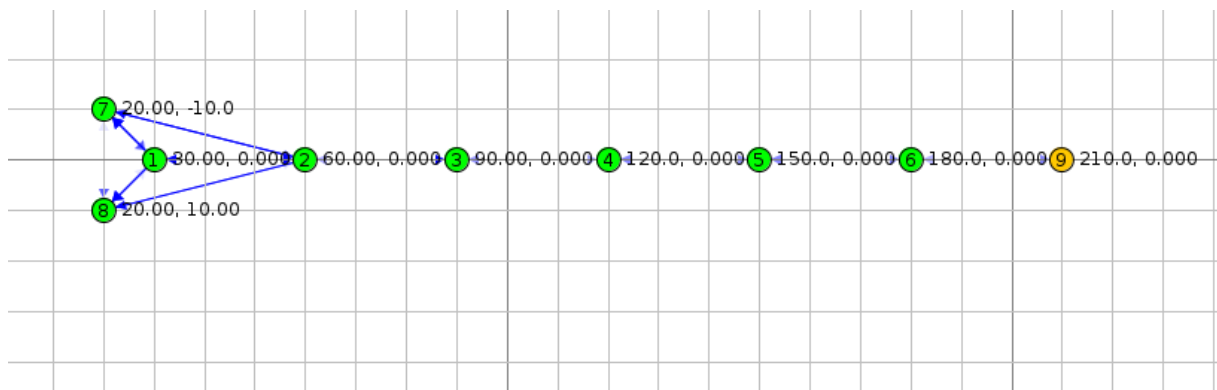
Quadro 2 - Quadro de Dados 1º Cenário CTP

Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	14	0	0	6.000	932.357	39.607	0	12 min, 02 sec
2.2	14	0	0	5.000	787.500	32.777	0	12 min, 33 sec
3.3	14	0	0	4.000	644.214	25.696	0	12 min, 18 sec
4.4	14	0	0	3.000	515.786	19.295	0	12 min, 02 sec
5.5	14	0	0	2.000	384.286	12.750	0	13 min, 15 sec
6.6	14	0	0	1.000	256.071	6.375	0	12 min, 13 sec
7.7	9	0	5	3.889	792.889	26.194	3	4 min, 58 sec
8.8	0	0	0	0.000	0.000	0.000	0	
Avg	13.286	0.000	0.714	3.556	616.158	23.242	0.429	11 min, 20 sec

Para finalizar o primeiro cenário do protocolo CTP, temos os dados representado pelo Quadro 2. Por meio da tabela, verificamos que o ETX do nó móvel é aproximadamente 26, e que dos 14 pacotes enviados oriundos do nó móvel, 9 foram recebidos e 5 perdidos.

4.1.2. Cenário 2

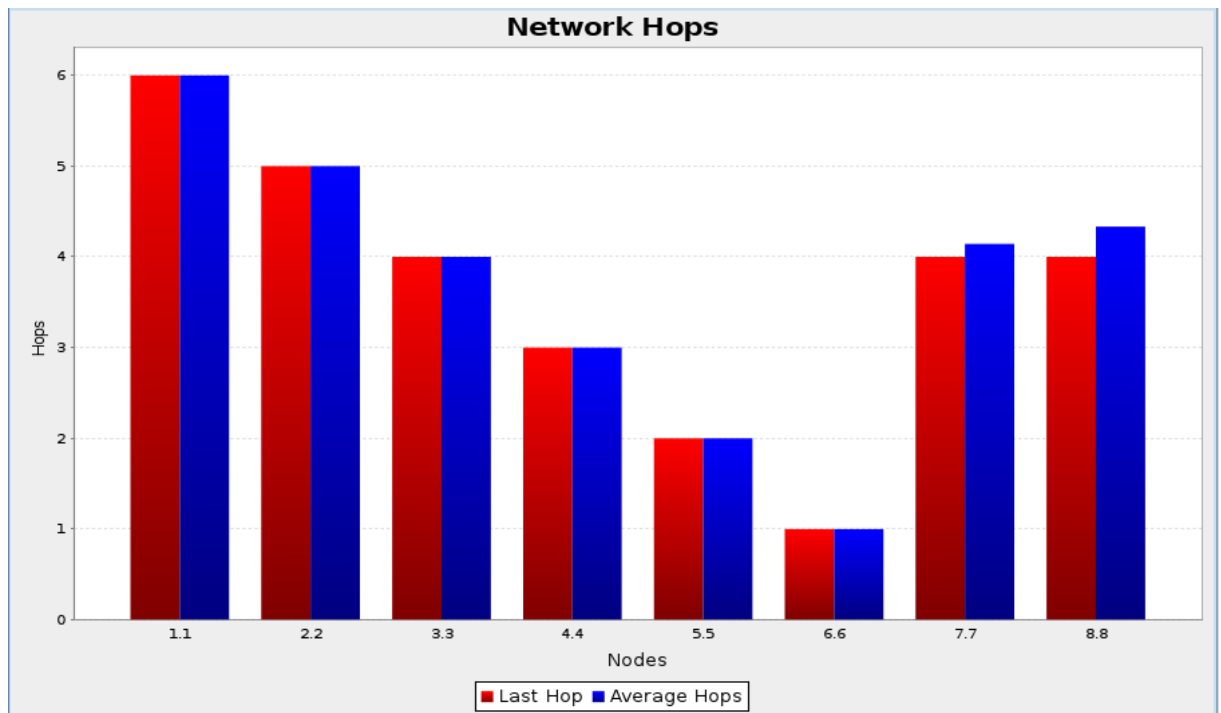
Figura 7 - Configuração 2º Cenário



FONTE: Contiki/Cooja

O segundo cenário do experimento foi montado conforme a Figura 7, onde o nó raiz é representado pela cor amarela e o numero 9 e, como neste cenário há dois nós moveis, são representados pelos números 7 e 8.

Gráfico 8 - Saltos 2º Cenário

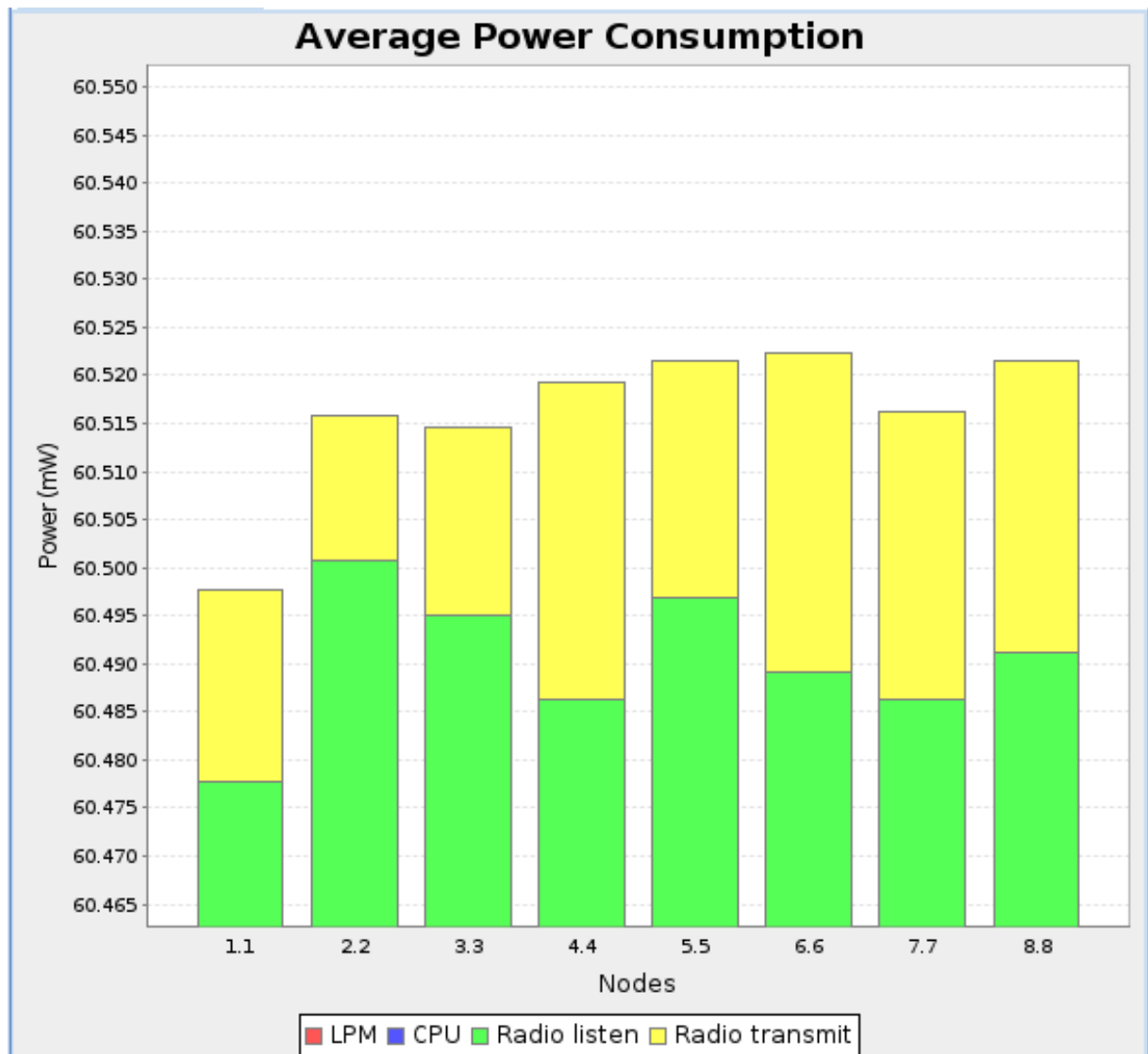


FONTE: Contiki/Cooja

No Gráfico 8, vimos a quantidade de saltos necessários para se alcançar o nó raiz. Na figura, percebe-se novamente que quanto mais próximo do nó raiz, menor a quantidade de saltos, e para os nós moveis, uma media de 4.2 saltos necessários para chegar ao nó raiz.

RPL

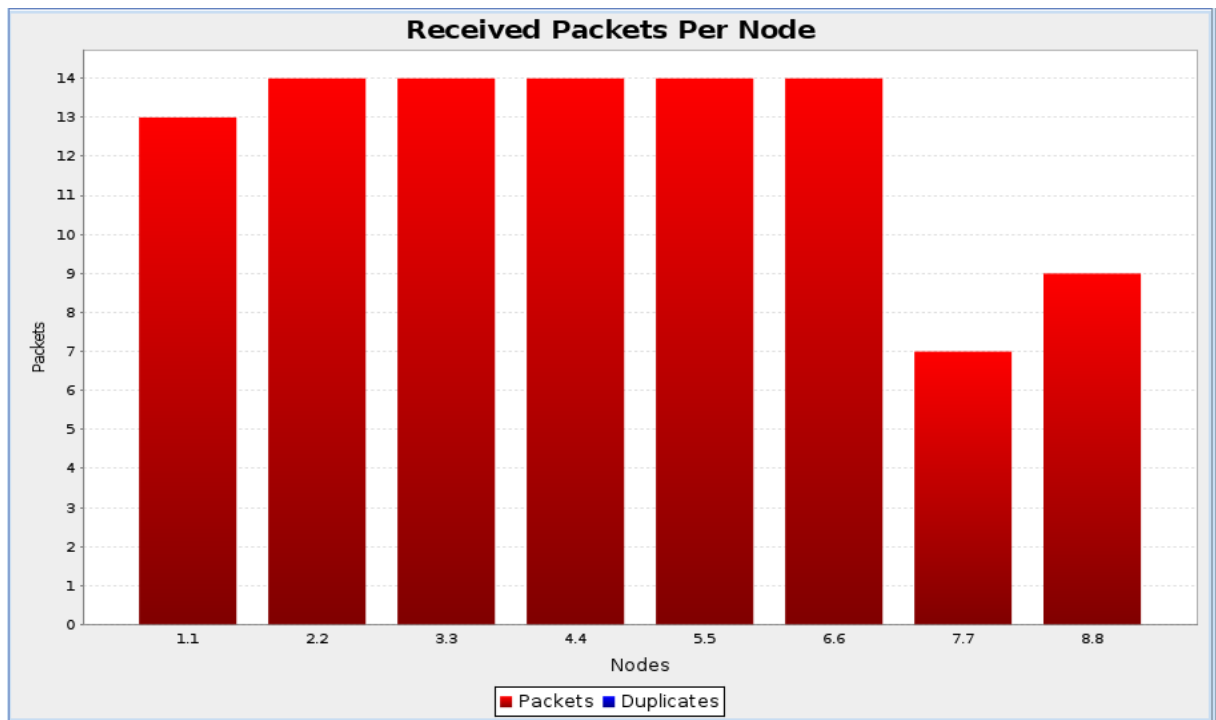
Gráfico 9 - Energia Consumida 2º Cenário RPL



FONTE: Contiki/Cooja

Para o cenário contendo dois nós móveis, percebe-se pelo Gráfico 9 que a energia consumida pelos nós móveis(7 e 8) se assemelha aos nós fixos, isso se da ao fato de que, os nos fixos tem de ficar com o “radio em escuta” mais tempo, elevando assim, o consumo de energia.

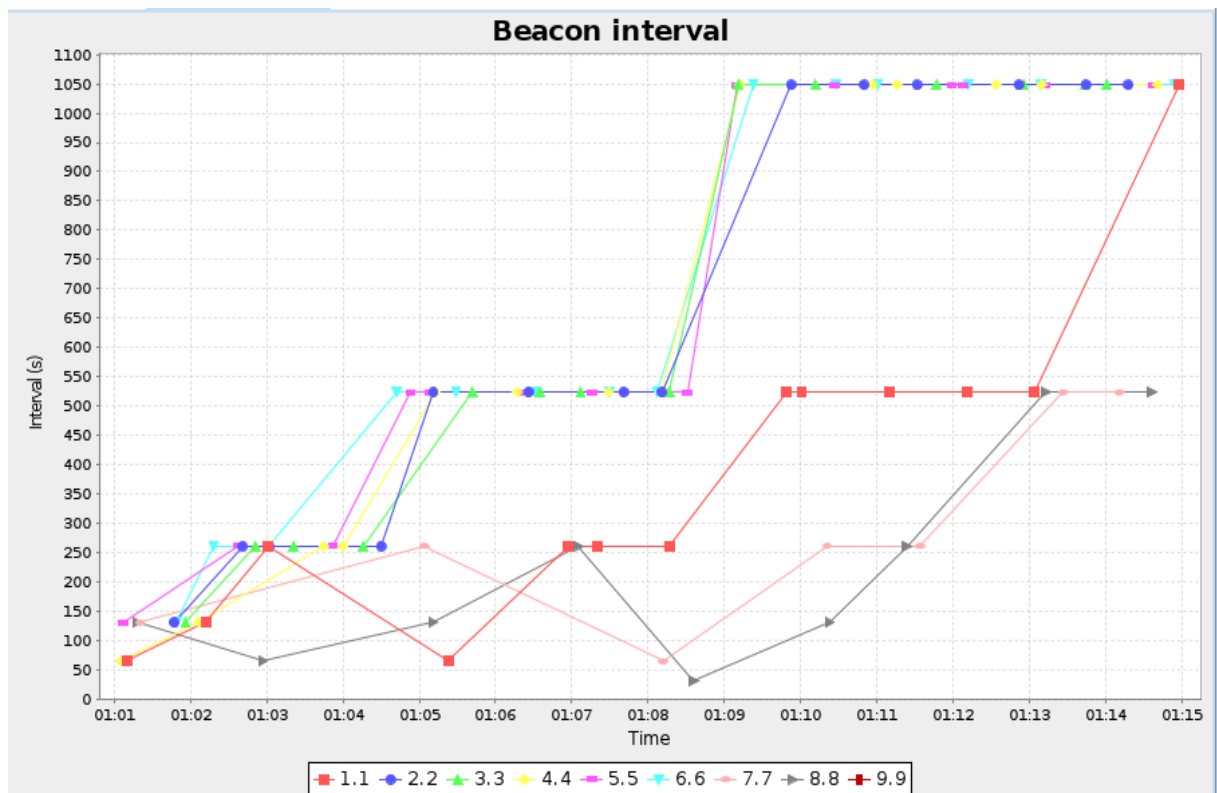
Gráfico 10 - Pacotes Recebidos 2º Cenário RPL



FONTE: Contiki/Cooja

No Gráfico 10 obtemos a quantidade de pacotes recebidos por nó. A menor quantidade de pacotes é oriunda dos nós moveis, tendo recebido o nó raiz 7 pacotes oriundos do nó 7, e 9 pacotes oriundos do nó 8.

Gráfico 11 - Intervalo de Sinalização 2º Cenário RPL



FONTE: Contiki/Cooja

O Gráfico 11 nos mostra o intervalo de sinalização para o 2º cenário. A cor “rosa” e “cinza” representam os nós móveis. Observa-se que o nó 1 demorou para se estabilizar, o que pode ser considerado intrínseco ao simulador, como por exemplo, demorou para “ligar” seu radio.

Quadro 3 - Quadro de Dados 2º Cenário RPL

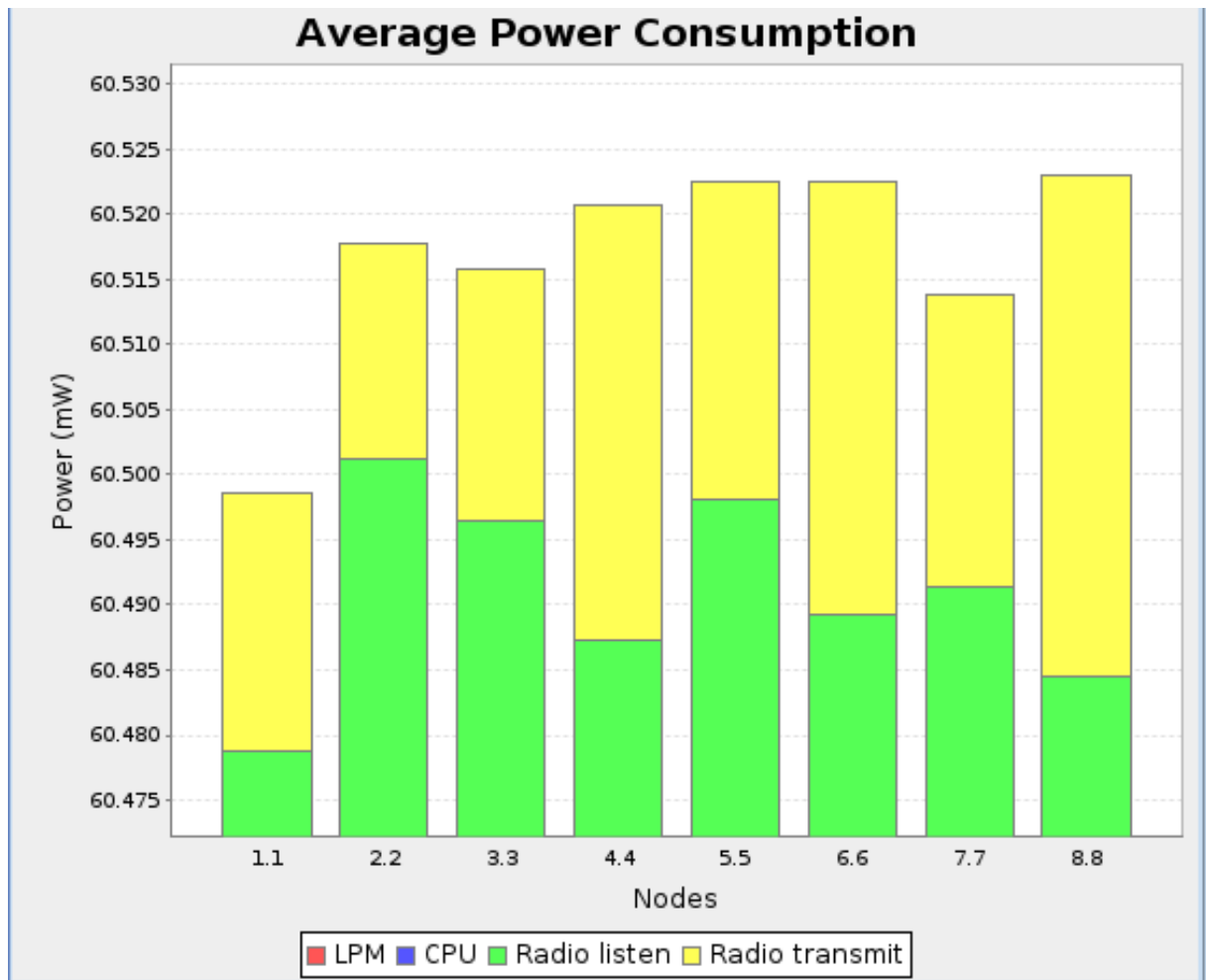
Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	13	0	1	6.000	919.769	48.990	0	6 min, 22 sec
2.2	14	0	0	5.000	777.000	40.223	0	11 min, 04 sec
3.3	14	0	0	4.000	641.071	32.054	0	11 min, 04 sec
4.4	14	0	0	3.000	512.643	24.009	0	10 min, 50 sec
5.5	14	0	0	2.000	384.429	16.000	0	11 min, 23 sec
6.6	14	0	0	1.000	256.000	8.000	0	11 min, 23 sec
7.7	7	0	7	4.143	982.000	34.018	2	4 min, 50 sec
8.8	9	0	5	4.333	879.111	35.347	3	3 min, 49 sec
9.9	0	0	0	0.000	0.000	0.000	0	
Avg	12.375	0.000	1.625	3.685	669.003	29.830	0.625	8 min, 50 sec

FONTE: Contiki/Cooja

Para finalizar o cenário 2 RPL temos o Quadro 3 demonstrando os dados dos resultados obtidos. Houve uma perda de 7 pacotes para o nó móvel 7. Já para o nó móvel 8, houve uma perda de 5 pacotes. O ETX são de 34 e 35, para os nós móveis 7 e 8, respectivamente. Essas altas perdas ocorrem devido ao cálculo do (*churn*), sendo que toda vez que se atinge um *churn* a rota é recalculada, o que ocasiona a perdas de pacotes.

CTP

Gráfico 12 - Energia Consumida 2º Cenário CTP

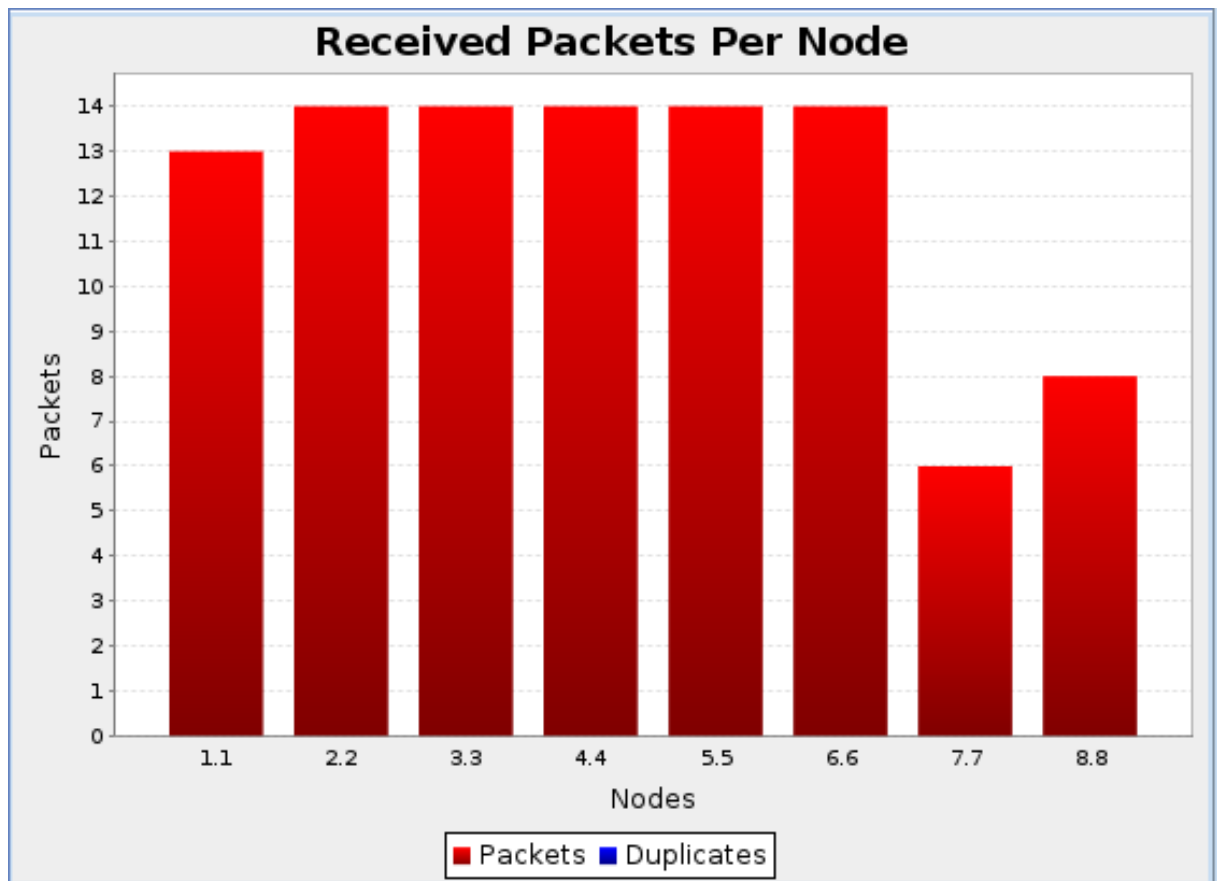


FONTE: Contiki/Cooja

O consumo de energia para os dois nós móveis é representado pelo Gráfico 12. Nota-se que o nó móvel 7 apresentou um gasto de energia menor em comparação aos outros nós, e, o nó móvel 8 apresentou um consumo de energia levemente superior aos nós fixos. Essa

representação do consumo de energia pode se dar ao fato das escolha dos nós como parente. Por exemplo, o nó 7 pode ter escolhido o nó 5 como seu parente mais próximo e permanecido com o mesmo por mais tempo.

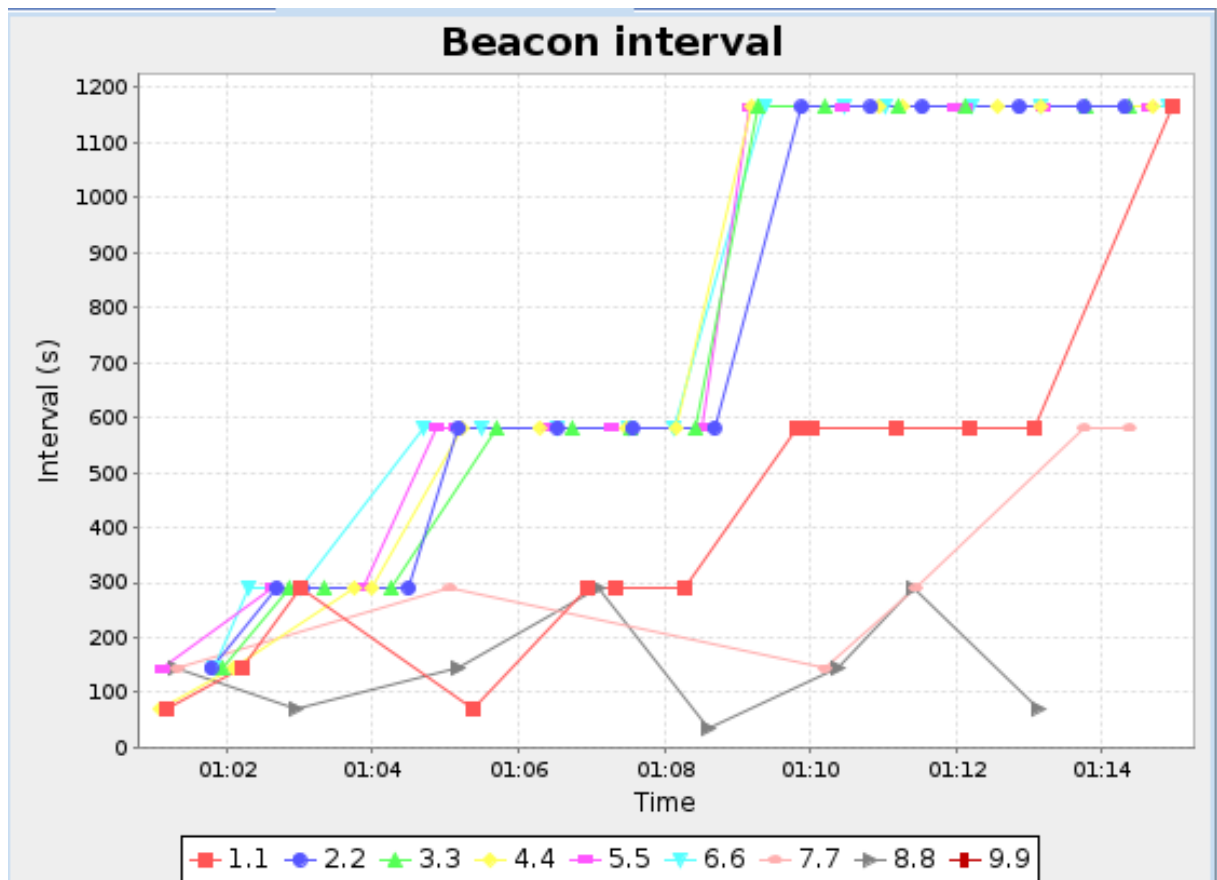
Gráfico 13 - Pacotes Recebidos 2º Cenário CTP



FONTE: Contiki/Cooja

O Gráfico 13 mostra a quantidade de pacotes recebidos oriundos dos dois nós móveis. Por meio dela verifica-se que o nó raiz recebeu 6 pacotes oriundos do nó móvel 7 e 8 pacotes oriundos do nó móvel 8.

Gráfico 14 - Intervalo de Sinalização 2º Cenário CTP



FONTE: Contiki/Cooja

O intervalo de sinalização para o segundo cenário do protocolo CTP é visto no Gráfico 14, onde a linha “cinza” e a linha “rosa” representam os nós móveis. Para este cenário, verifica-se que há bastante inconsistência, sendo necessário, a atualização do *trickle* constantemente.

Quadro 4 - Quadro de Dados 2º Cenário CTP

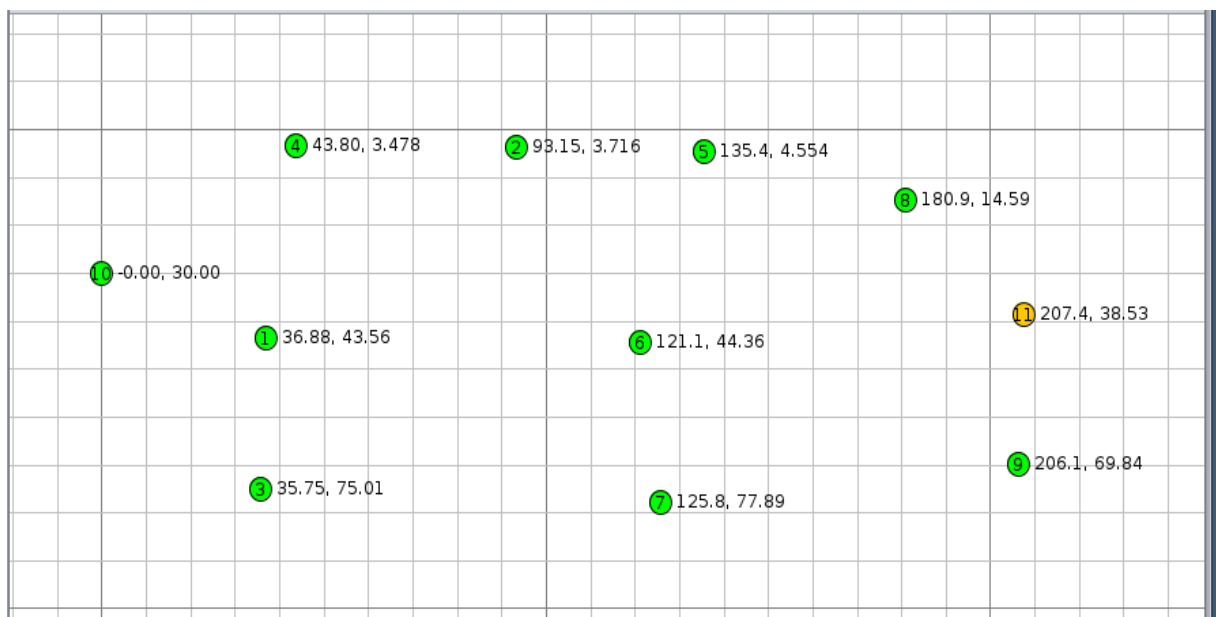
Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	13	0	1	6.000	914.692	38....	0	7 min, 05 sec
2.2	14	0	0	5.000	773.571	32....	0	12 min, 18 sec
3.3	14	0	0	4.000	640.714	25....	0	12 min, 18 sec
4.4	14	0	0	3.000	512.643	19....	0	12 min, 02 sec
5.5	14	0	0	2.000	384.071	12....	0	12 min, 39 sec
6.6	14	0	0	1.000	256.000	6.375	0	12 min, 39 sec
7.7	6	0	8	4.167	979.167	27....	2	5 min, 39 sec
8.8	8	0	5	4.625	868.375	30....	4	2 min, 29 sec
9.9	0	0	0	0.000	0.000	0.000	0	
Avg	12.125	0.000	1.750	3.724	666.154	24....	0.750	9 min, 38 sec

FONTE: Contiki/Cooja

Para finalizar o segundo cenário, a representação dos dados obtidos pela simulação do CTP é vista no Quadro 4 . Constata-se que para os nós móveis, houve uma perda de 8 pacotes oriundos do nó móvel 7, e de 5 pacotes oriundos do nó móvel 8. O ETX para os nós móveis 7 e 8, são 27 e 30, respectivamente.

4.1.3. Cenário 3

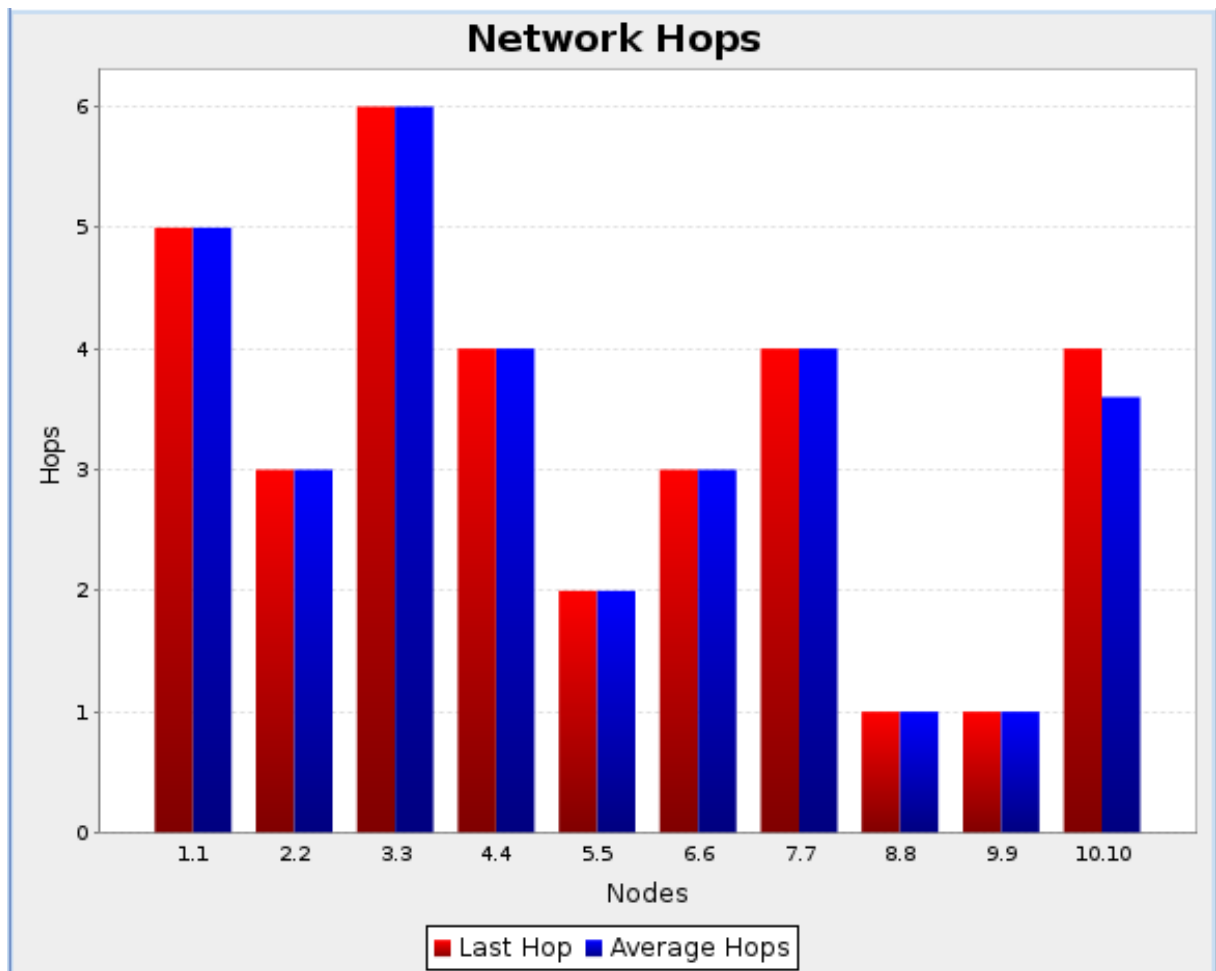
Figura 8 - Configuração 3º Cenário



FONTE: Contiki/Cooja

O ultimo cenário de teste foi formado de forma aleatória conforme descrito na Figura 8, onde o nó raiz é representado pelo numero 11 e de cor amarela, e o nó móvel é representado pelo numero 10. Para este cenário, o nó móvel percorrerá o caminho (0-80, 30) quando atingir essas coordenadas, seguirá pelo caminho (80, 30-60), e por fim até a posição (80-210, 60).

Gráfico 15 - Saltos 3º Cenário

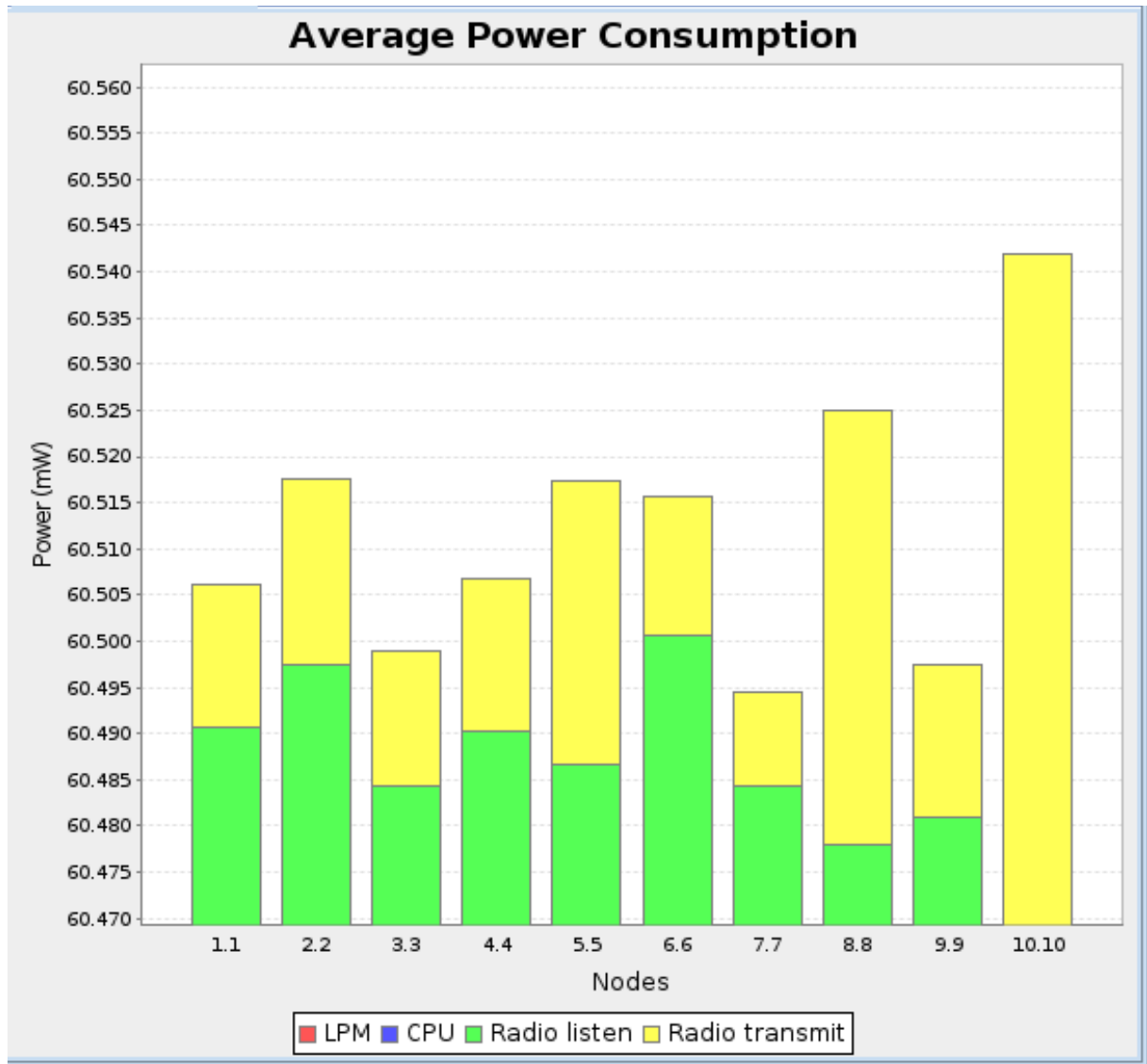


FONTE: Contiki/Cooja

Com o Gráfico 15, é possível ver a quantidade de saltos necessários para se chegar ao nó raiz. Tendo em vista que os nós 8 e 9 são os que estão mais próximos do nó raiz, estes precisam apenas de um salto. O nó móvel necessita de uma media de 3.6 saltos para se chegar ao nó raiz.

RPL

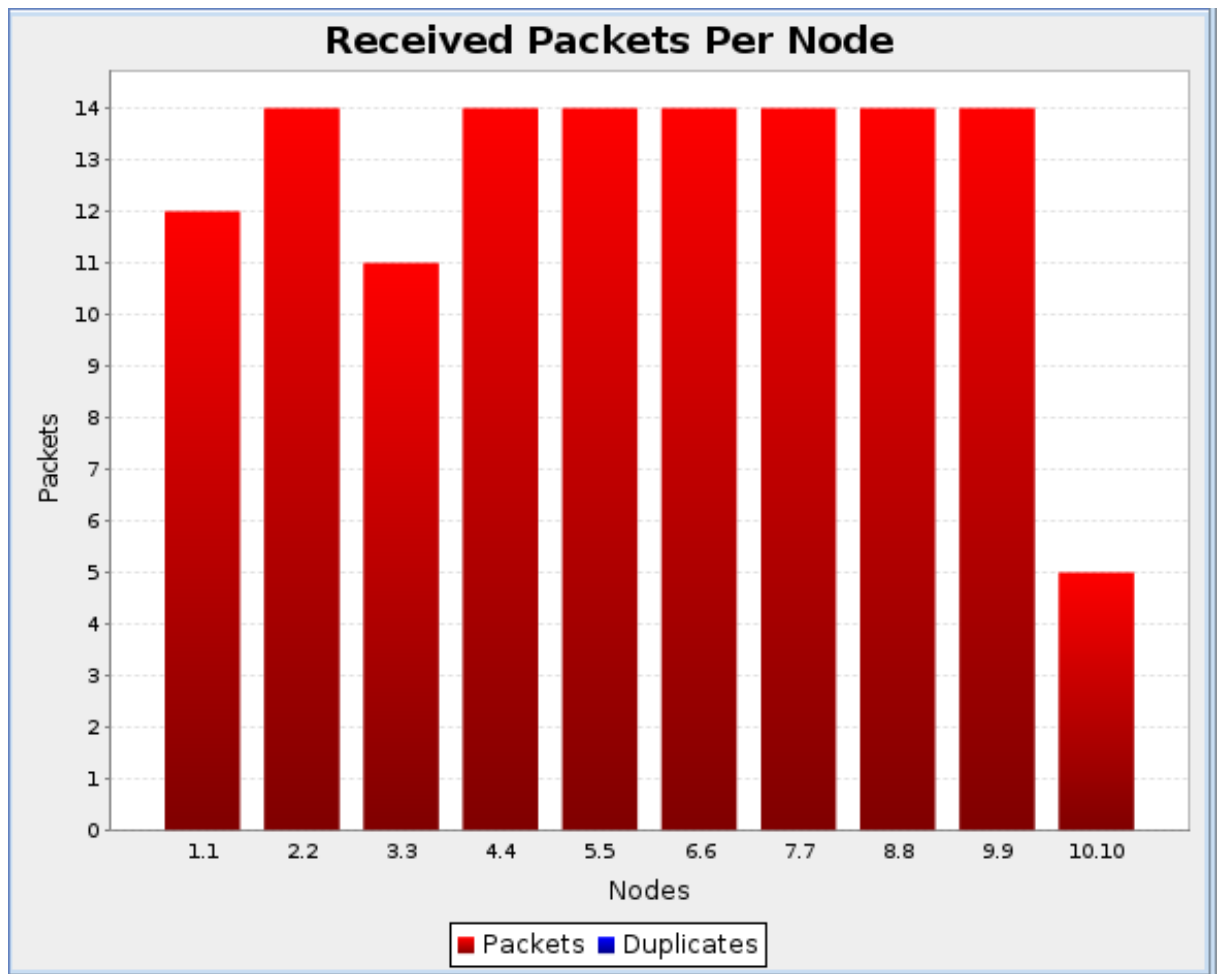
Gráfico 16 - Energia Consumida 3º Cenário RPL



FONTE: Contiki/Cooja

O consumo de energia do nó móvel é bem maior em relação aos outros nós. Isso se dá ao fato de que esse cenário é um pouco mais imprevisível do que os demais e, por conseguinte, é necessário enviar mais informações além de haver maiores trocas de parentescos. Como percebe-se pelo Gráfico 16, a maior parte do tempo o rádio do nó móvel (10) está em transmissão.

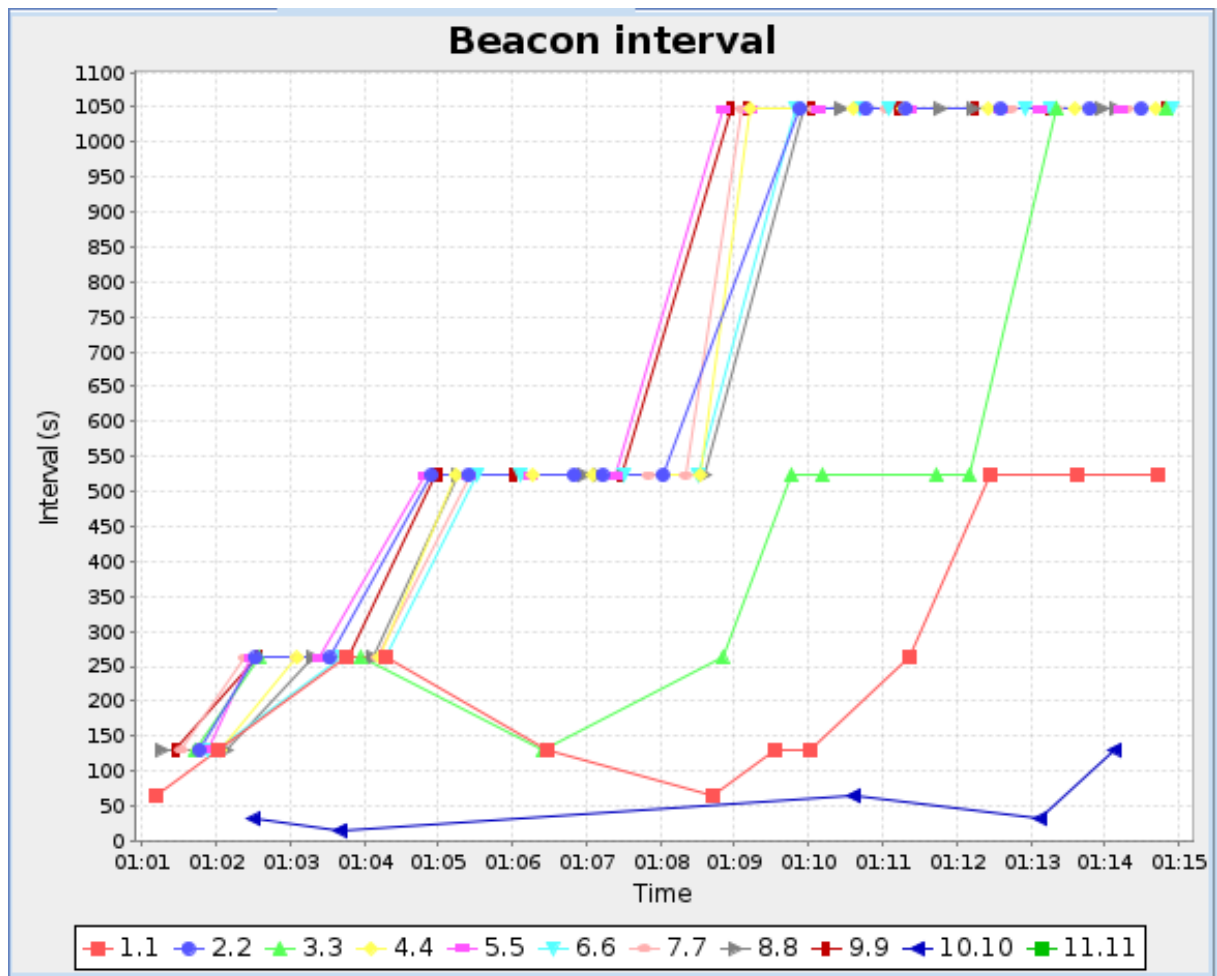
Gráfico 17 - Pacotes Recebidos 3º Cenário RPL



FONTE: Contiki/Cooja

O nó raiz recebeu 5 pacotes oriundos do nó móvel. Dado que os nós fixos 1 e 3 são os que estão mais distantes do nó raiz, e que o nó 3 tem como seu parente o nó 1, quando o nó 1 não consegue transmitir seus dados, os pacotes provindos do nó 3 também são perdidos, o que pode ser percebido pelo Gráfico 17.

Gráfico 18 - Intervalo de Sinalização 3º Cenário RPL



FONTE: Contiki/Cooja

O Gráfico 18 mostra o intervalo de sinalização do 3º cenário do experimento. Verifica-se que o menor intervalo, ou seja, a todo instante é enviado um *beacon*, provém do nó móvel, representado pela linha “Azul”. Verifica-se também que os nós fixos 1 e 3, linha vermelha e verde respectivamente, necessitaram um tempo maior para se estabilizarem, isso se da devido perda de pacotes que ambos sofreram.

Quadro 5 - Quadro de Dados 3º Cenário RPL

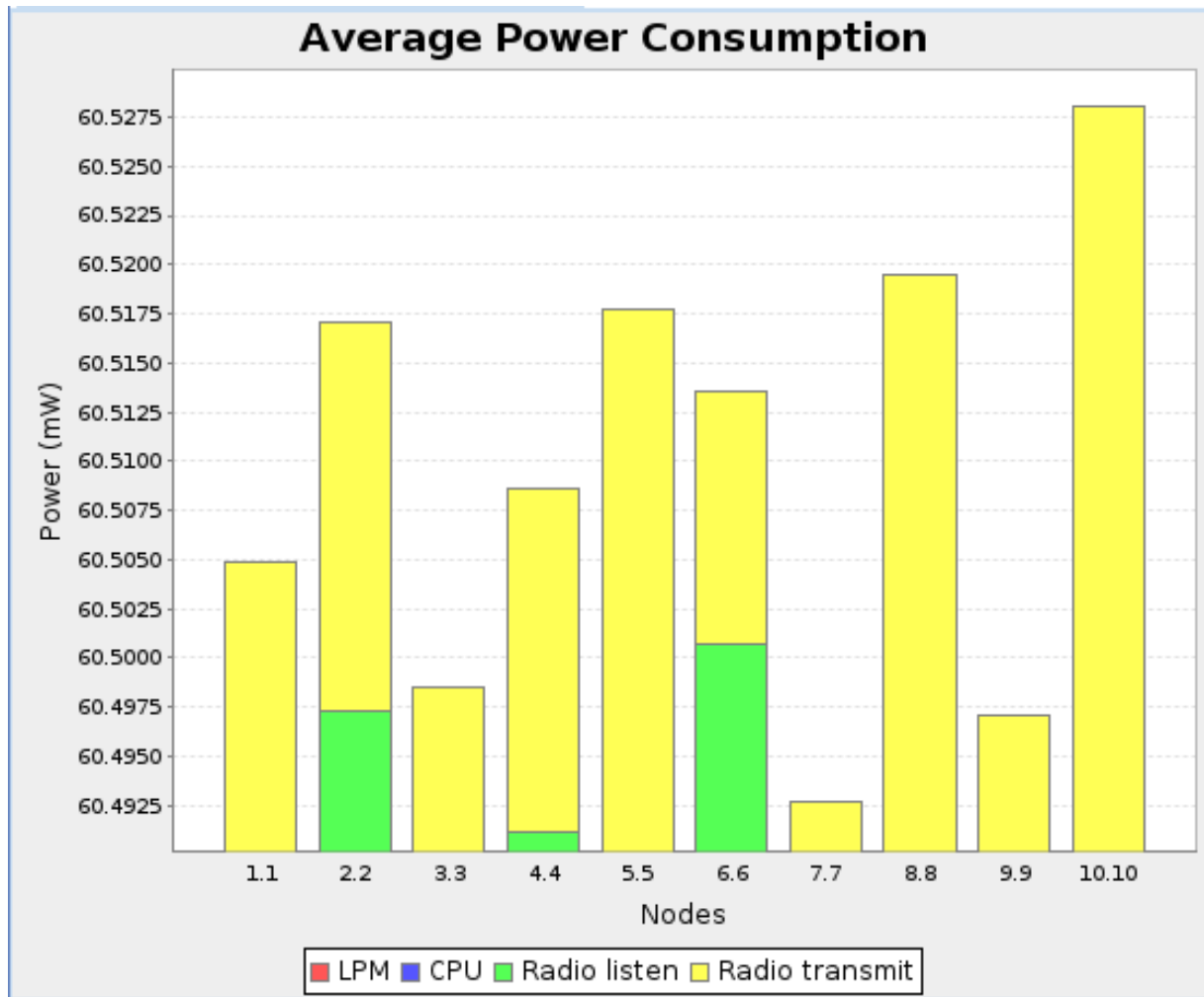
Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	12	0	2	5.000	808.000	42.313	0	4 min, 11 sec
2.2	14	0	0	3.000	526.786	24.670	0	11 min, 23 sec
3.3	11	0	3	6.000	944.636	50.625	0	7 min, 56 sec
4.4	14	0	0	4.000	669.929	33.661	0	10 min, 55 sec
5.5	14	0	0	2.000	388.286	16.241	0	12 min, 00 sec
6.6	14	0	0	3.000	546.500	24.750	0	10 min, 55 sec
7.7	14	0	0	4.000	691.214	34.679	0	11 min, 04 sec
8.8	14	0	0	1.000	260.429	8.000	0	10 min, 55 sec
9.9	14	0	0	1.000	260.714	8.000	0	12 min, 00 sec
10.10	5	0	8	3.600	841.200	32.325	3	0 min, 55 sec
11.11	0	0	0	0.000	0.000	0.000	0	
Avg	12.600	0.000	1.300	3.260	593.769	27.526	0.300	9 min, 13 sec

FONTE: Contiki/Cooja

Para finalizar o cenário de testes, verificamos que com o aumento da quantidade de nós, sendo estes espaçados de forma não uniformes, a quantidade de pacotes perdidos foi maior do que a quantidade de pacotes recebidos, como visto no Quadro 5, sendo perdidos 8 pacotes oriundos do nó móvel (10.10). o ETX do nó móvel foi de aproximadamente 32. O que vale ressaltar é o tempo de intervalo de sinalização, sendo necessário um *beacon* a cada 55 segundos para o nó móvel.

CTP

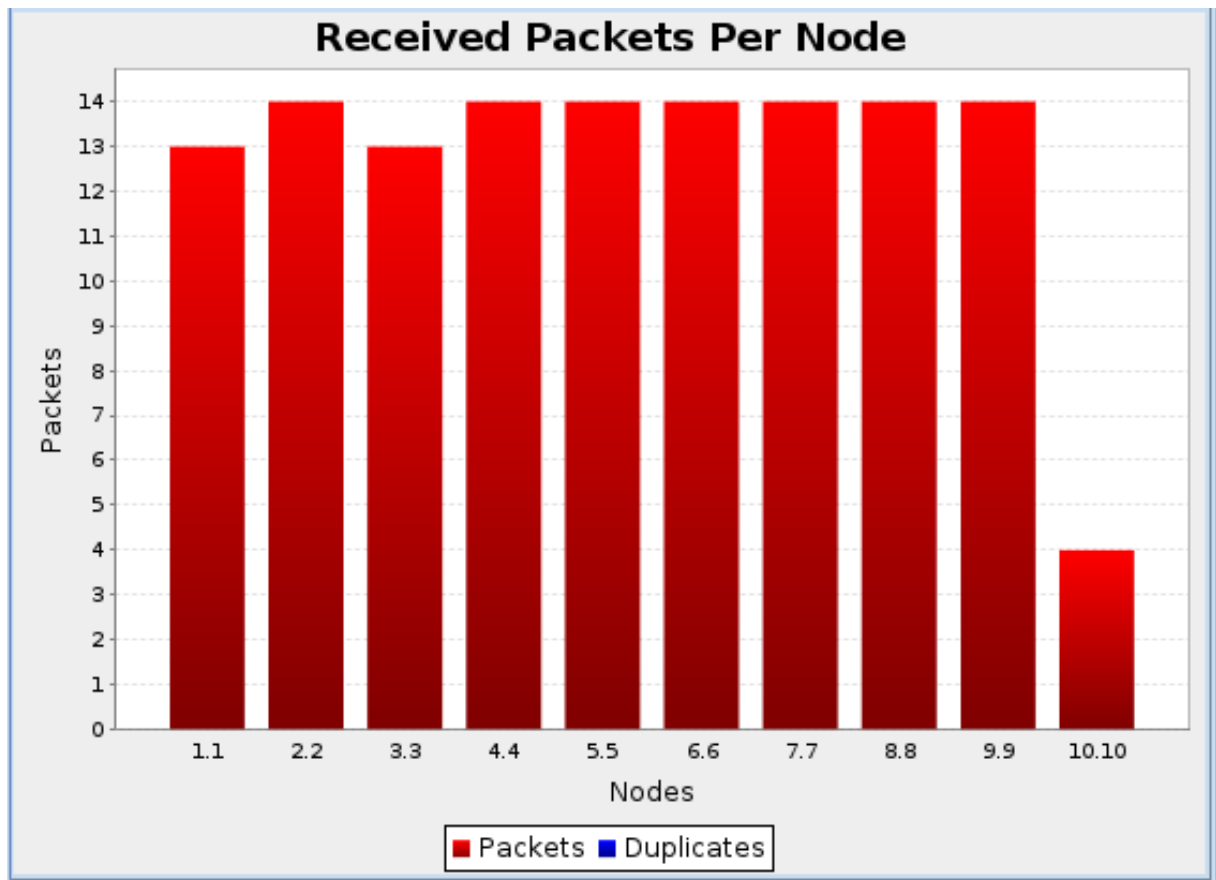
Gráfico 19 - Energia Consumida 3º Cenário CTP



FONTE: Contiki/Cooja

Com o Gráfico 19 temos o consumo de energia para a simulação do protocolo CTP, e, é possível verificar que o nó móvel é o que apresenta o maior consumo de energia além de estar a maior parte do tempo com seu radio em “transmissão”.

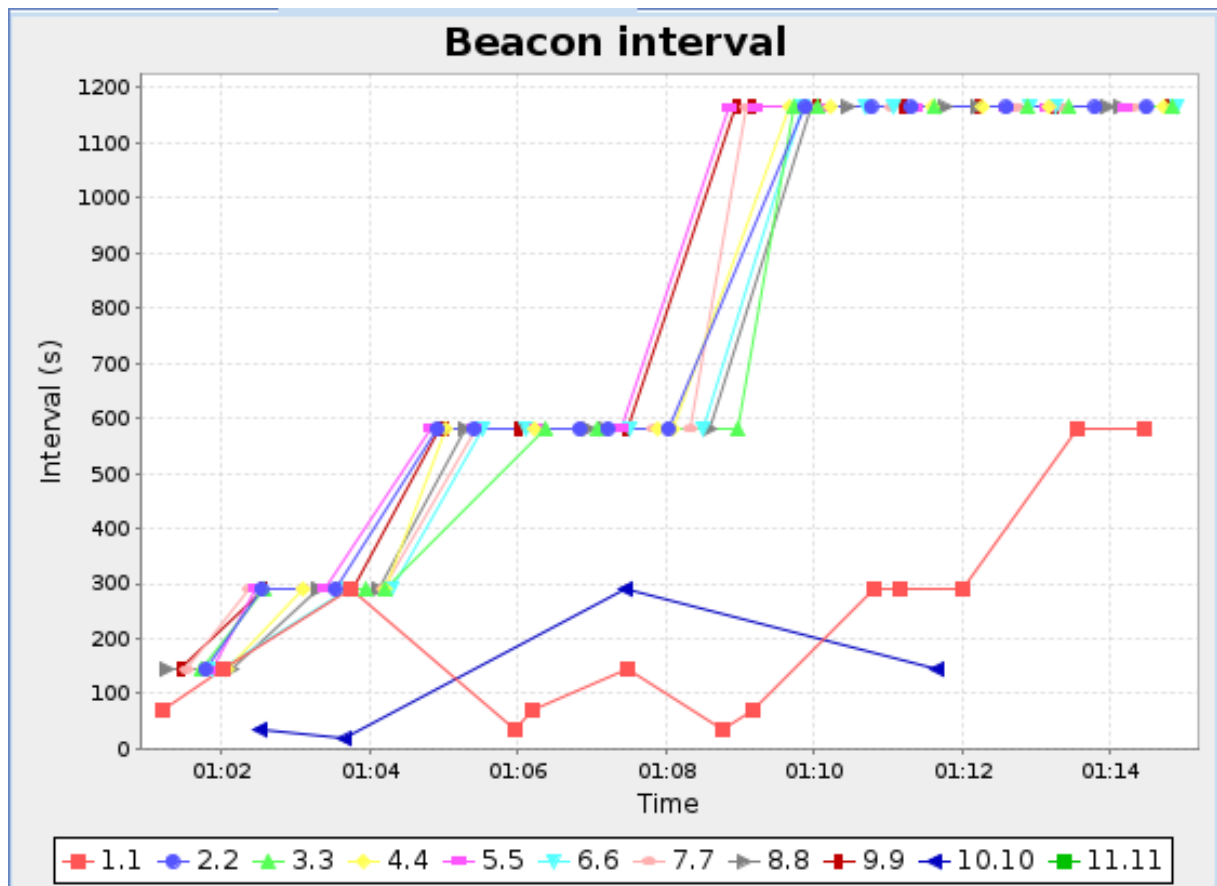
Gráfico 20 - Pacotes Recebidos 3º Cenário CTP



FONTE: Contiki/Cooja

A quantidade de pacotes recebidos pelo nó raiz, oriundos do nó móvel foi de 4 pacotes, como visto no Gráfico 20.

Gráfico 21 - Intervalo de Sinalização 3º Cenário CTP



FONTE: Contiki/Cooja

A linha “azul” representa o intervalo de sinalização do nó móvel. O Gráfico 21 mostra que há uma certa irregularidade. Para este cenário, verifica-se também que o nó fixo 1 demorou a se estabilizar.

Quadro 6 - Quadro de Dados 3º Cenário CTP

Node	Received	Dups	Lost	Hops	Rtmetric	ETX	Churn	Beacon Interval
1.1	13	0	1	5.000	806.615	33....	0	3 min, 43 sec
2.2	14	0	0	3.000	527.071	19....	0	12 min, 39 sec
3.3	13	0	1	6.000	945.692	40....	0	12 min, 30 sec
4.4	14	0	0	4.000	669.714	26....	0	12 min, 07 sec
5.5	14	0	0	2.000	388.500	12....	0	13 min, 20 sec
6.6	14	0	0	3.000	546.857	19....	0	12 min, 07 sec
7.7	14	0	0	4.000	692.357	27....	0	12 min, 18 sec
8.8	14	0	0	1.000	260.571	6.375	0	12 min, 07 sec
9.9	14	0	0	1.000	260.786	6.375	0	13 min, 20 sec
10.10	4	0	6	1.750	777.750	12....	1	2 min, 02 sec
11.11	0	0	0	0.000	0.000	0.000	0	
Avg	12.800	0.000	0.800	3.075	587.591	20....	0.100	10 min, 37 sec

FONTE: Contiki/Cooja

Para finalizar as simulações, temos o Quadro 6, que mostra os dados obtidos da simulação do protocolo CTP. Por meio do quadro, verifica-se que o nó móvel enviou 10 pacotes, dos quais 4 foram recebidos e 6 pacotes foram perdidos, e, apresenta um ETX de 12.

CAPÍTULO 5

5.1. CONSIDERAÇÕES FINAIS

Na primeira parte deste trabalho foi feita uma introdução sobre à Internet Das Coisas e os protocolos necessários para fazer o roteamento dos diversos dispositivos que poderão ser inseridos na internet.

No capítulo 2 foi explicado como é feito o roteamento dos protocolos CTP e RPL, sendo que o primeiro, o roteamento é feito utilizando somente pelo calculo do ETX, e o segundo, utiliza-se do ETX para calculo de rota, entretanto, possui uma métrica mais complexa de manutenção de rota, que é a criação de um DODAG. Também foi explicado como é feita a escolha de um nó, sua manutenção e atualização, através do protocolo ND, usado pelo RPL, ou pelo algoritmo *trickle*, utilizado por ambos os protocolos.

No capítulo 3 foi explicado os cenários de testes, sendo que os testes serão realizados em três partes, uma com seis nós fixos, um nó raiz e um nó móvel. O outro cenário será realizado com seis nós fixos, um nó raiz e dois nós móveis e, por último, será utilizado 9 nós fixos, um nó raiz e um nó móvel dispostos aleatoriamente. Estes cenários determinará o desempenho de ambos os protocolos em cenário de mobilidade.

No capítulo 4 foi feita a análise dos resultados obtidos com o protocolo RPL e CTP nos três cenários descritos no capítulo 3. Com os resultados obtidos verifica-se que os nós móveis apresentam um menor rendimento em comparação com os nós fixos em ambos os cenários, além de possuírem um maior consumo de energia e também, possuem uma maior taxa de pacotes perdidos devido as inconsistências encontradas durante o percurso. Vale ressaltar que, em um cenário real, como em uma casa com sensores, haveria porta, paredes, objetos diversos que poderiam acentuar ainda mais o desempenho dos sensores.

5.2. ESTUDOS FUTUROS

Este trabalho foi importante para compreender os protocolos RPL e CTP em um ambiente móvel. No que se diz respeito aos sensores fixos, há bastante estudos e até uma homologação do protocolo RPL como um dos mais importantes para redes de sensores sem fio

no IETF, entretanto, quando se trata de nó móvel ainda há muito estudo a ser feito. Para um trabalho futuro, poderia se fazer análises de como melhorar o desempenho, como a diminuição da taxa de pacotes perdidos, em ambientes inconsistentes ou móveis.

BIBLIOGRAFIA

- [1] J. H. K. Jeferson Rodrigues Cotrim, Avaliação de Desempenho do Protocolo RPL em Ambientes com Mobilidade, SANTARÉM, PA: SBrT2016,, 2016.
- [2] E. T. Winter, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," Março 2012. [Online]. Available: <https://tools.ietf.org/html/rfc6550>.
- [3] O. Iova, Standards Optimization and Network Lifetime Maximization for Wireless Sensor Networks in the Internet of Things, Universite de Strasbourg, 2014.
- [4] T.-L. & D. C. N. & C. W. & T. A. & S. K. Nguyen, "Comparative performance study of RPL in Wireless Sensor Networks," em *Conference: Communications and Vehicular Technology in the Benelux (SCVT)*, Benelux, 2012.
- [5] M. A. Sangeeta Vhatkar, Design Issues, Characteristics and Challenges in Routing Protocols for Wireless Sensor Networks, International Conference and Workshop on Emerging Trends in Technology 2013, 2013.
- [6] D. P. C. P. M. K. D. M. B. P. Bijal Chawla, Performance Analysis of Collection Tree Protocol in Mobile Environment, International Journal of Engineering Research and Applications (IJERA), 2013.
- [7] "Contiki," [Online]. Available: www.contiki-os.org.
- [8] F. Österlind, A Sensor Network Simulator for the Contiki OS, SICS Technical Report, 2006.
- [9] R. L. S. S. H. Dixit Sharma, Enhancing Collection Tree Protocol for Mobile Wireless Sensor Networks, International Workshop on Communications and Sensor Networks (ComSense-2013), 2013.
- [10] M. T. C. P. Fariborz Entezami, RCTP: An Enhanced Routing Protocol Based on Collection Tree Protocol, Wireless Multimedia & Networking (WMN) Research Group, Faculty of Science, Engineering and Computing (SEC), Kingston University London, London KT1 2EE, UK, 2015.
- [11] M. H. F. Ghazvini, Reliable Mobility Support in Low-Power Wireless Network, Doctoral Program in Electrical and computer Engineering, FEUP, 2015.

- [12] S. F. B. D. S. H. Abdelhamid Mellouk, Wired/Wireless Internet Communications, Paris, France: 12th International Conference, WWIC 2014, 2014.
- [13] R. F. K. J. P. L. Omprakash Gnawali, Collection Tree Protocol, ACM Sensys. 1-14. 10.1145/1644038.1644040., 2009.