

Mecanismo de Alocação de Recursos para LoRaWAN Ciente da Prioridade das Aplicações de IoT

Eduardo Lima¹, Nagib Matni¹, Jean Moraes¹,
Helder Oliveira¹, Denis Rosário¹ e Eduardo Cerqueira¹

¹Universidade Federal do Pará (UFPA), Belém – Pará – Brasil

{eduardo.lima.silva, jean.anjos.moraes}@itec.ufpa.br

{nagib, cerqueira, denis, heldermay}@ufpa.br

Abstract. *The increase in the number of LoRaWAN devices connected has increased collisions and interference on the network. In this sense, there is a need for a mechanism to manage the propagation factor used by devices during transmissions. However, the existing mechanisms do not address the priority of applications, causing devices with essential functions to have the same treatment as devices with less important functions, bringing less security from the application. This article proposes the PRA, a mechanism for allocating propagation factor according to the level of importance of the application, prioritizing applications with high and medium priority. Simulations show that the PRA reduces air time and energy consumption by up to 85% and 80% for groups of high and medium priority devices, respectively, compared to state-of-the-art mechanisms.*

Resumo. *O aumento no número de dispositivos LoRaWAN conectados, tem aumentado as colisões e interferências na rede. Neste sentido, surge a necessidade de um mecanismo para gerenciar o fator de propagação usado pelos dispositivos durante as transmissões. No entanto, os mecanismos existentes não tratam a prioridade das aplicações, ocasionando em dispositivos com funções importantes tenham o mesmo tratamento que dispositivos com funções menos importantes, trazendo menos segurança do ponto de vista da aplicação. Este artigo propõe o PRA, um mecanismo de alocação de fator de propagação de acordo com o nível de importância da aplicação, priorizando aplicações com alta e média prioridade. Simulações mostram que o PRA reduz em até 85% e 80% o tempo no ar e o consumo de energia para grupos de dispositivos de alta e média prioridade, respectivamente, se comparado a mecanismos do estado-da-arte.*

1. Introdução

O crescimento recente de Internet das Coisas (IoT), tem afetado uma grande quantidade de setores da indústria e quase todas as áreas de mercado [Firouzi et al. 2020]. Por exemplo, uma pesquisa recente, relatada no site da Forbes [Newman 2019], prevê mais de 75 bilhões de conexões de dispositivos IoT até 2025. Este cenário promete uma nova geração de comunicação sem fio que permite conectividade generalizada entre máquinas e objetos. Neste sentido, a tecnologia de área ampla de baixa potência (LPWAN, de *Low Power Wide Area Network*) é uma abordagem promissora para conectar milhões de “coisas”, em amplas áreas, com dispositivos que precisam transmitir dados bem específicos, num processo que consome pouca bateria [Sinha et al. 2017].

O LoRaWAN é uma LPWAN que destaca-se como uma das principais tecnologias emergentes, uma vez que, operam em faixas de frequência não licenciadas e de fácil implementação, além de possibilitar baixo consumo de energia e taxa de transferência adaptada ao contexto de IoT [Adelantado et al. 2017]. Na arquitetura LoRaWAN, os dispositivos finais (ED, de *End Device*) enviam dados para o servidor de aplicativos via *Gateway* (GW) por meio de uma comunicação sem fio de salto único. Com base nessa topologia, o LoRaWAN cobre uma área com dezenas de quilômetros e pode prover conectividade a milhares de EDs que exigem controle de acesso ao meio menos complexo às custas de baixa taxa de transferência [Qadir et al. 2018]. No entanto, a densificação do LoRaWAN para aplicações de IoT traz uma série de desafios devido a interferência e congestionamento de canais, por causa de muitas transmissões simultâneas utilizando as mesmas configurações de rádio [Adelantado et al. 2017]. Assim, configurar vários EDs utilizando o mesmo fator de propagação (SF, de *Spreading Factor*), na mesma frequência de portadora (CF, de *Carrier Frequency*), com a mesma largura de banda (BW, de *Bandwidth*) poderá causar um elevado número de colisões, onde o GW não poderá decodificar corretamente os sinais simultâneos.

A possibilidade de implementar um mecanismo de alocação de recursos mostrou-se eficiente para resolver problemas de colisões, onde parâmetros de rádio são configuradas de forma adaptável [Sallum et al. 2019]. Por exemplo, valores de SF mais altos aumentam o alcance do rádio ao custo de maior tempo no ar (ToA, de *Time on Air*) e consumo de energia [Cuomo et al. 2017]. Contudo, as soluções para IoT e LoRaWAN concentra-se na dificuldade em atender aos diversos requisitos de Qualidade de Serviço (QoS) gerados pela heterogeneidade das aplicações e suas demandas [Kavitha and Suseendran 2019]. Por exemplo, as aplicações IoT tem diferentes requisitos de QoS em termos de taxa de entrega de pacote ou atraso [Dawaliby et al. 2019]. Portanto, um mecanismo eficiente de alocação de recursos deve ajustar os parâmetros relacionados ao rádio em tempo real, tal como SF, de acordo com a prioridade de cada aplicação e condições da rede para prover melhor QoS para aplicações mais prioritárias.

Para atender a essas questões, este artigo apresenta um mecanismo de alocação de recursos para LoRaWAN ciente da prioridade das aplicações IoT, denominado de *Priority-aware Resource Allocation* (PRA). O mecanismo proposto tem como principal destaque a distribuição de SFs de acordo com a prioridade, de forma que EDs de alta prioridade usem o menor SF para obter menor ToA, bem como realiza o cálculo de limite de EDs em cada SF para diminuir a quantidade de colisões. Resultados de simulação demonstram a eficiência do PRA em termos de Taxa de extração de dados (DER, de *Data Extraction Rate*), ToA e consumo de energia comparado a mecanismos de alocação de recursos existentes.

O presente artigo será organizado conforme descrito a seguir. A Seção 2 apresenta uma visão geral sobre trabalhos que exploram alocação de recursos em LoRaWAN. A Seção 3 introduz o modelo de sistema e detalhes do PRA para definir as configurações ideais de SF. A Seção 4 explora o modelo de simulação desenvolvido para avaliar o modelo PRA, bem como os resultados obtidos. Por fim, a Seção 5 conclui o artigo com considerações finais e direções para trabalhos futuros.

2. Trabalhos Relacionados

Nesta seção apresentam-se os trabalhos mais relevantes para este trabalho, considerando principalmente as configurações dos dispositivos LoRaWAN. Além disso, em cada traba-

lho, discutimos suas vantagens e desvantagens.

Bouguera *et al.* propõem um modelo de consumo de energia para dispositivos LoRaWAN [Bouguera et al. 2018]. Os autores derivam a quantidade de energia gasta por bit em função do SF. Assim, este trabalho mostra como o SF afeta a transmissão de dados com diferentes cargas de tráfego. Apesar do modelo de consumo energético ser muito útil para aplicações IoT, os autores desconsideram QoS que é um dos requisitos chave para o sucesso do LoRaWAN IoT.

Dawaliby *et al.* desenvolveram uma solução baseada em Redes Definidas por Software em conjunto com o fatiamento de rede para otimização e gerência em larga escala do LoRaWAN [Dawaliby et al. 2019]. Para essa otimização, múltiplas fatias virtuais são construídas em cima de um GW LoRaWAN, na qual cada fatia da rede possui pesos para indicar sua importância ou urgência, confiabilidade e energia. Os autores buscam encontrar a estratégia ótima de fatiamento com as configurações corretas de SF e potência de transmissão que simultaneamente maximizam os indicadores de QoS no fatiamento e também minimiza o consumo de energia e custos de confiabilidade. Contudo, este trabalho adiciona uma sobrecarga de processamento para realizar o fatiamento do LoRaWAN o que reduz a disponibilidade de recursos de rede, bem como não considera o CF (*i.e.*, canais) na tomada de decisão. Além de não analisar o custo computacional para a aplicação da solução proposta, este trabalho também não analisa métricas de avaliação importante para aplicações IoT, tal como atraso, DER e colisões.

Cuomo *et al.* introduziu os modelos Explora-SF e Explora-AT visando otimizar desempenho do LoRaWAN configurando os valores de SFs para os dispositivos de acordo com a potência do sinal recebido (RSSI, de *Received Signal Strength Indication*), de forma que quanto melhor for o sinal recebido de dispositivo menor será o SF atribuído a ele [Cuomo et al. 2017]. Porém, os dois mecanismos limitam a quantidade de dispositivos em cada SF de formas diferentes. O Explora-SF equilibra a quantidade de dispositivos em cada SF, evitando assim excesso de dispositivos em um determinado SF com o objetivo de diminuir a quantidade de colisão de pacotes. Por outro lado, o Explora-AT faz a distribuição de dispositivos em cada SF baseado no ToA das transmissões. Porém, os modelos não possuem um mecanismo para priorizar dispositivos de acordo com a aplicação.

Com base na análise do estado da arte é possível concluir que um mecanismo de alocação de recursos para LoRaWAN é capaz de reduzir o impacto da colisão de pacotes em uma LoRaWAN densa. Porém, os mecanismo existentes não priorizam a transmissão de dispositivos de acordo com os requisitos aplicação.

3. Mecanismo PRA para Aplicações IoT com Diferentes Prioridades

Esta seção apresenta um mecanismo de alocação de recursos para configurar parâmetros do LoRaWAN de acordo com os requisitos da aplicação. O mecanismo proposto, Priority-aware Resource Allocation (PRA), distribui os dispositivos de alta e média prioridade nos menores SFs possíveis, consequentemente diminuindo o ToA de pacotes importantes ao custo de aumentar o ToA de pacotes de baixa prioridade.

3.1. Visão Geral do Cenário

Espera-se, que os cenários futuros de IoT tenham diferentes aplicações de monitoramento, tais como, saúde, logística, automação residencial/predial, rede inteligente e monitoramento de infraestrutura crítica. Tais aplicações IoT possuem diferentes requisitos em

termos de taxa de entrega de pacote ou atraso [Dawaliby et al. 2019]. Por exemplo, uma aplicação de monitoramento em tempo real tem requisito de confiabilidade e atraso mais restrito do que uma aplicação de monitoramento de rede elétrica inteligente.

O cenário simulado considera uma LoRaWAN com diferentes aplicações IoT e três níveis de prioridade diferentes. A rede é composta por um servidor de rede, um conjunto de GWs e um conjunto de EDs espalhados ao redor do GW. As prioridades são atribuídas aos EDs de forma pseudo-aleatória, onde um ED tem igual probabilidade de receber alta, média ou baixa prioridade, pois supõem-se que a prioridade do dispositivo depende apenas da aplicação. Os EDs são implantados no ambiente para coletar medições periodicamente, e enviar esses dados para o servidor de rede via GW. Para atingir um melhor desempenho, aumentando a entrega de pacotes, cobertura e diminuir os custos da implantação física, os GWs são dispostos no cenário por meio de um algoritmo de posicionamento ótimo, o qual define a topologia baseada em um método de clusterização, onde o cenário é dividido em vários *clusters*, *i.e.*, grupos de dispositivos e o GW é implantado na região central de cada grupo, conforme realizado em [Matni et al. 2019].

Projeções apontam que as aplicações IoT utilizando LoRaWAN serão densas [Dao et al. 2018], gerando a incapacidade de um GW decodificar corretamente os sinais simultâneos enviados por dispositivos que utilizam o mesmo SF no mesmo CF, tornando-se necessário considerar um mecanismo de alocação de recursos eficiente para ajustar os parâmetros de rádio para mitigar os efeitos da densificação de uma LoRaWAN. Desta forma, é possível fornecer um *tradeoff* entre o aumento da cobertura ao mesmo tempo que reduz o atraso, energia e interferência.

O SF pode ser definido como a razão entre a taxa de símbolo e de chirp. Assim, valores mais altos de SF aumentam a sensibilidade e o alcance do rádio ao custo de um aumento no ToA e no consumo de energia para transmitir um pacote [Adelantado et al. 2017]. Porém, pacotes enviados com valores de SF mais altos ocupam o canal por mais tempo, resultando em mais colisões. Por exemplo, para o pacote com uma carga útil de 20 bytes transmitida com SF 12 em vez de 11, o ToA aumenta de 659ms para 1318ms, respectivamente [Cuomo et al. 2017]. Além disso, um dispositivo usando SF11 consome dez vezes mais energia para transmissão do que quando utiliza SF7 [Duda and Heusse 2019]. Por fim, valores menores de SF (*i.e.*, SF7) suportam significativamente mais dispositivos com menor interferência em comparação com outros SFs, devido à relação entre a taxa de transmissão e ToA. Portanto, os EDs devem ter valores de SF ajustados pelo mecanismo de alocação de recurso de acordo com os requisitos da aplicação e condições de rede.

3.2. Algoritmo PRA

O algoritmo PRA descreve um mecanismo de alocação de recursos desenvolvido para LoRaWAN que gerencia os SFs disponíveis na rede e os distribui para os EDs. Essa distribuição leva em consideração um limite de EDs em cada SF, a prioridade do dispositivo, dada pela aplicação, e o ToA de cada SF. O PRA tem como objetivo diminuir o ToA de dispositivos de alta e média prioridade e diminuir a quantidade de colisões por SF de dispositivos de baixa prioridade.

Para proporcionar o maior alcance de transmissão, o PRA considera que a rede inicia todos os EDs no maior SF disponível, visto que maior o SF utilizado, maior o alcance de transmissão. Através do primeiro pacote recebido de cada ED, uma matriz

$RSSI_{mat}$ é criada para representar o RSSI da transmissão de cada ED recebido em cada GW, na qual cada linha da matriz $RSSI_{mat}$ representa um GW e cada coluna representa um ED da rede e os seus valores de RSSI a partir de cada GW. A partir da matriz $RSSI_{mat}$ podemos associar cada ED ao GW na qual ele possui maior RSSI e definir a ordem em que os EDs serão setados em cada SF. Para tal, um vetor $Prior_{vet}$ é criado para cada GW somente com os EDs na qual ele possui maior RSSI, ou seja, para cada coluna, que representa um dispositivo em $RSSI_{mat}$, será escolhido o maior valor de RSSI. Cada valor em $Prior_{vet}$ será multiplicado pelo nível de prioridade do respectivo ED. Os níveis de prioridade foram definidos como 1,2 e 3, para EDs de alta, média e baixa prioridade, respectivamente.

O mecanismo PRA é utilizado para cada $Prior_{vet}$, ou seja, para cada GW da rede. A partir do $Prior_{vet}$, o PRA calcula o limite de EDs para cada SF e seta o SF de cada ED levando em consideração a sua respectiva prioridade. Para diminuir a quantidade de colisões, o ToA entre os SFs deve ser equilibrado, evitando por exemplo que um GW se ocupe por mais tempo com EDs de um SF específico, para uma mesma taxa de envio de pacotes. Sendo assim, a quantidade de EDs em cada SF é limitada de acordo com o ToA gasto em cada transmissão no determinado SF. O limite de EDs em cada SF é computado pelo vetor lim_{vet} na Eq. 2, onde W_{vet} é o vetor pesos para cada SF dado pela Eq.c1, ToA_{vet} é o vetor que armazena o ToA de cada SF, como no exemplo da Tabela 1, $SomaToA_{vet}$ é o somatório de todos os elementos de ToA_{vet} , $SomaW_{vet}$ é o somatório de todos os elementos de W_{vet} e $nEDs$ é o tamanho de $Prior_{vet}$, ou seja, a quantidade de EDs para o respectivo GW.

$$W_{vet} = \left(\frac{ToA_{vet}}{SomaToA_{vet}} \right)^{-1} \quad (1)$$

$$lim_{vet} = \frac{W_{vet}}{SomaW_{vet}} \cdot nEDs \quad (2)$$

Tabela 1. ToA para pacotes de 20 bytes transmitidos em CR=4/5 e BW=125 kHz

SF	7	8	9	10	11	12
ToA(ms)	56.576	102.912	185.344	370.688	741.376	1318.912

O mecanismo PRA calcula o lim_{vet} de acordo com a Eq. 2, para cada ED é atribuído um novo valor de SF, iniciando no SF7 (Linha 5), visto que deseja-se setar os EDs de maior prioridade e RSSI no menor SF. Após essa etapa, o $newSF$ é incrementado enquanto o limite do SF não for atingido, e caso o máximo valor de SF, ou seja SF12, não for alcançado (Linhas 6 e 7). A ordem de prioridade é dada pelo vetor $Prior_{vet}$, na qual o ED de maior valor em $Prior_{vet}$, será setado para o SF calculado (Linha 8). A partir disso, é feito o decréscimo no limite de EDs (Linha 9), sinalizando que o respectivo SF encontra-se com menos espaços disponíveis em lim_{vet} . Assim, o valor de prioridade do ED em $Prior_{vet}$ é substituído por -666 (Linha 10), um valor menor que o mínimo possível em $Prior_{vet}$, para sinalizar que este ED está configurado. Desta forma, ao iniciar uma nova iteração, o maior valor de $Prior_{vet}$ será o próximo ED.

4. Avaliação

Esta seção apresenta o ambiente de simulação, bem como os parâmetros e métricas utilizadas, tais como DER, ToA e consumo de energia, para realizar um comparativo entre modelos de alocação de recursos do estado-da-arte, com o objetivo de validar o desempenho do mecanismo PRA.

Algorithm 1: Algoritmo PRA

```
1 Leia vetor  $Prior_{vet}$ , nEDs;  
2 Calcula vetor  $lim_{vet}$ ;  
3 for  $i \leftarrow 0$ ;  $i < nEDs$ ;  $i++$  do  
4   O ED escolhido é o que possui maior valor em  $Prior_{vet}$ ;  
5   Inicia o cálculo em SF7;  
6   while  $Limite\ de\ SF < 1$  e  $SF < 12$  do  
7     Passe para o próximo SF  
8   O ED recebe o SF calculado;  
9   O limite do novo SF em  $lim_{vet}$  é decrescido;  
10  O valor de ED em  $Prior_{vet}$  recebe -666;
```

4.1. Metodologia

O ambiente de simulação usado para gerar os resultados foi o LoRaSim [Voigt and Bor 2017], porém ao longo do trabalho algumas modificações foram implementadas. A partir do LoRaSim, é possível simular uma rede LoRa constituída por EDs e GWs, distribuídos em uma área bidimensional, e um servidor de rede, na qual é abstraído das funções do próprio LoRaSim.

A prioridade dos EDs foi definida como níveis 1,2 e 3, onde cada nível representa alta, média e baixa prioridade, respectivamente. O nível de cada ED foi escolhido de forma aleatória, onde cada ED possui a mesma probabilidade para cada nível de prioridade. Para avaliar o desempenho, foram feitas simulações com o PRA e os mecanismos Explora-SF e Explora-AT, usando os parâmetros da Tabela 2 de acordo com o padrão EU868 [Alliance 2017] e características elétricas do módulo LoRa SX1272 [Semtech 2015].

Tabela 2. Parâmetros de simulação

Parâmetro	Valor
Canal de Frequência	[867.1, 867.3, 867.5] MHz
BW	125 kHz
CR	4/5
Tamanho de pacote	20 bytes
Taxa de envio de pacotes	20 mins
Tempo de simulação	60 mins
Quantidade de GWs	1
Quantidade de EDs	[100 - 2000]
Área de distribuição dos EDs	2×2 km
Potência de transmissão	13 dBm

Foram utilizadas três métricas para avaliar o desempenho do modelo e/ou heurísticas de alocação de recursos LoRaWAN, nomeadamente, DER, ToA e consumo de energia. O DER avalia o desempenho em um intervalo numérico entre 0 e 1, no qual, nas implantações ótimas de rede, o valor é igual a 1. O DER não captura o desempenho de um dispositivo individual ele é uma métrica que analisa a implantação da rede como um todo, em que o DER é severamente afetado pelo número de colisões [Bor et al. 2016]. Para o cálculo do DER é necessário o número de pacotes recebidos e o número total de pacotes enviados. Porém, para avaliar o desempenho dos mecanismos, o DER foi calculado para cada prioridade, conforme a Eq. 3, onde DER_{prior} ,

$PacRec_{prior}$ e $PacEnv_{prior}$ são, respectivamente, o DER, a quantidade de pacotes recebidos e a quantidade de pacotes enviados, para cada prioridade.

$$DER_{prior} = \frac{PacRec_{prior}}{PacEnv_{prior}} \quad (3)$$

O ToA de um pacote em LoRaWAN é o tempo que um pacote leva para ser demodulado com sucesso. O ToA em milissegundos(ms) de cada pacote é dado por $ToAPac = D_{pream} + D_{carga}$, onde D_{pream} é a duração do preâmbulo e D_{carga} é a duração da carga útil. A duração do preâmbulo é calculada por $D_{pream} = (n_{pream} + 4.25) \times D_{simb}$, onde n_{pream} é a quantidade de símbolos no preâmbulo e D_{simb} é a duração de cada símbolo, dada por $D_{simb} = \frac{2^{SF}}{BW}$. A duração da carga útil é dada por $D_{carga} = nSimbCarga \times D_{simb}$, onde $nSimbCarga$ representa a quantidade de símbolos na carga útil e é calculado pela Eq. 4, onde PL é o número de bytes na carga útil, H é igual a 1 quando o cabeçalho está ativado(SF6) e igual a 0 quando está desativado, DE é igual a 1 para a configuração de otimização de baixa taxa de dados(BW=125 kHz e SF11/SF12) e igual a 0 para outras configurações, e CR é a taxa de código, que pode variar de 1 à 4.

$$nSimbCarga = 8 + \max \left(\text{Ceil} \left(\frac{(8PL - 4SF + 28 + 16 - 20H) \times (CR + 4)}{4(SF - 2DE)} \right), 0 \right) \quad (4)$$

Analogamente ao cálculo do DER, o ToA foi calculado individualmente para cada prioridade, visto que deseja-se avaliar o desempenho dos mecanismos para cada prioridade. O ToA total de cada prioridade ToA_{prior} em ms foi calculado pela Eq. 5, onde $ToAPac$ é o ToA em ms de cada pacote, $Pacs$ são os pacotes de cada ED e EDs_{prior} são os EDs de cada prioridade.

$$ToA_{prior} = \sum_{EDs_{prior}} \sum_{Pacs} ToAPac \quad (5)$$

O consumo energético é definido como o total de energia consumida pela rede em todas as transmissões demoduladas com sucesso. O consumo total $Cons_{prior}$ em milijoule(mJ) de cada prioridade é dado pela Eq. 6, onde $T = 3.3V$ e $C = 0.28A$ são, respectivamente, a tensão e a corrente nominal do módulo LoRa SX1272 de acordo com o *datasheet* [Semtech 2015] para potência de transmissão de 13dBm, e ToA_{prior} é a soma do ToA(ms) de todos os pacotes de cada ED para a determinada prioridade, como visto na Eq. 5.

$$Cons_{prior} = T \times C \times ToA_{prior} \quad (6)$$

4.2. Resultados

As características da alocação de SFs de cada mecanismo são mostradas na Figura 1. pode-se perceber que os mecanismos Explora-SF (Figura 1(a)) e Explora-AT(Figura 1(b)) não levam em consideração a prioridade de cada ED. O PRA por outro lado, além de fazer a distribuição de EDs de acordo com o ToA de cada SF, consequentemente concentrando os EDs nos SFs mais baixos possíveis, concentra os EDs de maior prioridade também nos SFs mais baixos. Assim, aplicações com maiores requisitos de QoS poderão fazer uso da vantagem de maior taxa de envio de pacotes, tendo como consequência uma maior confiabilidade na entrega dos dados, além de menor consumo energético, características dos SFs mais baixos.

A Figura 2 apresenta o DER de cada prioridade de acordo com o mecanismo de alocação de recurso usado. Como pode ser observado, apesar de possuir uma variação muito baixa de DER, o PRA obteve o melhor desempenho em todas as prioridades, em especial para a de EDs de baixa prioridade representada na Fig 2(c). Isso ocorre pois, como esperado, EDs de baixa prioridade

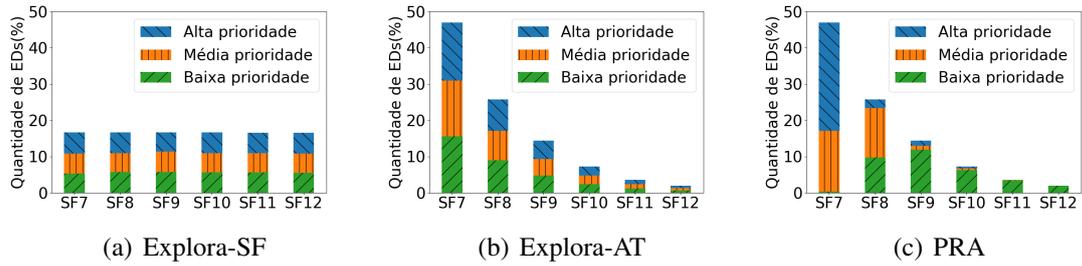


Figura 1. Quantidade de EDs em cada SF de acordo com a prioridade

foram setados ao maior SF, que apesar de possuir maior ToA, tem a menor porção de EDs da rede, e portanto menor probabilidade de colisão de pacotes. Além disso, o Explora-AT, por igualmente alocar os dispositivos nos SF de acordo com o ToA, demonstra um comportamento similar ao PRA. Por outro lado, o Explora-SF exprime menores valores em DER em relação aos demais, uma vez que distribui igualmente os dispositivos dentre os SFs, alocando uma grande quantidade de dispositivos em SFs mais altos, cujo detém de maior ToA. Assim, ocupa-se o canal por mais tempo e conseqüentemente elevam-se as chances de colisões e perda de pacotes [Caillouet et al. 2019].

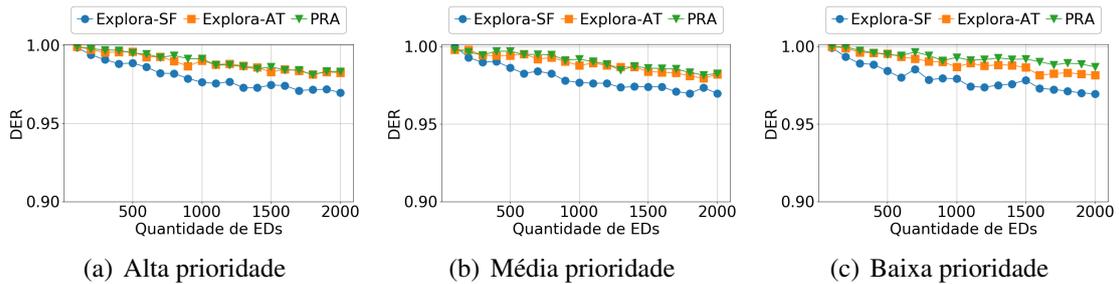


Figura 2. DER das prioridades de acordo com o mecanismo usado

A Figura 3 apresenta o ToA para diferentes quantidades de dispositivos e prioridades de acordo com o mecanismo de alocação de recurso usado. Como esperado, alocando os EDs de alta e média prioridade nos menores SFs, o PRA obteve o menor ToA para esses EDs. No grupo de prioridade alta, o PRA reduz em até 85% e no grupo de prioridade média até 80% o ToA em relação ao Explora-SF e em comparação ao Explora-AT, o PRA reduz 58% e 46% para os grupos prioridade alta e média, respectivamente (Figuras 3(a) e 3(b)), elucidando que o mecanismo proposto no artigo fornece uma maior velocidade na entrega de pacotes. Além disso, por alocar o grupo de dispositivos de baixa prioridade em SFs mais altos, o PRA obteve maior ToA em comparação aos outros dois grupos de prioridade. Porém, podemos observar na Figura 3(c) que o PRA demonstra-se um mecanismo notoriamente superior aos mecanismos de ADR os quais não consideram uma alocação baseada na prevalência de dispositivos nos SF mais baixo, como o Explora-SF.

A Figura 4 apresenta o consumo total de energia para diferentes quantidades de dispositivos e prioridades de acordo com o mecanismo de alocação de recurso utilizado. Como o consumo de um ED é diretamente proporcional ao ToA de seus pacotes transmitidos, os dispositivos de alta e média prioridade utilizando o mecanismo PRA são mais econômicos energeticamente, o que pode ser observado nas Figuras 4(a) e 4(b). Com a utilização do PRA o consumo energético foi reduzido em 85% e 80% nos grupos de EDs de alta e média prioridade, respectivamente, relativo ao Explora-SF e em relação ao Explora-AT houve uma redução de 58% para os EDs de alta prioridade e 46% para os EDs média prioridade.

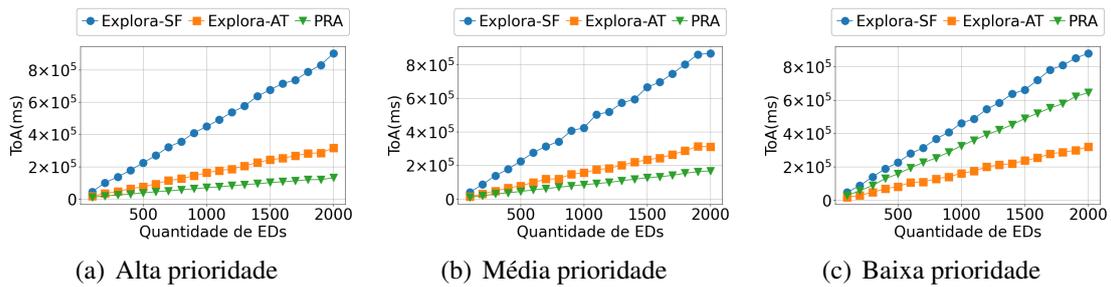


Figura 3. ToA total das prioridades de acordo com o mecanismo usado

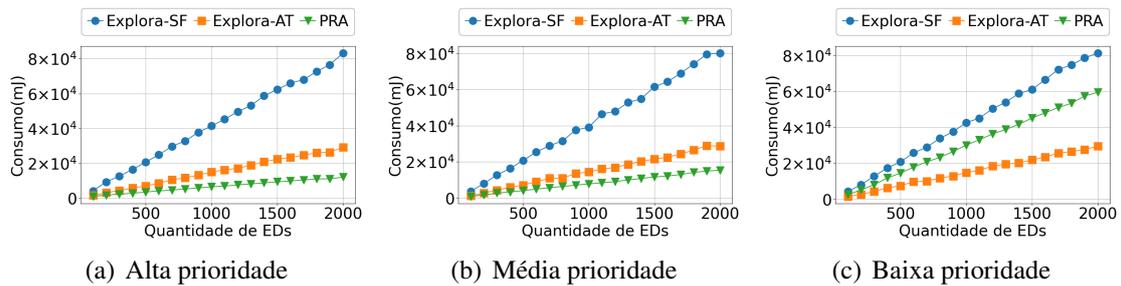


Figura 4. Consumo de energia das prioridades de acordo com o mecanismo usado

5. Conclusão

A densificação de dispositivos em LoRaWAN, sem a devida estratégia de distribuição de SFs, traz problemas de QoS à medida que aumenta a probabilidade de colisão de pacotes. Neste sentido, este trabalho apresentou um mecanismo para distribuir os dispositivos entre os SFs disponíveis levando em consideração o RSSI e a prioridade do dispositivo. O mecanismo PRA, proposto, possibilita que dispositivos com alta e média prioridade recebam o menor SF possível, com a finalidade de diminuir o delay dessas transmissões. Os resultados apresentados mostram que o PRA diminui o ToA de dispositivos de alta e média prioridade, e conseqüentemente reduz o delay e o consumo energético destes grupos. Além disso, apesar de aumentar o ToA e o consumo de dispositivos de baixa prioridade, o PRA melhora o DER reduzindo a quantidade de colisão de pacotes deste grupo. Como direcionamento futuro, espera-se que o algoritmo também gere outros parâmetros LoRaWAN, como potência de transmissão, afim de diminuir o consumo energético dos dispositivos, e a largura de banda, com o objetivo de reduzir a colisão de pacotes.

Referências

- Adelantado, F., Vilajosana, X., Tuset-Peiro, P., Martinez, B., Melia-Segui, J., and Watteyne, T. (2017). Understanding the limits of lorawan. *IEEE Communications magazine*, 55(9):34–40.
- Alliance, L. (2017). Lorawanm 1.1 regional parameters. *LoRa Alliance*.
- Bor, M. C., Roedig, U., Voigt, T., and Alonso, J. M. (2016). Do lora low-power wide-area networks scale? In *19th International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 59–67. ACM.
- Bouguera, T., Diouris, J.-F., Chaillout, J.-J., Jaouadi, R., and Andrieux, G. (2018). Energy consumption model for sensor nodes based on lora and lorawan. *Sensors*, 18(7):2104.

- Caillouet, C., Heusse, M., and Rousseau, F. (2019). Optimal SF Allocation in LoRaWAN Considering Physical Capture and Imperfect Orthogonality. In *IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, USA.
- Cuomo, F., Campo, M., Caponi, A., Bianchi, G., Rossini, G., and Pisani, P. (2017). Explora: Extending the performance of lora by suitable spreading factor allocations. In *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8. IEEE.
- Dao, N.-N., Vu, D.-N., Na, W., Kim, J., and Cho, S. (2018). Sgco: Stabilized green crosshaul orchestration for dense iot offloading services. *IEEE Journal on Selected Areas in Communications*, 36(11):2538–2548.
- Dawaliby, S., Bradai, A., and Pousset, Y. (2019). Network slicing optimization in large scale lora wide area networks. In *Proceedings of the IEEE Conference on Network Softwarization (NetSoft)*, pages 72–77. IEEE.
- Duda, A. and Heusse, M. (2019). Spatial issues in modeling lorawan capacity. In *22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 191–198.
- Firouzi, F., Farahani, B., Weinberger, M., DePace, G., and Aliee, F. S. (2020). Iot fundamentals: Definitions, architectures, challenges, and promises. In *Intelligent Internet of Things*, pages 3–50. Springer.
- Kavitha, K. and Suseendran, G. (2019). Priority based adaptive scheduling algorithm for iot sensor systems. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, pages 361–366. IEEE.
- Matni, N., Moraes, J., Rosário, D., Cerqueira, E., and Neto, A. (2019). Optimal Gateway Placement Based on Fuzzy C-Means for Low Power Wide Area Networks. In *IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–7. IEEE.
- Newman, D. (2019). Return on IoT: Dealing with the IoT skills gap. <https://www.forbes.com/sites/danielnewman/2019/07/30/return-on-iot-dealing-with-the-iot-skills-gap/#73f55d747091>. Accessed: 2020-03-20.
- Qadir, Q. M., Rashid, T. A., Al-Salihi, N. K., Ismael, B., Kist, A. A., and Zhang, Z. (2018). Low Power Wide Area Networks: A Survey of Enabling Technologies, Applications and Interoperability Needs. *IEEE Access*, 6:77454–77473.
- Sallum, E., Pereira, N., Alves, M., and Santos, M. (2019). Improving qos in lora low-power wide-area networks through optimized radio resource management. *Preprints*.
- Semtech, S. (2015). Datasheet (2015). <https://www.semtech.com/products/wireless-rf/lora-transceivers/sx1272>. Accessed: 2020-03-20.
- Sinha, R. S., Wei, Y., and Hwang, S.-H. (2017). A survey on lpwa technology: Lora and nb-iot. *Ict Express*, 3(1):14–21.
- Voigt, T. and Bor, M. (Última atualização: 10-07-2017). LoRaSim a discrete-event simulator based on simpy. <https://www.lancaster.ac.uk/scc/sites/lora/lorasim.html>, acessado em: 26-11-2019.