# Relatório Análise da Rede e Tráfego

Lucas Santos, 14/0151010
Eduardo Scartezini, 14/0137084
Pedro Aurélio, 14/0158103

## I. Introdução

Este trabalho foi proposto na disciplina de Teleinformática e Redes 1. Após construir uma simulação usando o simulador de eventos discretos NS-3, foi delegada aos alunos a tarefa de analisar os protocolos no trace de tráfego e compará-los a um ambiente real.

Desse modo, os alunos da disciplina poderiam fixar os conceitos aprendidos ao longo do semestre, bem como poder visualizar um exemplo (através da simulação) de como os conceitos vistos poderiam ser aplicados. Como conclusão para esse processo de aprendizado, haverá a comparação dos resultados da simulações com casos e protocolos reais.

## II. Conceitos Teóricos

### A. Arquitetura da Rede

NO presente trabalho, é considerada uma rede cuja arquitetura se divide em 5 camadas: aplicação (trata da segurança da rede), transporte (trata de informar se a transmissão é confiável ou não), rede (cuida do roteamento), enlace (trata de detecção de alguns erros e verificação de colisões nas transmissões ) e física (trata de enviar os sinais elétricos correspondentes à mensagem por algum meio físico). A camada de interesse para o trabalho é a de enlace, uma vez que esta contém os protocolos de controle de acesso ao meio (MAC protocols, em inglês).

### B. MAC

MAC protocols (medium access control protocols) são protocolos de rede que tentam solucionar o problema de múltiplas estações compartilharem o mesmo meio de transmissão.

Uma vez que múltiplas estações são obrigadas a compartilhar um mesmo meio de transmissão (um cabo coaxial, por exemplo), é razoável imaginar que haverá um número elevado de colisões entre elas (visto que tanto o tamanho de pacotes quanto o momento que eles chegam à estação para serem enviados podem ser considerados aleatórios).

Caso muitas colisões ocorram, a eficiência do canal será drasticamente comprometida, pois a informação de cada transmissão será perdida. Assim, será necessário reenviar todos os quadros que colidiram e acreditar que as outras estações não reenviaram seus quadros ao mesmo tempo.

Procurando aumentar a eficiência das redes de internet, foram criados diversos protocolos MAC. Dentre estes, serão analisadas simulações de redes contendo os protocolos : CSMA/CD e CSMA/CA. Além desses, serão analisados os protocolos ALOHA, SLOTTED ALOHA e ARP.

### C. IEEE

IEEE (Institute of Electrical and Electronics Engineers, ou, Instituto de Engenheiros Eletricistas e Eletrônicos) é a maior organização técnica e profissional de engenheiros e profissionais de áreas afins. O grupo é responsável por criar diversos padrões usados para os mais variados fins (padrões para ponto flutuante, IEEE 754, padrões para redes com e sem fio, IEEE 802.11 e IEEE 802.3 respectivamente). Dentre os diversos padrões criados, serão analisados comportamentos de redes que utilizam os padrões IEEE 802.3 e 802.11.

### D. Ferramentas utilizadas

PAra criar o ambiente de simulação (usando os protocolos IEEE 802.3 e IEEE 802.11), foi utilizado o simulador de circuitos NS-3. Para se usar o NS-3, necessita-se criar o ambiente de simulação usando a linguagem de programação C++.

Para obter as informações de trace necessárias para analisar o comportamento de uma estação ao utilizar um determinado protocolo, foi feito uso do analisador de pacotes Wireshark.

## III. Análise Experimental

### A. Controle de fluxo

AAnalise será baseada em uma rede IEEE 802.11, neste tipo de rede o protocolo encapsula um datagrama da camada de rede em um quadro contendo mais alguns cabeçalhos que servem para prove os serviços da camada de enlace, entre esses serviços estão: controle de erro, controle de fluxo, multiplexação. Vemos na figura 1 os campos que a camada de enlace insere para criação de um quadro.

Campos como *More Fragments*, *Fragments number*, *Retry*, *Duration*, são usados no protocolo para controle no enlace. Vemos na literatura que o IEEE 802.11 usa o CSMA/CA para acesso ao enlace, e nele é feito um controle de acesso ao enlace por requisições de RTS, e CTS ,uma explicação mais detalhada sobre esse processo estará na próxima seção deste relatório. Essa parte do protocolo é deixada transparente pela simulação.

Agora olhando para uma mensagem trocada por uma rede ethernet, que esta rodando o protocolo 802.3. Temos um pacote como exemplo na figura 2 vemos que agora a camada de enlace insere menos campos para criação de um quadro. Isso se deve ao fato, de que uma rede ethernet cabeada é muito mais confiável precisando de menos campos de controle, tendo assim um *overhead* menor. Observe que o *wireshark* não mostra o campo FCS que é o campo usado para identificação de erros conforme vimos na teoria.

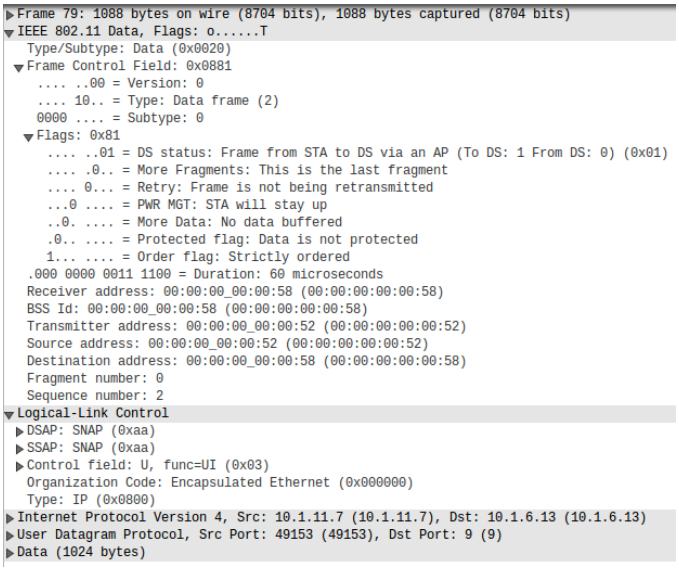


Figura 1. Pacote de uma rede IEEE 802.11

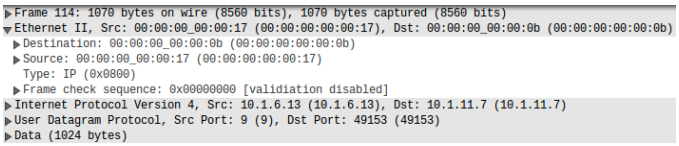


Figura 2. Pacote de uma rede IEEE 802.3

### *B. CSMA/CD,CSMA/CA, ALOHA, SLOTTED ALOHA*

O Protocolo ALOHA é dos mais antigos protocolos de acesso ao meio. Foi criado na década de 70 para uma comunicação entre vários campi em diferentes ilhas da Universidade do Havaí. A rede criada passou a ser chamada de ALOHAnet. O protocolo funciona de forma extremamente simples, para os padrões atuais.

1) Tem dados para enviar, envie-os;
2) Se houver colisão, termine de enviar o quadro e tente novamente após um tempo aleatório.

Logicamente quando começar a ter muitos pacotes circulando nessa rede, esse protocolo fica impraticável. Esse protocolo tem uma taxa de sucesso máxima de $18\%$. Isso significa que $82\%$ da largura de banda disponível é desperdiçada.

Naturalmente foram feitas tentativas de aumentar o desempenho do ALOHA. Uma dessas tentativas foi feita com *sloted* ALOHA. Ele consiste em dividir o tempo em *slots*, para que quando tiver dados para serem enviados, o emissor espere o inicio do próximo *slot* para enviá-los. Com isso, diminuímos a janela de colisão (quando existir colisão calcula-se uma probabilidade $p$ de se obter sucesso em enviar no próximo *slot*) e assim calcula-se quando será feito a retransmissão. Este protocolo possui uma grande desvantagem, que é a necessidade de haver sincronismo entre os nós da rede. Porém, tem um desempenho melhor do que o ALOHA com uma taxa de sucesso máxima de $37\%$.

Continuando com as evoluções dos protocolos, chegamos ao CSMA. Esse consiste em escutar o canal antes de transmitir e, se o canal estiver ocupado, adia a transmissão. Mesmo assim, colisões podem acontecer (devido a atrasos na transmissão, um dos nós pode não ouvir a transmissão do outro). Por esse motivo, a distância entres os nós é um grande fator nas colisões deste protocolo. No CSMA, todo o tempo de transmissão é desperdiçado quando ocorre uma colisão.

Para diminuir o tempo perdido por conta de uma colisão, foi proposto o CSMA/CD. Agora, além das caracteristicas do CSMA, as colisões são detectadas em pouco tempo, pois quando um transmissor começa a transmissão o mesmo fica observando o canal para detectar uma possível colisão. Se for detectada, a transmissão é abortada e um sinal informando colisão é lançado para que todos percebam que houve colisão e não tentem enviar mensagens pela rede desnecessariamente.

O protocolo CSMA/CD tem uma taxa de sucesso de $92\%$, muito melhor que os antigos ALOHA e *sloted* ALOHA. Por isso o protocolo CSMA/CD é utilizado em redes Ethernet $802.3$.

Detectar colisão, ou seja, ouvir o canal enquanto esta transmitindo é fácil para redes cabeadas, que medem a potência do sinal e comparam o sinal recebido com o transmitido. Já em redes sem fio é complicado pois o receptor é desligado durante a transmissão. Devido a esse fato, precisou-se modificar o CSMA para que redes sem fio pudessem ser feitas.

Para redes sem fio, temos o CSMA/CA. Nele, o nó que deseja se comunicar com outro pede autorização para ele enviando um sinal RTS (*Request To Send*). Se um nó receber um RTS e estiver livre para se comunicar, ele envia um sinal chamado CTS (*Clear To Send*). Somente depois de receber um CTS, um nó pode começar a transmitir dados para outro. Toda vez que um nó que não está envolvido na troca de dados percebe um RTS ou CTS na rede, ele fica sem enviar dados por algum tempo. Desta forma, a taxa de sucesso de envio chega próxima a $100\%$. Todas as colisões são evitadas, já que os nós só podem enviar dados quando recebem a confirmação do receptor de que eles podem transmitir sem problemas. As colisões com este protocolo só são possíveis em situações especiais em que cada nó possui um alcance de transmissão diferente ou caso existam nós móveis capazes de se deslocar pela área de transmissão. Problema tipico deste cenário é o problema do *host* escondido (na literatura há diversos textos abordando esse problema). Mesmo com uma taxa de sucesso sendo quase perfeita, temos um grande *overhead* de todas essas mensagens trocadas para o inicio de uma transmissão, afetando o *throughput* da rede.

Em suma, o protocolo de acesso de controle de acesso do meio (MAC protocol, em inglês) chamado CSMA/CD (utilizado no padrão IEEE 802.3) consiste em adotar, por padrão, que todas as estações prestes a iniciar uma transmissão no canal escutem para conferir se este está ocupado. A diferença principal entre este protocolo e protocolos iniciais (como ALOHA e Slotted ALOHA) está no fato de que esses últimos não escutam o canal de transmissão para verificar se ele está ocupado (esse comportamento ocorre pelo fato de que os protocolos iniciais tem o objetivo de proporcionar um modelo de comunicação simples para um ambiente com tráfego de dados leve). Além disso, para transmissões em meios sem fio (nos quais a detecção de colisões é extremamente difícil), o

protocolo CSMA/CA (protocolo CSMA para evitar colisões) é usado. Este faz com que todas as estações monitorem constantemente o canal para detectar se ele está livre ou não.

**Traces de Tráfego (IEEE 802.3)**

A seguir é mostrada uma imagem contendo uma parte do tráfego de uma rede que utiliza o padrão IEEE 802.3.

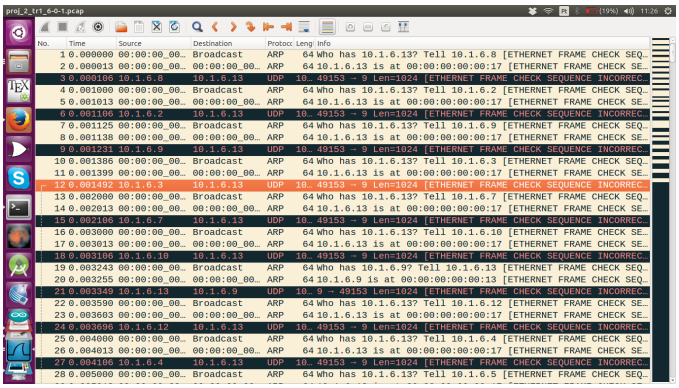

Figura 3. Comportamento da rede ao utilizar o padrão IEEE 802.3.

A figura 3 mostra diversos procedimentos adotados por redes que utilizam o padrão IEEE 802.3. Dentre eles, pode-se mencionar

- Quando há pacotes para enviar, a estação escuta a rede para ver se ela está ocupada. No trace, todas as estações perguntam se a rede está em uso. Se não estiver, elas tomam a rede para si através de ”Who has X.Y.Z.K? Tell A.B.C.D”, onde X.Y.Z.K é o endereço da rede e A.B.C.D o endereço da fonte que quer transmitir. Essa característica é típica de protocolos do tipo CSMA.
- Se uma estação quer transmitir, mas percebe que a rede está ocupada, ela agenda um tempo aleatório para novamente conferir se a rede está ocupada. Esse procedimento é tomado para espalhar as requisições, procurando evitar que mais de uma estação analise se a rede está ocupada ao mesmo tempo.

**Traces de Tráfego (IEEE 802.11)**

A seguir é mostrada uma imagem contendo uma parte do tráfego de uma rede que utiliza o padrão IEEE 802.11.

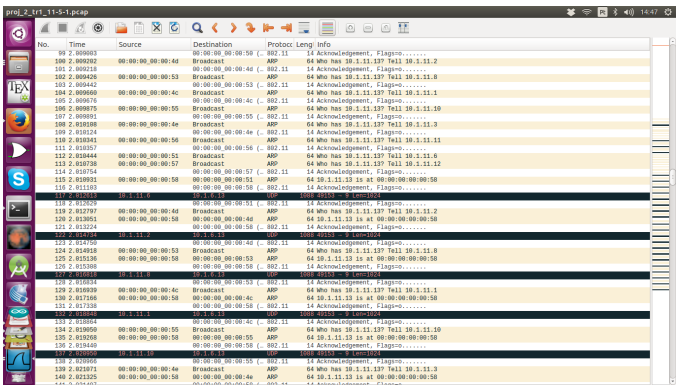

Figura 4. Comportamento da rede ao utilizar o padrão IEEE 802.11.

A figura 4 mostra diversos procedimentos adotados por redes que utilizam o padrão IEEE 802.11. Dentre eles, pode-se mencionar

- Quando há pacotes para enviar, a estação escuta a rede para ver se ela está ocupada. No trace, todas as estações perguntam se a rede está em uso. Se não estiver, elas tomam a rede para si através de ”Who has X.Y.Z.K? Tell A.B.C.D”, onde X.Y.Z.K é o endereço da rede e A.B.C.D o endereço da fonte que quer transmitir. Essa característica é típica de protocolos do tipo CSMA.
- Todas as estações que querem enviar ficam constantemente conferindo o estado da rede (se ela está desocupada, em ’idle’, ou ocupada, ’busy’). No momento em que a rede é desocupada, uma das estações toma o controle para si e começa a enviar. Na imagem todas as estações perguntam frequentemente com quem está a rede 10.1.11.13 . Assim que ela consegue, a estação que está com o controle da rede no momento (00:00:00:00:00:58) responde a todas as redes solicitantes que a rede está ocupada com a mensagem ”10.1.11.13 at 00:00:00:00:00:58”. Esse comportamento de sempre escutar a rede é devido ao fato de que o protocolo MAC utilizado no padrão IEEE 802.11 é o CSMA/CA (protocolo CSMA evitando colisões)

**Diagrama de fluxo**

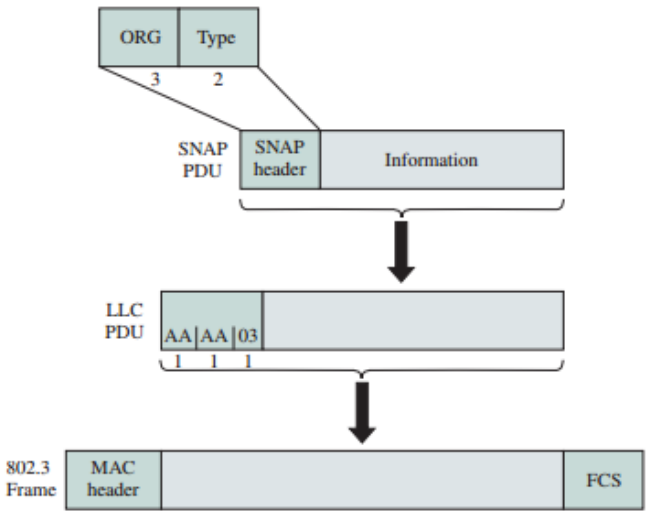


Figura 5. Encapsulamento de um quadro Ethernet [3]

É interessante observar a formação da pilha de protocolo (imagem 5) conforme foi estudado no livro de referência. A partir da imagem 6, observamos que um dado da aplicação é encapsulado em um pacote na camada de transporte, que por sua vez é encapsulado em um datagrama na camada de rede, que é encapsulado por um quadro na camada de enlace, para que por fim possa ser enviado pela camada física. Cada protocolo de suas respectivas camadas colocam cabeçalhos e caudas que irão ser lidos tanto pelos nós da rede quanto pelo receptor final (é assim que cada camada consegue fazer seus respectivos controles).

```
▶ Frame 117: 1088 bytes on wire (8704 bits), 1088 bytes captured (8704 bits)
▼ IEEE 802.11 Data, Flags: o......T
    Type/Subtype: Data (0x0020)
  ▶ Frame Control Field: 0x0881
    .000 0000 0011 1100 = Duration: 60 microseconds
    Receiver address: 00:00:00_00:00:58 (00:00:00:00:00:58)
    BSS Id: 00:00:00_00:00:58 (00:00:00:00:00:58)
    Transmitter address: 00:00:00_00:00:54 (00:00:00:00:00:54)
    Source address: 00:00:00_00:00:54 (00:00:00:00:00:54)
    Destination address: 00:00:00_00:00:58 (00:00:00:00:00:58)
    Fragment number: 0
    Sequence number: 2
▶ Logical-Link Control
▶ Internet Protocol Version 4, Src: 10.1.11.9 (10.1.11.9), Dst: 10.1.6.13 (10.1.6.13)
▶ User Datagram Protocol, Src Port: 49153 (49153), Dst Port: 9 (9)
▶ Data (1024 bytes)
```

Figura 6. Encapsulamento na pilha de protocolos

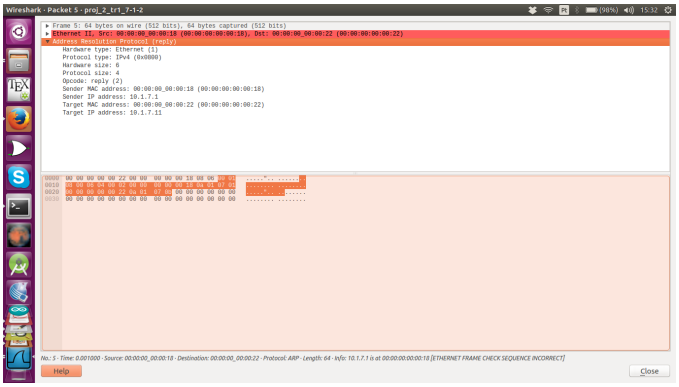

Figura 7. Conteúdo do protocolo ARP para um dos pontos da simulação.

### C. *Protocolo da camada de enlace: ARP*

O Address Resolution Protocol, em português: Protocolo de Resolução de Endereços, é um protocolo que roda na camada de enlace, definido pela RFC 826. O ARP é responsável por mapear um endereço de rede para um endereço físico, como por exemplo o endereço Ethernet (MAC - Medium Access Control, aceeso de controle ao meio).

Para fazer essa tradução de endereço IP para um endereço MAC, cada nó tem em sua RAM, uma tabela ARP, que contém o mapeamento entre esses dois endereços. Essa tabela também possui um campo TTL (time to alive) que indica por quanto tempo esse mapeamento esta valido.

Quando um hospedeiro desejar enviar um pacote para outro, eles devem consultar a tabela para identificar o endereço MAC de destino, se esse registro existir basta enviar. Agora se não tiver esse valor na tabela o remetente usa o protocolo ARP para converter o endereço. Primeiro, monta um pacote especial denominado pacote ARP. Um pacote ARP tem diversos campos, incluindo os endereços IP e MAC de envio e de recepção. Os pacotes ARP de consulta e de resposta têm o mesmo formato. A finalidade do pacote de consulta ARP é pesquisar todos os outros hospedeiros e roteadores na sub-rede para determinar o endereço MAC correspondente ao endereço IP que está sendo convertido.

Esse pacote ARP de consulta é enviado no endereço de difusão, todos os nós recebem esse pacote, o que satisfaz o endereço IP responde assim o hospedeiro pode, então, atualizar sua tabela ARP e enviar o seu datagrama.

Desejando enviar um datagrama para um host que estiver fora de sua sub-rede. Se usa, de maneira análoga, o protocolo ARP. Só que agora na tabela vai conter um endereço MAC para uma faixa de endereços IP, esse endereço MAC corresponde a um ponto de acesso que dar acesso a saída daquela sub-rede. Assim quando for enviado o datagrama o mesmo vai primeiro para o ponto de acesso, ele por sua vez vai enviar o pacote para dentro da próxima sub-rede que se aproxima do host final. Vemos assim que endereço MAC serve para identificar hospedeiros que estão no mesmo domínio de colisão.

**Traces de Tráfego (protocolo ARP)**

A figura 7 mostra o conteúdo do protocolo ARP em um dos momentos da simulação. Os itens contidos neles aparecem na seguinte ordem e com os seguintes valores:

- Tipo de Hardware: Ethernet
- Tipo de Protocolo: IPv4
- Tamanho do hardware: 6
- Tamanho do protocolo: 4
- Opcode: resposta (reply)
- Endereço MAC do emissor: 00:00:00:00:00:18
- Endereço IP do emissor: 10.1.7.1
- Endereço MAC do destino: 00:00:00:00:00:22
- Endereço IP do destino: 10.1.7.11

Esses campos estão de acordo com o esperado para um protocolo ARP comum, uma vez que este é responsável por mapear um endereço de rede para um endereço físico (por exemplo, de um endereço IP para um endereço MAC, como foi descrito acima).

### D. *Eficiência do protocolo de uma rede IEEE 802.11*

| Protocol | % Packets | Packets | % Bytes | Bytes | Mbit/s | End Packets | End Bytes | End Mbit/s |
|---|---|---|---|---|---|---|---|---|
| ▼ Frame | 100,00 % | 878 | 100,00 % | 74124 | 0,008 | 0 | 0 | 0,000 |
| ▼ IEEE 802.11 wireless LAN | 100,00 % | 878 | 100,00 % | 74124 | 0,008 | 84 | 1176 | 0,000 |
| IEEE 802.11 wireless LAN management frame | 80,75 % | 709 | 57,92 % | 42932 | 0,005 | 709 | 42932 | 0,005 |
| ▼ Logical-Link Control | 9,68 % | 85 | 40,49 % | 30016 | 0,003 | 0 | 0 | 0,000 |
| Address Resolution Protocol | 6,95 % | 61 | 5,27 % | 3904 | 0,000 | 61 | 3904 | 0,000 |
| ▼ Internet Protocol Version 4 | 2,73 % | 24 | 35,23 % | 26112 | 0,003 | 0 | 0 | 0,000 |
| ▼ User Datagram Protocol | 2,73 % | 24 | 35,23 % | 26112 | 0,003 | 0 | 0 | 0,000 |
| Data | 2,73 % | 24 | 35,23 % | 26112 | 0,003 | 24 | 26112 | 0,003 |

Figura 8. *Protocol Hierarchy Statistics*

Para esse cálculo, usamos um arquivo *pcap* obtido na simulação correspondente a uma rede IEEE 802.11, ou seja, que está rodando o protocolo CSMA/CA.

A eficiência consiste em comparar a quantidade de dados (em byte)s que circulará na rede, com a quantidade de informação real que foi transmitida.

Na simulação, foram enviadas 12 mensagens de 1024 *bytes* para servidores de *echo*, ou seja, há outras 12 mensagens de 1024 *bytes* de resposta.

Então no total circularam na rede $(12+12)*1024 = 24576$ *bytes* de informação real (desconsiderando os *bytes* usado para o funcionamento do protocolo).

Agora, para calcular o *overhead* temos que, para cada mensagem de 1024 *bytes*, foi enviado na camada física 1088 *bytes*. Essa diferença de 64 *bytes* é por conta do encapsulamento de cada camada.

Se fosse só esse o *overhead*, a rede teria uma eficiência de $\frac{1024}{1088} = 0.94$, ou seja, 94%. Porém, o protocolo CSMA/CA troca diversas mensagens antes de começar a enviar dados da aplicação.

Entretanto, como vemos na imagem 8, circulou na rede um total de $74124$ *bytes* para um total de $24576$ *bytes* de dados gerado pela aplicação. Assim, temos uma eficiência de $E = \frac{24576}{74124} = 0.33$. Logo, $E = 33\%$. É uma eficiência relativamente baixa, mas é necessário todo esse *overhead* para evitar colisões (como já foi explicado anteriormente). Toda essa análise foi relativa a captura *proj_2_tr1_11-5-1.pcap* da simulação.

## IV. Conclusões

A Partir de uma simulação com o simulador de eventos discretos ns-3, foi possível identificar e analisar os protocolos visto em sala de aula, que são usados em uma rede real.

Com grande enfoque na camada de enlace do modelo TCP/IP, vimos protocolos que rodam em diferentes tipo de rede, IEEE 802.3 e IEEE 802.11. Cada um deles com seus desafios inerentes a redes que os rodam.

Vimos também como cada um dos protocolos usam a rede que estão rodando para conseguir entregar comunicação confiável.

Assim, pode-se dizer que os objetivos do presente trabalho foram realizados.

## Referências

[1] Solis, Priscila, **Teleinformática e Redes 1 - Notas de aula e transparências**. Brasília, Brasil: Universidade de Brasília, 2017.

[2] Ross,Keith W.; Kurose,Jim, **Redes de Computadores e A Internet - Uma Abordagem Top-Down**, 1a edição. São Paulo, Brasil: Pearson Education - Br, 2013.

[3] Leon-Garcia, Alberto; Widjaja, Inra, **Communication Networks**, 2nd edition. New York, New York, USA: McGraw-Hill Education, July 16, 2003, ISBN-13: 978-0072463521, ISBN-10: 007246352X.