



**UNIVERSIDADE FEDERAL DO PARÁ  
CAMPUS UNIVERSITÁRIO DE CASTANHAL  
FACULDADE DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**LEONARDO WALLACE DA SILVA SOUZA**

**SIMULAÇÃO E AVALIAÇÃO DE DESEMPENHO EM REDES DE SENSORES  
AQUÁTICAS**

**CASTANHAL  
2019**



**UNIVERSIDADE FEDERAL DO PARÁ  
CAMPUS UNIVERSITÁRIO DE CASTANHAL  
FACULDADE DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**LEONARDO WALLACE DA SILVA SOUZA**

**SIMULAÇÃO E AVALIAÇÃO DE DESEMPENHO EM REDES DE SENSORES  
AQUÁTICAS**

Trabalho de Conclusão de Curso apresentado  
para obtenção do grau de Bacharel em Sistemas  
de Informação.

Orientador: Prof. Dr. José Jailton Henrique Fer-  
reira Júnior

**CASTANHAL  
2019**

**LEONARDO WALLACE DA SILVA SOUZA**

**SIMULAÇÃO E AVALIAÇÃO DE DESEMPENHO EM REDES DE  
SENSORES AQUÁTICAS**

Trabalho de Conclusão de Curso apresentado  
para obtenção do grau de Bacharel em Sistemas  
de Informação.

Data da Defesa: 26 de Junho de 2019

Conceito:

**Banca Examinadora**

---

**Prof. Dr. José Jailton Henrique Ferreira  
Júnior**

Faculdade de Computação - UFPA  
Orientador

---

**Prof. MSc. Felipe André da Costa Brito**

Faculdade de Computação - UFPA  
Membro da Banca

---

**Prof. Dr. Igor Ruiz Gomes**

Faculdade de Computação - UFPA  
Membro da Banca

CASTANHAL  
2019

*Dedico este trabalho aos meus Pais que sempre estiveram comigo nos momentos mais difíceis, em especial minha Avó, que foi uma pessoa essencial para que eu conseguisse chegar até aqui, por toda ajuda e sacrifício nesses anos, durante esta jornada.*

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, minha avó, minha mãe, meu pai e minha namorada por todo apoio nas horas mais difíceis a qual precisei, todos foram de fundamental importância no apoio as dificuldades, sem a ajuda de vocês esse sonho não seria possível.

Agradeço aos meus amigos de classe pela troca de conhecimento, principalmente a equipe da HUE Systems, meus amigos Juan Carlos, Kennedy Souza, Elizeu Gabriel, Higor Silva e Carlos André. Todos contribuíram bastante nas atividades em equipe e na troca de conhecimento. E ao meu orientador Prof. Dr. José Jaílton Júnior (JJJ), pela oportunidade e o desafio de me oferecer esse projeto.

Agradeço com um carinho enorme aos meus amigos Fábio Silva e Juan Carlos pois foram de fundamental importância não só nas atividades extraclasse, mas que também atuaram como irmãos e amigos e sempre colaboraram com ensinamentos que levarei para o resto da vida. Agradeço aos meus professores que passaram pela FACOMP por me ensinarem o caminho do conhecimento. Muito Obrigado Mestres e Doutores.

*“Conhece-te a ti mesmo e conhecerás o universo e os deuses.”*  
*(Sócrates)*

## RESUMO

As redes de sensores aquáticas (RSAs) com transmissão acústica começaram a ser estudadas a princípio com a motivação do desenvolvimento de novas aplicações para a exploração, estudo e a proteção dos oceanos. As RSAs apresentam diversas diferenças em relação às redes de sensores terrestres, sendo as principais o maior consumo de energia necessário para a transmissão. Nas RSAs o som é a principal ferramenta usada como o meio de comunicação, devido às restrições da comunicação via rádio e da comunicação ótica no meio aquático, a transmissão acústica tornou-se a alternativa mais viável, mesmo com as restrições impostas pelo canal: Largura de banda reduzida, atraso, alta atenuação, alta latência e largura de banda limitada de canais acústicos tornam os protocolos MAC existentes projetados para redes de rádio impraticáveis ou pouco eficientes de energia para o sensor submarino de redes. Este trabalho avalia através de simulações a viabilidade de aplicações de monitoramento, controle e consumo de energia utilizando as redes de sensores submarinas. Para a realização das simulações foi utilizado o simulador *Network Simulator-2* juntamente com o módulo *Aqua-sim*, que efetivamente simula a atenuação no canal acústico submarino, os atrasos e as colisões na rede. São usadas três topologias básicas como padrão para as redes submarinas: quadro, estrela e linha. Os resultados mostram quais topologias associadas aos protocolos MAC atendem melhor as necessidades e as peculiaridades da grande maioria das aplicações de monitoramento e controle utilizando redes acústicas submarinas. O objetivo final deste trabalho é encorajar os esforços de pesquisa para estabelecer princípios fundamentais para o desenvolvimento de novas técnicas avançadas para eficiência na comunicação submarina e redes para aplicações melhoradas de monitoramento e exploração aquática.

**Palavras-chave:** Redes de sensores aquáticas; MAC; R-MAC; Aqua-sim; Consumo de Energia; Simulação; NS-2; Monitoramento; RSAs; Vazão.

## ABSTRACT

The underwater sensor networks with acoustic transmission began to be studied with the motivation of the development of new applications for the exploration, study and the protection of the oceans. The underwater sensor networks present different differences in relation to terrestrial sensor networks, the main ones being the greater energy consumption required for the transmission. In submarine sensor networks, sound is the main tool used as the communication medium, due to the restrictions of radio communication and optical communication in the aquatic environment, acoustic transmission has become the most viable alternative, even with the restrictions imposed by the Channel: Reduced bandwidth, delay, high attenuation, high latency and limited bandwidth of acoustic channels make existing MAC protocols designed for impractical or inefficient radio networks for underwater network sensor. This work evaluates through simulations the feasibility of monitoring, control and energy consumption applications using the submarine sensor networks. For the simulations, the simulator textit Network Simulator-2 and the module textit Aqua-sim were used, which effectively simulates the attenuation in the submarine acoustic channel, delays and collisions in the network. Three basic topologies are used as the standard for underwater networks: frame, star and line. In relation to the MAC protocols, we propose the R-MAC. The results show that topologies associated with MAC protocols better meet the needs and peculiarities of the vast majority of monitoring and control applications using subsea acoustic networks.

**Keywords:** Underwater sensor networks; MAC; R-MAC; Aqua-sim; Energy consumption; Simulation; NS-2; Monitoring; Acoustic Networks; Flow-rate.



## LISTA DE ILUSTRAÇÕES

Figura 1 – Redes de sensores aquáticas. . . . .	20
Figura 2 – Rede de sensor aquática - Arquitetura. . . . .	22
Figura 3 – ROVs. . . . .	24
Figura 4 – Arquitetura interna de um nó sensor subaquático. . . . .	25
Figura 5 – Classificação de aplicações de redes de sensores aquáticas. . . . .	26
Figura 6 – Topologias . . . . .	39
Figura 7 – Cenário . . . . .	40
Gráfico 1 – Gráfico - Topologia quadro . . . . .	42
Gráfico 2 – Gráfico - Topologia estrela . . . . .	42
Gráfico 3 – Gráfico - Topologia linha . . . . .	43
Gráfico 4 – Atraso - Topologia Quadro . . . . .	43
Gráfico 5 – Atraso - Topologia Estrela . . . . .	44
Gráfico 6 – Atraso - Topologia Linha . . . . .	44
Gráfico 7 – Média consumo de energia. . . . .	45

## **LISTA DE TABELAS**

Tabela 1 – Aplicações de monitoramento . . . . .	28
Tabela 2 – Aplicações de prevenção de desastres . . . . .	30
Tabela 3 – Aplicações R-MAC . . . . .	40

## LISTA DE ABREVIATURAS E SIGLAS

1D	<i>Uma dimensão</i>
2D	<i>Duas dimensões</i>
3D	<i>Três dimensões</i>
4D	<i>Quatro dimensões</i>
AUV	<i>Autonomous Underwater Vehicle</i>
CDMA	<i>Code Division Multiple Access</i>
CE	<i>Condutividade Elétrica</i>
CTS	<i>Clear-to-send</i>
CSMA	<i>Carrier Sense Multiple Access</i>
DSSS	<i>Direct Sequence Spread Spectrum</i>
Enor	<i>Energy-aware Opportunistic Routing</i>
FAMA	<i>Floor Acquisition Multiple Access</i>
FDMA	<i>Frequency Division Multiple Access</i>
FHSS	<i>Frequency Hopped Spread Spectrum</i>
LUM	<i>Luminosidade Subaquática</i>
MAC	<i>Media Access Control</i>
MACA	<i>Multiple Access with Collision Avoidance</i>
NTF	<i>Notification Thresholds</i>
NS-2	<i>Network Simulator</i>
QoS	<i>Quality of Service</i>
RF	<i>Radio Frequency</i>
R-MAC	<i>Reservation Based MAC</i>
RTS	<i>Request-to-send</i>
ROV	<i>Remotely Operative Vehicle</i>
RSAs	<i>Redes de Sensores Aquáticas</i>

SNR	<i>Relação Sinal Ruído</i>
TDMA	<i>Time Division Multiple Access</i>
VBF	<i>Vector-Based Forwarding</i>

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
<b>1.1</b>	<b>Visão Geral</b>	<b>14</b>
<b>1.2</b>	<b>Motivação</b>	<b>15</b>
<b>1.3</b>	<b>Objetivos</b>	<b>15</b>
<b>1.4</b>	<b>Organização do Trabalho</b>	<b>16</b>
<b>2</b>	<b>TRABALHOS CORRELATOS</b>	<b>17</b>
<b>2.1</b>	<b>Visão Geral</b>	<b>17</b>
<b>3</b>	<b>REDES DE SENSORES AQUÁTICAS</b>	<b>19</b>
<b>3.1</b>	<b>Estado da Arte e Tendências</b>	<b>19</b>
<b>3.2</b>	<b>Características do canal acústico submarino</b>	<b>20</b>
<b>3.3</b>	<b>Diferenças para Redes Terrestres</b>	<b>21</b>
<b>3.4</b>	<b>RSAs - Arquitetura de Rede</b>	<b>22</b>
3.4.1	Arquitetura 1D	22
3.4.2	Arquitetura 2D	23
3.4.3	Arquitetura 3D	23
3.4.4	Arquitetura 4D	23
<b>3.5</b>	<b>Arquitetura do sensor</b>	<b>24</b>
<b>3.6</b>	<b>Noções básicas de propagação aquática</b>	<b>25</b>
<b>3.7</b>	<b>Redes de sensores aquáticas: Aplicações</b>	<b>26</b>
3.7.1	Aplicações de Monitoramento	26
3.7.1.1	Qualidade da água	27
3.7.1.2	Habitat	27
3.7.1.3	Exploração submarina	28
3.7.2	Aplicações de prevenção de desastres	29
3.7.2.1	Inundações	29
3.7.2.2	Vulcão, terremoto e tsunami	30
3.7.2.3	Derramamento de óleo	30
<b>3.8</b>	<b>Camada de Enlace</b>	<b>31</b>
3.8.1	Protocolos Baseados em Partição	31
3.8.1.1	FDMA	31
3.8.1.2	TDMA	32
3.8.1.3	CDMA	32
3.8.2	Protocolos Baseados em Acesso Aleatório	33
3.8.2.1	ALOHA	33
3.8.2.2	CSMA	34
<b>3.9</b>	<b>Procolos abordados neste trabalho</b>	<b>35</b>
3.9.0.1	R-MAC: Procoloto MAC Baseado em Reserva	35

3.9.1	Protocolos de roteamento . . . . .	36
4	<b>METODOLOGIA . . . . .</b>	<b>38</b>
4.1	<b>Recursos Utilizados . . . . .</b>	<b>38</b>
4.2	<b>Implementação . . . . .</b>	<b>38</b>
4.3	<b>Cenário . . . . .</b>	<b>39</b>
5	<b>RESULTADOS . . . . .</b>	<b>41</b>
5.1	<b>Simulações . . . . .</b>	<b>41</b>
5.2	<b>Média do consumo de energia . . . . .</b>	<b>45</b>
6	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>47</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>48</b>
	 <b>APÊNDICES</b>	 <b>51</b>
	<b>APÊNDICE A – TOPOLOGIA QUADRO . . . . .</b>	<b>52</b>
	<b>APÊNDICE B – TOPOLOGIA ESTRELA . . . . .</b>	<b>62</b>
	<b>APÊNDICE C – TOPOLOGIA LINHA . . . . .</b>	<b>72</b>

# 1 INTRODUÇÃO

Este capítulo irá expor brevemente as Redes de Sensores Aquáticas (RSAs), apresentando as principais vantagens e desafios enfrentados por esse tipo de rede, sintetizando suas principais aplicações e suas diferenças em relação às redes de sensores terrestres.

## 1.1 Visão Geral

As RSAs começaram a ser estudadas com a motivação do desenvolvimento de novas aplicações para a exploração, estudo e a proteção dos oceanos. Apesar de sua importância, o monitoramento de ambientes subaquáticos ainda é uma tarefa difícil e dispendiosa. Para que seja possível a transmissão através do canal acústico os protocolos MAC e de rede devem ser capazes de operar dentro das condições do meio aquático, que impõe grandes dificuldades à comunicação. Devido às restrições da comunicação via rádio e da comunicação ótica no meio aquático a transmissão acústica tornou-se a alternativa mais viável, mesmo com as restrições impostas pelo canal: largura de banda limitada, *fading*, alta taxa de atenuação, alta latência e interferência multicaminhos; características que influenciam todas as camadas de protocolos, a arquitetura do sistema e o desenvolvimento das aplicações. As redes submarinas de sensores apresentam diversas diferenças em relação às redes de sensores terrestres, sendo as principais o maior consumo de energia necessário para a transmissão, a necessidade de uma maior capacidade de memória e o custo dos sensores (VIEIRA L., 2010).

O planeta Terra é composto de água. Cerca de dois terços da superfície da Terra é coberta por oceanos, que em grande parte está inexplorada. Além disso, há uma grande quantidade de recursos naturais a serem descobertos. Também, temos o fato de que os oceanos são responsáveis, em algumas situações, de serem os grandes causadores de desastres naturais como tsunamis e/ou enchentes causando um enorme impacto no clima global. As RSAs são um tema recente de pesquisa que possui inúmeras aplicações. Algumas áreas e aplicações que podem se beneficiar de rede de sensores aquáticas são (KONG J.; GERLA, 2005): oceanografia, biologia marinha, estudos da interação entre oceanos e atmosfera, estudos do clima, aquecimento global, arqueologia no fundo do mar, predições sísmicas, detecção de poluentes e substâncias contaminantes, controle da qualidade da água e exploração e monitoração de campos de gás, óleo e petróleo.

É importante observar que cada uma dessas aplicações pode ter características distintas e, conseqüentemente, levar a diferentes projetos. A análise das características do canal acústico é fundamental para a definição de quais protocolos serão utilizados com o objetivo de determinar a viabilidade de uma aplicação. Além disso, estas informações podem auxiliar muito no desenvolvimento de novos protocolos, específicos para redes submarinas com transmissão acústica. O estudo e o desenvolvimento de uma tecnologia de transmissão acústica e de topologias de rede que superem as limitações do meio aquático são essenciais para a viabilidade das aplicações.

Listamos abaixo algumas aplicações que podem ser implementadas através de uma rede acústica submarina:

- Monitoramento ambiental: monitoramento de poluição química, biológica ou nuclear (ZHANG B, 2004);
- Monitoramento de dados oceanográficos: monitoramento da temperatura, salinidade, condutividade, correntes marítimas, pressão, ondas, marés e emissão de poluentes;
- Exploração submarina: detecção de reservas de petróleo e/ou auxílio na exploração;
- Controle de robôs ou veículos submarinos: rede para controle e localização de robôs e veículos submarinos utilizados na exploração submarina;
- Prevenção de desastres: sensores instalados no leito oceânico podem medir variações sísmicas com o objetivo de se analisar os efeitos de terremotos submarinos e detectar a formação de tsunamis (SOREIDE N.N., 2004);
- Aplicações militares: as redes podem ser instaladas em determinadas áreas de interesse militar para a detecção de mergulhadores, submarinos ou navios inimigos. (CAYIRCI E., 2004)
- Monitoramento de reservatórios de água potável: utilização de uma rede de sensores para o monitoramento e o controle de reservatórios de água potável (YANG ONG, 2002).

## 1.2 Motivação

Como pode ser observado, as RSAs possuem uma grande diversidade de aplicações em que podem ser utilizadas em benefício da sociedade. Por mais que tenham suas restrições devido ao ambiente em que são aplicadas. Oceanos, rios e lagos (VIEIRA L. F. M.; VIEIRA, 2018) são fundamentais para a vida em nosso planeta e uma série de segmentos da sociedade exige monitoramento subaquático. Este tipo de atividade é extremamente importante em um país como o Brasil, visto que o nosso país é extremamente rico em recursos naturais que podem ser explorados. Também temos o fato que, algumas vezes, nossa sociedade acaba sendo vítima de alguns desastres naturais, pelo simples fato de não haver sensores de monitoramento em algumas áreas podendo de alguma forma prevenir tragédias maiores. Esse tipo de atividade requer mão-de-obra qualificada e a capacidade de realizar pesquisa de ponta em sensoriamento aquático. Para que se possa usufruir dos recursos naturais presentes na plataforma costeira é necessário que seja possível gerarmos o conhecimento desejável à exploração e defesa de tais recursos.

## 1.3 Objetivos

O principal objetivo deste trabalho é a análise do comportamento das redes acústicas submarinas através da ferramenta NS-2 (*Network Simulator*), juntamente com o módulo *Aqua-*



*sim*, que efetivamente simula a atenuação no canal acústico submarino, os atrasos e as colisões na rede. O *Aqua-sim* é uma ferramenta que está integrada no NS-2 para lidar com redes aquáticas. Este simulador suporta nova camada MAC, novos protocolos que são úteis no roteamento dentro da água. Serão utilizadas diferentes topologias com o objetivo de auxiliar o desenvolvimento de aplicações baseadas neste tipo de rede. Este trabalho avalia através de simulações a viabilidade de aplicações de monitoramento e controle e consumo de energia utilizando as redes acústicas submarinas. Em relação ao protocolo MAC, propomos o R-MAC (*Reservation Based MAC*). As topologias apresentadas como básicas para as redes submarinas: quadro, estrela e linha. Os resultados mostram qual topologia associada ao protocolo MAC atende melhor as necessidades e as peculiaridades da grande maioria das aplicações de monitoramento e controle utilizando redes acústicas submarinas.

Para validação e obtenção da visualização dos gráficos, serão utilizadas as ferramentas XGRAPH, MATLAB e EXCEL que são ferramentas que servem para plotagem dos gráficos. Essas ferramentas trabalham paralelamente com os arquivos gerados pelo NS-2, são chamados arquivos de *trace*, onde todas as informações do tráfego das simulações estão contidas, dando a possibilidade da análise dos gráficos. O objetivo final deste trabalho é encorajar os esforços de pesquisa para estabelecer princípios fundamentais para o desenvolvimento de novas técnicas avançadas para eficiência na comunicação submarina e redes para aplicações melhoradas de monitoramento e exploração aquática.

## 1.4 Organização do Trabalho

O trabalho está organizado da seguinte forma, no segundo capítulo serão apresentados trabalhos correlatos comparando as propostas de outros autores com a proposta a qual se propõe este trabalho. No capítulo terceiro, serão apresentados conceitos relacionados as RSAs. Para o quarto capítulo, serão apresentados a proposta de qual topologia associada ao protocolo MAC atende melhor as necessidades e as peculiaridades da grande maioria das aplicações de monitoramento e controle utilizando redes acústicas submarinas. No quinto capítulo, serão apresentados os resultados das pesquisas do trabalho, com todos os cenários e gráficos. Já no sexto e último capítulo serão apresentados as considerações finais, em seguida das referências e apêndices com a codificação das simulações.

## 2 TRABALHOS CORRELATOS

Neste capítulo, será realizado um apanhado geral sobre alguns trabalhos relacionados as RSAs. Com intuito de apontar algumas soluções relacionados a qualidade dessas redes e justificar através dos trabalhos propostos a grande contribuição a qual esse trabalho se propõe a fazer.

### 2.1 Visão Geral

Em (XIE; CUI, 2007a), foi proposto um protocolo MAC de eficiência energética baseado em reserva, o R-MAC, para redes de sensores aquáticas. Onde o R-MAC agenda cuidadosamente as transmissões de controle e pacote de dados para evitar colisões de pacotes de dados, ele agenda cuidadosamente as transmissões de controle e pacotes de dados para evitar colisões de pacotes de dados. Os algoritmos de escalonamento não apenas evitam completamente as colisões de pacotes de dados, mas também resolvem o problema do terminal exposto. No R-MAC, cada nó sensor usando escuta periódica e modos de dormir para reduzir a energia gasta em estados ociosos. Além disso, o R-MAC afrouxa o requisito de sincronização, permitindo que cada nó aleatoriamente seleciona seu próprio agendamento.

Os autores (XIE ZHONG ZHOU, 2009), apresentaram um desing e implementação do *Aqua-sim*, um simulador de rede de sensores subaquáticos baseado no NS-2. Através de vários estudos de caso, foi mostrado que o *Aqua-sim* é uma poderosa ferramenta de simulação, com alta fidelidade e flexibilidade, para pesquisa de redes subaquáticas propondo como algum trabalho futuro, avaliar o uso do *Aqua-sim* com experimentos reais, especialmente depois de incorporar modelos de canais avançados.

Numa tentativa de reduzir o custo de energia e também melhorar a transferência e a equidade em toda a rede, os autores (ALFOUZAN ALIREZA SHAHRABI; BOUTALEB, 2017), proporam um novo protocolo MAC baseado em reservas (*R-MAC*), que emprega um mecanismo de ciclo de trabalho para efetivamente tentar reduzir as colisões e retransmissões. Neste trabalho foi feito um extensivo estudo de simulação que mostra que o protocolo proposto pode lidar eficientemente com a contenção de tráfego para alcançar melhorias significativas em termos de consumo de energia, produtividade e equidade.

Os autores (XIE; LAO, 2006) o protocolo de roteamento, chamado de *VBF (Vector-Based Forwarding)*, para fornecer roteamento robusto, escalável e energeticamente eficiente. O *VBF* é essencialmente uma abordagem de roteamento baseada em posição: os nós próximos ao “vetor” da origem ao destino encaminharão a mensagem. Dessa forma, apenas uma pequena fração dos nós está envolvida no roteamento. O *VBF* também adota um algoritmo de auto-adaptação localizado e distribuído que permite aos nós pesar o benefício dos pacotes de encaminhamento e, assim, reduzir o consumo de energia descartando os pacotes de baixo benefício.

O trabalho proposto pelos autores (FELEMBAN FAISAL KARIM SHAIKH, 2015), apresentaram uma revisão bastante abrangente da literatura sobre aplicações de Redes de Sensores Aquáticas e sua classificação. Foi observado que um bom número de aplicações é utilizado pelas RSAs, onde eles classificaram as aplicações subaquáticas em cinco classes principais que são elas: monitoramento, desastre, militar, navegação e esportes. As RSAs se tornaram um dos principais focos dos pesquisadores. Se esses aplicativos forem devidamente explorados e atendidos, muitas vidas, tempo e dinheiro podem ser salvos. Embora as RSAs tenham visto uma enorme quantidade de crescimento nos últimos anos, ainda há espaço para contribuições amplas, particularmente nas implantações físicas dos sistemas em grande escala.

Os autores, (COUTINHO AZZEDINE BOUKERCHE, 2017) proporam um novo protocolo de roteamento oportunístico, chamado de EnOR. O EnOR rotaciona a prioridade de transmissão dos nós de próximo salto, considerando a energia residual, confiabilidade dos enlaces acústicos e o avanço do pacote em direção ao destinatário. Visto que apesar de seus imensos benefícios, esses protocolos geralmente introduzem um consumo desigual de energia nos sensores. Isso acontece por causa da atribuição imutável das prioridades de transmissão nos nós sensores, que faz com que alguns nós participem extensivamente do roteamento de dados e esgotem suas baterias rapidamente, causando partições na rede e, conseqüentemente, uma diminuição do tempo de vida útil de uma rede de sensores sub-aquática. Os resultados da simulação mostram que o protocolo proposto, quando comparado com trabalhos relacionados, estende o tempo de vida da rede de sensores sub-aquática.

## 3 REDES DE SENSORES AQUÁTICAS

Neste capítulo, será realizada uma abordagem acerca das RSAs, suas características e diferenças em relação as redes terrestres e uma melhor compreensão sobre o seu meio de propagação. Além disso, serão apresentados sua arquitetura, protocolos de roteamento, aplicações, bem como tecnologias auxiliares.

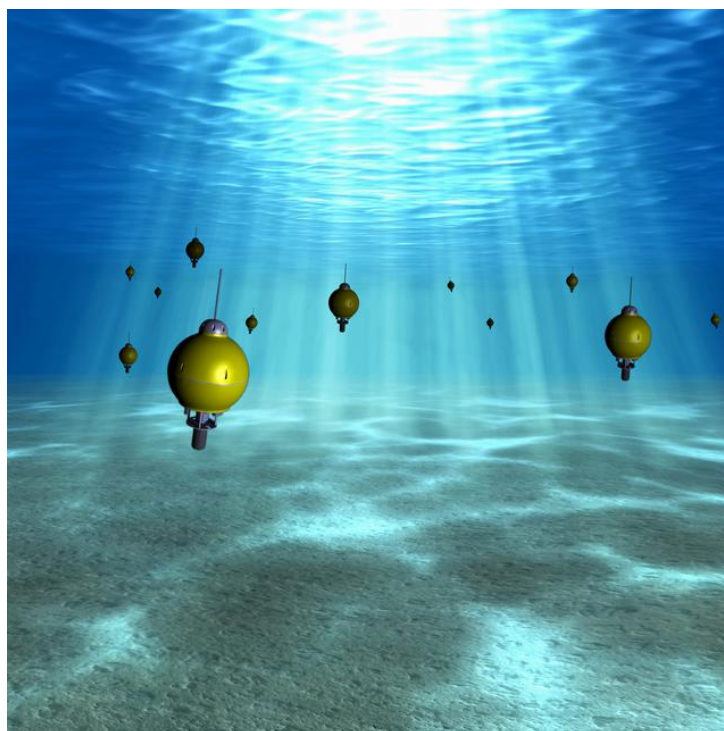
### 3.1 Estado da Arte e Tendências

Uma grande quantidade de recursos preexistentes ficam submersos e devem ser explorados. A chave para explorações bem-sucedidas sempre foi dependente de tecnologia. Avanços recentes em tecnologias levaram as possibilidades de fazer explorações submarinas usando sensores em todos os níveis que não eram possíveis anteriormente. Conseqüentemente, as RSAs estão emergindo como uma tecnologia capacitadora para explorações submarinas. As RSAs são uma fusão da tecnologia sem fio com uma tecnologia de sensores micro-cênica extremamente pequena, com recursos de sensoriamento inteligente, computação inteligente e comunicação. As RSAs são uma rede de nós sensores autônomos que são espacialmente distribuídos submersos para detectar as propriedades relacionadas à água, como qualidade, salinidade, temperatura, e pressão. Os dados detectados podem ser utilizados por uma variedade de aplicações que podem ser usadas para o benefício dos seres humanos. Os nós de sensores sem fio, fixos ou móveis, são conectados através de módulos de comunicação para transferir vários eventos de interesse. A comunicação subaquática é feita principalmente com um conjunto de nós transmitindo seus dados para nodos flutuantes de *gateway* que transmitem os dados para a estação costeira de monitoramento e controle, também chamada de estação remota. Geralmente, os receptores acústicos são usados para comunicação. As ondas acústicas são ondas de baixa frequência que oferecem menor largura de banda, mas possuem comprimentos de onda longos. Por isso, ondas acústicas podem percorrer longas distâncias e são usadas para retransmitir informações em quilômetros.

O ritmo atual de pesquisa na área de redes aquáticas é lenta devido às dificuldades que surgem na transferência das RSAs de última geração para seu equivalente submarino. As implantações subaquáticas dependem da acústica para permitir a comunicação combinada com sensores especiais tendo a capacidade de enfrentar ambientes adversos dos oceanos. No entanto, o sensoriamento e a transmissão subsequente tendem a variar conforme os ambientes do submarino; Por exemplo, a exploração em águas profundas requer uma abordagem completamente diferente para a comunicação, em comparação com a comunicação da água subterrânea. As RSAs são utilizadas para uma ampla gama de aplicações, como monitoramento do ambiente marinho para exploração científica, exploração comercial e proteção da costa, desde a prevenção de catástrofes baseadas na água até à facilitação de desportos aquáticos. As RSAs oferecem uma solução de aprimoramento para aplicações cada vez mais exigentes. No entanto, suas aplicações são excitantes, mas desafiadoras ao mesmo tempo. A razão é que as condições imprevisíveis do

ambiente aquático criam sérias restrições ao projeto e à implantação de tais redes.

**Figura 1 – Redes de sensores aquáticas.**



Fonte: Jaffe Laboratory (2009)

## 3.2 Características do canal acústico submarino

De acordo com (HARRIS A., 2007), o canal acústico submarino é diferente em relação aos meios de transmissão terrestres em vários aspectos, apresentando as seguintes características:

- Velocidade de propagação: a velocidade de propagação do som na água é de 1500 m/s, cinco ordens de grandeza menor do que a velocidade de propagação eletromagnética no ar que é de  $3 \times 10^8$  m/s. Essa baixa velocidade gera uma grande latência nas transmissões acústicas subaquáticas (aproximadamente 0,67 s/km ou 1 segundo a cada 1500 m). Esse é um dos maiores problemas do canal acústico, que gera implicações nos processos de localização e sincronização dos nós na rede. Além disso, a velocidade de propagação do som na água é variável, dependendo da profundidade, da temperatura e da salinidade. A velocidade aumenta com a temperatura cerca de 4 m/s a cada grau celsius, com a salinidade cerca de 1 m/s por cada unidade ppm e 1,7 m/s a cada 100 metros de profundidade;

- Atenuação e ruído: a atenuação total no canal acústico é calculada com base na perda de propagação proposta em (URICK, 1983) e pela aproximação de *Thorp*, apresentada em (BREKHOVSKIKH L. M. E LYSANOV, 1991), para perdas por espalhamento. Estas informações são utilizadas para calcular a relação sinal ruído (SNR) no receptor;

- **Interferência de multicaminhos:** a interferência multicaminhos é causada pela reflexão do sinal acústico no fundo, na superfície ou em algum objeto no oceano, causando a interferência intersímbolos. A interferência intersímbolos no canal acústico é de dezenas a centenas de intervalos, já nas transmissões eletromagnéticas terrestres esta interferência é de alguns intervalos. O som na água pode se propagar em trajetórias curvilíneas e, em alguns casos, pode não atingir determinadas regiões denominadas zonas de sombra, que são locais inalcançáveis por transmissões acústicas. O mesmo fenômeno é responsável pela formação de zonas de convergência, locais onde transmissões provenientes de trajetórias diferentes convergem em tempos diferentes. Nas transmissões digitais esse fenômeno causa o aumento da interferência intersímbolos. O fenômeno da convergência ocorre mais intensamente em águas profundas;

- **Largura de banda:** a largura de banda é extremamente limitada e varia em relação à distância e à frequência.

### 3.3 Diferenças para Redes Terrestres

O projeto das RSAS é significativamente diferente de quaisquer redes terrestre existentes. Quando comparadas às redes terrestres, as redes de sensores aquáticas apresentam diversas distinções. A primeira delas é a restrição da largura de banda, bem menor nas redes aquáticas. Também, ao contrário de enlaces sem fio no meio terrestre, cada enlace aquático apresenta alta latência e baixa largura de banda. A latência chega a ser cinco ordens de grandeza maior que nas redes terrestres. Isso significa que um sinal que demora 1 ms na rede terrestre, gasta 100 ms na rede aquática. A natureza das aplicações também é diferente. Aplicações para o oceano devem considerar o tamanho das regiões para deposição dos nós sensores. Em geral, as regiões de deposição são vastas e a tendência das redes é de possuírem baixa densidade. A alta taxa de erros presente no meio acústico é maior do que nas redes terrestres, dificultando ainda mais a comunicação e requerendo soluções inteligentes para as aplicações e protocolos. As principais diferenças entre redes de sensores terrestres e aquáticas são listadas abaixo:

- **Consumo de energia:** A energia necessária para transmitir ondas acústicas é maior que o de ondas de rádios terrestres devido à diferença da tecnologia na camada física. O alto consumo de energia é necessário devido às longas distâncias e ao processamento de sinal mais complexo, que tenta compensar pela natureza de ruídos do canal aquático.

- **Capacidade de memória:** Nós sensores terrestres são equipados com pequena capacidade de armazenamento para manter o baixo preço dos nós sensores. Nós sensores acústicos não precisam seguir essa tendência. Grandes quantidades de armazenamento podem ser instaladas neles, permitindo *caching* de dados para melhor lidar como canal de comunicação tipicamente intermitente.

- **Nível de correlação espacial:** Redes aquáticas são menos densas que as terrestres, devido ao custo mais elevado e o longo alcance dos modems acústicos. Consequentemente,

a correlação espacial de informações sensoriadas é relativamente pequena. Isto dificulta a agregação de dados, que é tipicamente realizada em redes de sensores para reduzir o volume do tráfego de dados.

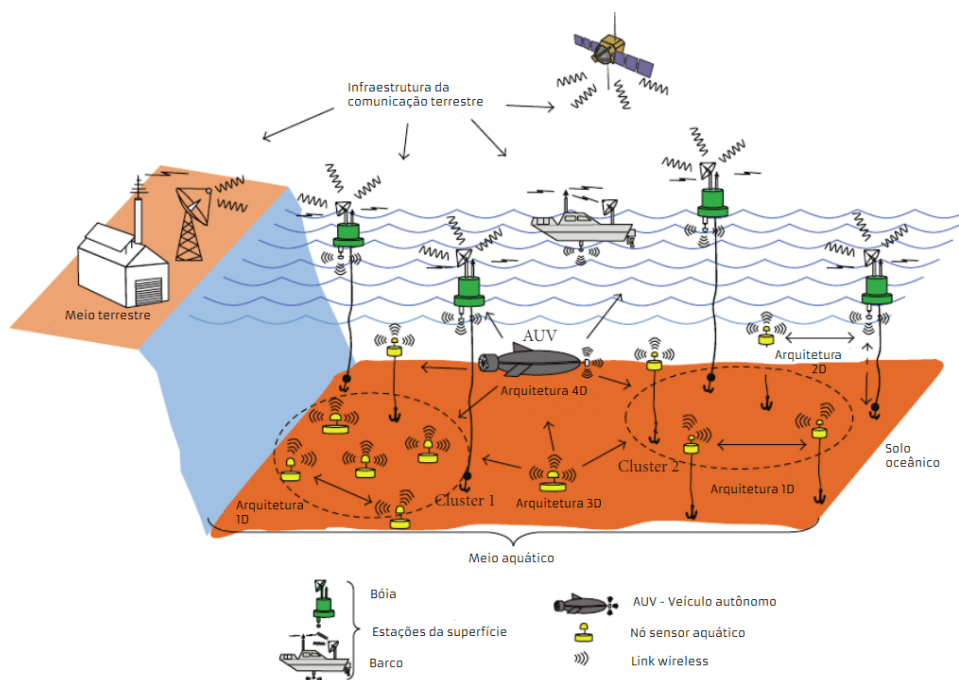
- *Preço do sensores:* Enquanto nós sensores em redes de sensores terrestres têm se tornado mais barato com o avanço da tecnologia, nós sensores acústicos usados em redes aquáticas continuam mais caro. O elevado preço se deve à complexidade dos modems aquáticas e da proteção necessária para ambientes aquáticos extremos. Do ponto de vista econômico, o pequeno número de dispositivos em redes aquáticas também é responsável pelo preço elevado.

Essas diferenças são causadas principalmente devido ao fato que redes aquáticas usam ondas acústicas para comunicação, ao contrário das redes terrestres (AKYILDIZ I. F.; MELODIA, 2005).

### 3.4 RSAs - Arquitetura de Rede

Nesta seção, discutimos as arquiteturas das RSAs mais comuns que são a base para projetar suas respectivas aplicações. Figura 2.

**Figura 2 – Rede de sensor aquática - Arquitetura.**



**Fonte: Próprio Autor (2019)**

#### 3.4.1 Arquitetura 1D

A arquitetura unidimensional (1D) refere-se a uma rede na qual os nós sensores são implantados de forma autônoma. Cada nó sensor é uma rede independente, responsável por

detectar, processar e transmitir as informações para a estação remota. Um nó nesse tipo de arquitetura pode ser uma boia flutuante que pode detectar propriedades submersas ou pode ser implantada debaixo d'água durante um determinado período de tempo para detectar informações e, em seguida, flutuar em direção à superfície para transmitir as informações detectadas à estação remota. Pode ser um veículo subaquático autônomo (AUV) que mergulha dentro da água, detecta ou coleta as propriedades subaquáticas e retransmite as informações para a estação remota. Na arquitetura 1D, os nós podem se comunicar usando comunicação acústica, radiofrequência (RF) ou óptica. Além disso, a natureza topológica de 1D é a estrela, onde a transmissão através do nó sensor e da estação remota é transportada por um único salto.

### 3.4.2 Arquitetura 2D

A arquitetura bidimensional (2D) refere-se a uma rede na qual um grupo de nós sensores (*cluster*) é implantado embaixo d'água. Cada *cluster* possui uma cabeça de *cluster* (também chamada nó âncora). Os aglomerados são fixos quando ancorados na superfície submersa. Cada membro do *cluster* reúne os dados subaquáticos e os comunica ao nó âncora. O nó âncora reúne as informações / dados de todos os seus nós membros e os retransmite para os nós flutuantes da superfície. Na arquitetura 2D, a comunicação é realizada em duas dimensões; isto é, cada membro do *cluster* se comunica com seu nó de âncora com o link de comunicação horizontal, enquanto o nó de âncora se comunica com o nó flutuante de superfície com o link de comunicação vertical. Na arquitetura 2D, a comunicação pode ser acústica, óptica ou RF pode ser usada dependendo do tipo de aplicação e da natureza do ambiente subaquático. Na arquitetura 2D, a comunicação acústica é preferida para o nó de ancoragem submarino e o nó flutuante de superfície devido à distância tipicamente alta entre eles. Para o *cluster* de nós, o arranjo de rede pode ser estrela, malha(*mesh*) ou anel, dependendo do requisito do aplicativo. A arquitetura 2D pode ser usada para aplicações de tempo crítico e tolerantes a atrasos.

### 3.4.3 Arquitetura 3D

Neste tipo de rede, os sensores são implantados debaixo d'água na forma de *clusters* e são ancorados em diferentes profundidades. Devido à implantação dos sensores em alturas variáveis, a comunicação entre os sensores ultrapassa as duas dimensões. Existem três cenários de comunicação nessa arquitetura: (i) comunicação entre nós em diferentes profundidades, (ii) comunicação intracluster (nó âncora-sensor) e (iii) comunicação de nó flutuante-âncora. Nos três tipos de cenários de comunicação, os links acústicos, ópticos e RF podem ser usados.

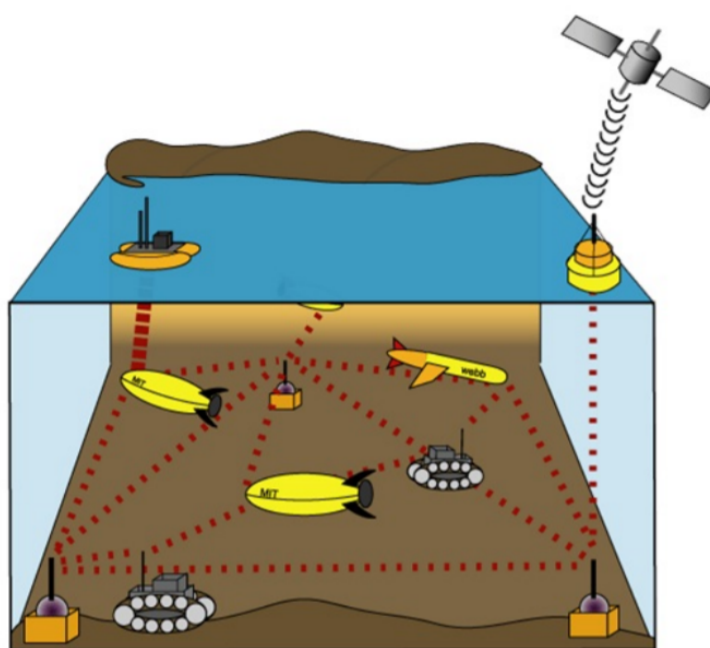
### 3.4.4 Arquitetura 4D

A arquitetura quadridimensional (4D) é projetada pela combinação de RSAs fixos, isto é, RSAs 3D e RSAs móveis. As RSAs móveis consistem em veículos submarinos operando remotamente (ROVs) para coletar dados dos nós âncora e retransmitir os dados para a estação



remota. Os ROVs podem ser robôs submersíveis autônomos, veículos, navios e até submarinos. Cada nó do sensor subaquático pode ser autônomo ao transmitir os dados diretamente para os ROVs, dependendo de quão próximo esse nó do sensor é do ROV. O cenário de comunicação entre o ROV e o nó sensor submerso depende da distância e dos dados entre eles e tanto o acústico quanto a rádio frequência podem ser usados. Como a transmissão deve ser diretamente retransmitida para ROV, os sensores que possuem dados grandes e estão próximos dos ROVs podem usar links de rádio, enquanto os sensores que possuem dados pequenos para transmitir ou estão longe do ROV podem usar links acústicos.

Figura 3 – ROVs.



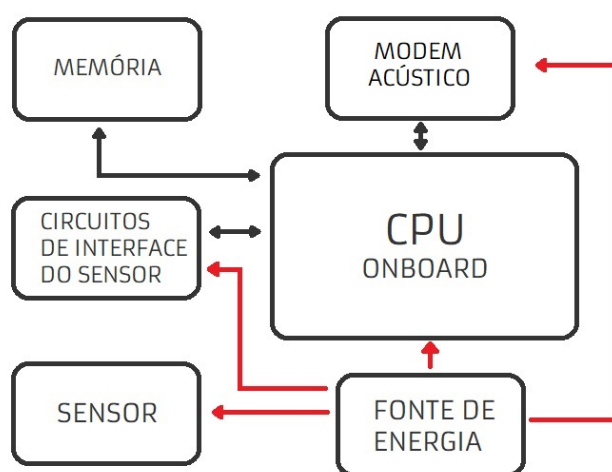
Fonte: (VIEIRA L., 2010)

### 3.5 Arquitetura do sensor

A arquitetura interna típica de um sensor aquático é mostrada na figura 2, Ela consiste em um controlador/CPU que é interfaceado com um instrumento ou sensor aquático através de um circuito de interface do sensor. O controlador recebe dados do sensor e pode armazená-lo na memória *onboard*, processá-lo e enviá-lo para outros dispositivos de rede controlando o modem acústico. Os eletrônicos são normalmente montados em uma estrutura que é protegida por um invólucro de PVC. Às vezes, todos os componentes do sensor são protegidos por estruturas de instrumentos montadas no fundo, que são projetadas para permitir comunicações acústicas omnidirecionais, e protegem sensores e modems do impacto potencial de equipamentos de arrasto, especialmente em áreas sujeitas a atividades de pesca.

Os sensores aquáticos incluem sensores para medir a qualidade da água e estudar suas características, como temperatura, densidade, salinidade (sensores interferométricos e refratométricos), acidez, propriedades químicas, condutividade, pH (sensores magnetoelásticos), oxigênio, hidrogênio, gás dissolvido em metano e turbidez. Existe um sensor descartável que detecta a ricina, a proteína altamente venenosa encontrada na mamona e considerada como agente terrorista de grande potencial. Além disso, sensores de força/torque para aplicações submarinas que exigem medições simultâneas de várias forças e momentos também foram desenvolvidos, bem como sensores quânticos para medir a radiação de luz e sensores para medições de algas nocivas.

**Figura 4 – Arquitetura interna de um nó sensor subaquático.**



Fonte: Próprio Autor (2019)

### 3.6 Noções básicas de propagação aquática

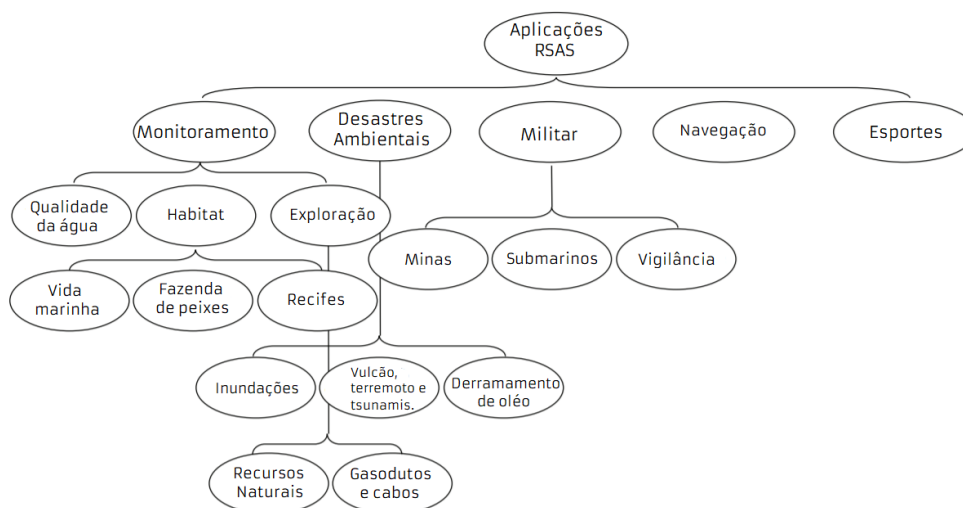
Devido às limitações do canal acústico, como Largura de banda reduzida, atraso, alta atenuação, alta latência e largura de banda limitada alguns requisitos devem ser observados na implementação de aplicações baseadas em redes de sensores com transmissão acústica.

- Taxa de transmissão: na comunicação acústica é limitada pelas características de propagação no meio. Na prática não se deve considerar taxas acima de 10 kbps, que é a taxa útil máxima que se consegue alcançar com os atuais modems acústicos submarinos;
- Distância entre os nós: no canal acústico a banda disponível é inversamente proporcional à distância,

### 3.7 Redes de sensores aquáticas: Aplicações

As aplicações das RSAs estão possibilitando avanços na área de monitoramento e sistemas de observação oceânica, vigilância em águas profundas, rastreamento de várias entidades do ambiente aquático e recursos de exploração de recursos. As RSAs são aplicadas em campos como extração de petróleo, gás, derramamentos de óleo, vigilância militar, detecção de minas, monitoramento da poluição, calamidades naturais como previsão de tsunamis e furacões, monitoramento de habitats da vida marinha, e piscicultura. A classificação abrangente das possíveis aplicações das RSAs é mostrada na Figura 3. Esta seção apresenta uma pesquisa de desenvolvimentos recentes no domínio de aplicações das RSAs. No nível superior, classificamos as aplicações das RSAs como monitoramento, previsão e gerenciamento de desastres, militares, navegação e esportes que terão um impacto significativo no mundo subaquático e fornecerão benefícios aos seres humanos. Depois, classificamos monitoramento, desastres e aplicações militares em subclasses relevantes.

**Figura 5 – Classificação de aplicações de redes de sensores aquáticas.**



**Fonte: Próprio Autor (2019)**

#### 3.7.1 Aplicações de Monitoramento

As aplicações de monitoramento aquático refere-se a uma rede de sensores que é implantada no meio aquático para monitoração do ambiente, suas características, propriedades ou qualquer objeto de interesse. Estas aplicações são particularmente relacionado ao monitoramento do ambiente físico. Aplicações de monitoramento aquático podem ainda ser classificadas em (i) monitoramento da qualidade da água, (ii) monitoramento de habitat e (iii) monitoramento de explorações subaquáticas.

### 3.7.1.1 Qualidade da água

A água é um recurso precioso e é o principal fator para a sobrevivência dos seres vivos abaixo ou acima superfície da água. Portanto, é importante monitorar a qualidade de água. As aplicações de monitoramento de qualidade subaquática variam de monitorar a qualidade da água dos canais até os oceanos. Os autores (YALCUK; POSTALCIOGLU, 2015) em [1] desenvolveram um aplicativo para monitorar a qualidade da água da piscina para fazendas de trutas. Para o crescimento de trutas em uma fazenda/piscina, vários parâmetros foram monitorado, como demanda química de oxigênio, amônia, pH e condutividade elétrica (CE). Os parâmetros foram monitorados por 270 dias entre agosto de 2011 e abril de 2012. Um algoritmo foi proposto pelos autores que pode exibir a informação da entrada e saída de todos as quatro piscinas. A comparação foi feita usando a lógica *fuzzy* para avaliar os dados detectados e notificar, no caso de estado crítico sempre que os parâmetros ultrapassem o limiar valores. Em [2], os autores (TUNA; GULEZ, 2013) desenvolveram duas abordagens baseados em RSAs para monitorar a qualidade da água na água potável em reservatórios. A primeira abordagem é baseada no uso de sensores integrados com um *mini-AUV* responsável por coletar as amostras de água e transmissão da informação para a estação remota. A segunda abordagem consiste em sondas montadas com painéis solares usando a interface sem fio baseada em IEEE 802.15.4 para coletar e transmitir a informação. Foi demonstrado que a segunda abordagem provou para ser mais aplicável, eficiente e de baixo custo para monitoramento da qualidade da água.

### 3.7.1.2 Habitat

Monitoramento de habitats subaquáticos lida com o estudo do ambiente de qualquer organismo vivo que na água. É um dos campos mais interessantes e desafiadores das ciências naturais. O monitoramento de habitat, quando considerado submerso, torna-se ainda mais desafiador devido às condições subaquáticas vulneráveis.

Os autores (ALIPPI R. CAMPLANI; ROVERI, 2011) em [3] desenvolveram uma estrutura para monitoramento aquático. A estrutura tem capacidade de monitorar o ambiente marinho em diferentes níveis, cujos dados podem ser usados para desenvolver e verificar modelos de ecossistemas, prever a possibilidade de qualquer mudança nos fenômenos ecológicos e também adaptar-se a quaisquer mudanças climáticas ambientais/subaquáticas. O sistema possui recursos de detecção, transmissão sem fio para estação remota, visualização e alarme sobre os eventos. O sistema está atualmente implantado em Queensland, na Austrália. Os autores projetaram e desenvolveram um sistema ad hoc, baseado em *clusters* (topologia em estrela) onde a cabeça do *cluster* transmite dados diretamente ao gateway flutuante. O sistema proposto foi utilizado para monitorar a luminosidade subaquática (LUM) e temperatura.

### 3.7.1.3 Exploração submarina

Há um grande número de minerais presentes debaixo d'água que é necessário para ser explorado, como petróleo e gás. Além disso, é preciso encontrar a extremidade da água enquanto une uma extremidade da superfície a outra. Enquanto a água está cobrindo a grande parte da superfície da Terra, a superfície seca restante da terra é conectada colocando os cabos embaixo d'água. São implantados cabos submarinos que trazem algumas das necessidades sociais mais básicas, como petróleo, gás oleodutos e cabos de fibra ótica. Portanto, as RSAs podem ser usadas para explorar os recursos preciosos e também para monitorar o encanamento e os cabos submarinos.

Uma RSA com transmissão acústica proposta pelos autores (ACAR; ADAMS, 2006) em [4] também é usada para exploração em mar profundo e monitoramento de áreas costeiras. O sistema é especificamente testado para aplicação costeira por implantar câmeras subaquáticas. A literatura detalhada está presente, motivando os pesquisadores a desenvolver um aplicativo para explorações e aplicações em águas profundas. A literatura apresenta a arquitetura de comunicação e possível design do sistema e destaca condições e desafios adequados. As aplicações de monitoramento das RSAs estão resumidas na Tabela 1.

**Tabela 1 – Aplicações de monitoramento**

Autores	Arquitetura	Desenvolvimento	Comunicação
(1)	1D	Piscina	RF
(2)	3D, 4D	Reservatório	RF, Acústica
(3)	3D, 4D	Mar	Acústica
(4)	3D	Mar	Acústica

**Fonte: Próprio Autor (2019)**

Pode-se observar que, quando se trata de monitoramento da qualidade da água, a maioria das aplicações recentes tem usado RF para alcançar seus objetivos. Considerando qualquer tipo de impurezas que contaminem a água, como derramamentos de óleo, elas estão em área restrita de água; portanto, devido as distâncias, os sinais RF preenchem os requisitos. No entanto, há uma ênfase igual no uso de RF e ondas acústicas como o tipo de comunicação. Como a RF tem limitações em relação à distância, o acústico tem sido considerado para distâncias maiores, que variam em alguns quilômetros. A acústica é empregada principalmente para a vida marinha, recifes e aplicações de exploração em mar profundo, uma vez que é necessária uma maior cobertura. Uma tendência de empregar várias arquiteturas também é vista quando se lida com essas aplicações, enquanto, para aplicações de vida marinha, recife e exploração, as arquiteturas 3D e 4D são mais preferidas devido a diferentes profundidades e distâncias dos objetos sob consideração.

### 3.7.2 Aplicações de prevenção de desastres

Geralmente, os desastres naturais são inevitáveis. Entre outros, os desastres naturais à base de água são mais perigosos e produzem uma enorme destruição para a Terra. Consequentemente, os mecanismos preventivos e de monitoramento de desastres são muito necessários. As RSAs oferecem uma ampla gama de aplicativos para gerenciamento e recuperação de tais desastres. Mais particularmente, refere-se ao monitoramento de eventos que agravam os desastres. Juntamente com recursos inadequados para o monitoramento abrangente da vasta área de água (por exemplo, oceano), a tarefa se torna ainda mais desafiadora com o clima ocasionalmente implacável. Portanto, o monitoramento eficiente da dinâmica marinha e aquática é um desafio significativo de pesquisa. As estratégias de monitoramento da RSA para gestão e prevenção de desastres podem ser formuladas em uma ampla variedade de aplicações, como inundações, erupções vulcânicas submarinas, terremotos submersos e seus tsunamis resultantes, e derramamentos de petróleo que levam a instabilidades ecológicas subaquáticas e acima d'água.

#### 3.7.2.1 Inundações

As repercussões de uma inundação e seu aumento da frequência levaram os pesquisadores a encontrar formas de alertas de enchente oportunos. Os alertas não precisam ser colocados apenas em áreas urbanas e, portanto, requerem implantação remota. Uma RSA ajuda a desenvolver soluções de implantações de sensores submarinos com agentes de transmissão sobre a água para calcular os sinais vitais aquáticos.

Estes sinais vitais são recolhidos na estação remota e inspecionados para indicações de inundação. (PASI; BHAVE, 2015) abordaram a questão do monitoramento de inundações e alarmes com a ajuda de RSAs [5]. O sistema projetado consiste em um módulo sensor, um módulo observatório e um módulo transponder. O módulo sensor é responsável pelo monitoramento da água e pela coleta de parâmetros relacionados à água, como nível, empuxo e intensidade da água, como indicadores de inundação. A informação é transmitida para a estação remota para observações adicionais. O objetivo do módulo observatório é observar as informações e prever a inundação. O módulo transponder é usado para retransmitir a informação em caso de inundação. O sistema projetado é simulado e testado na perspectiva de sua eficiência, variando o número de nós implantados em relação ao tempo de atraso. O sistema também é testado quanto ao seu erro de localização e à área de cobertura.

(TYAN; OH, 2013)[6] propuseram uma RSA baseada em acústica para monitoramento de inundações em rios. O sistema projetado é baseado na arquitetura 4D, que consiste em sensores submersos, AUV e uma estação remota. O sensor transmite a informação acusticamente ao AUV que recolhe a informação e transmite a informação à estação remota. Os autores testaram e implementaram com sucesso a rede em tempo real.

### 3.7.2.2 Vulcão, terremoto e tsunami

Terremotos e vulcões subaquáticos são desastres naturais e causam risco para os seres vivos. Essas calamidades naturais podem ocorrer em qualquer lugar e em qualquer lugar da superfície da Terra e são ainda mais alarmantes quando ocorrem debaixo d'água, dependendo das mudanças sísmicas e geológicas que ocorrem sob a terra. Portanto, é importante monitorar essas condições. (KUMAR; SRIJA, 2012)[7] discutiram a arquitetura das RSAs e propuseram a 4D para geração de alerta antecipado em caso de qualquer evento perigoso, como terremotos e tsunamis. (CASEY; DOZIER, 2008) [8] propuseram uma arquitetura eficiente para a detecção de tsunamis. A arquitetura baseada em sensores proposta utiliza sensores de pressão sísmica para prever o tsunami subaquático e o relé detectado informado pelo protocolo de roteamento por difusão direcionada. A arquitetura funciona no mecanismo de sentido e resposta. Os autores detalharam várias abordagens e testaram a rede proposta em um ambiente de teste.

### 3.7.2.3 Derramamento de óleo

A poluição causada pelo homem é um fator importante a considerar quando se fala em saúde da vida marinha. A vida marinha é altamente afetada pela poluição causada por derramamentos de óleo e, portanto, as RSAs contribuíram com uma maneira de descobrir a localização e a espessura dos derramamentos de óleo na água, o que pode acelerar o procedimento de limpeza. Em [9], os autores (KHAN; JENKINS, 2008) criaram uma RSA ad hoc que detecta a poluição oceânica. Eles especificaram diferentes sensores, seus algoritmos de sincronização, protocolos de roteamento e a pilha completa de protocolos. O trabalho se concentra em maximizar a vida útil da rede e *QoS* (Quality of service). (IWENDI; ALLEN, 2011) em [10] demonstraram as questões de segurança no Setor de Petróleo e Gás do Níger-Delta, onde trabalharam especificamente no desenvolvimento de uma rede que possui um esquema de distribuição de chaves muito seguro. Além do sistema de gerenciamento de chaves, o trabalho também se concentra em algoritmos de clustering rápidos e robustos com sistema de alocação dinâmica de chaves.

**Tabela 2 – Aplicações de prevenção de desastres**

Autores	Arquitetura	Desenvolvimento	Comunicação
(5)	2D	Canais	RF
(6)	4D	Rio	Acústica
(7)	4D	Mar	Acústica
(8)	3D	n/a	Simulação
(9)	3D	Mar	Acústica
(10)	4D	n/a	Simulação

**Fonte: Próprio Autor (2019)**

## 3.8 Camada de Enlace

Esta seção descreve os desafios e dificuldades na camada de enlace, bem como apresenta os resultados mais recentes nessa área de pesquisa e os protocolos mais recentes. A seguir, são descritos os protocolos baseados em partição: na frequência (FDMA), no tempo (TDMA), e via código (CDMA). Depois serão descritos os protocolos baseados em acesso aleatório como Aloha e CSMA. Finalmente, serão apresentados os protocolos baseados em reserva do meio (RMAC) e escalonamento.

Protocolos MAC servem para controlar o acesso ao meio comum compartilhado por múltiplos usuários. O objetivo dos protocolos MAC é maximizar a eficiência do e melhorar o a justiça *fairness*. Embora protocolos MAC em redes de pacotes de rádio terrestre tenham sido intensamente estudados, eles não podem ser aplicados diretamente às redes de sensores aquáticas devido à dificuldade da natureza do canal de comunicação aquático. Portanto, é imperativo que o desenvolvimento de novos protocolos MAC considerem especificamente as características únicas do meio aquático (PARTAN J.; LEVINE, 2006)

Nas RSAs, o espectro disponível é severamente limitado. O problema é fortemente exacerbado devido à velocidade de propagação do som que é cinco ordens de grandeza mais lenta que a baseada em ondas de rádios na superfície terrestre. Dessa forma, esses recursos limitados devem ser compartilhados de forma eficiente e de forma confiável por meios de um protocolo MAC adequado. Nesta seção, os protocolos MAC existentes são categorizados e são discutidas suas vantagens e desvantagens em redes aquáticas. Os protocolos MAC abordados são divididos em três categorias: baseados em partição, acesso aleatório ao meio e baseados em reserva e escalonamento

### 3.8.1 Protocolos Baseados em Partição

Na literatura, há três correntes de pesquisa em protocolos baseados em partição: acesso múltiplo dividido pela frequência (FDMA, do inglês frequency division multiple access), acesso múltiplo dividido pelo tempo (TDMA, do inglês time division multiple access), e acesso múltiplo dividido codificado (CDMA, em inglês code division multiple access). Todos os três tipos de protocolos são livres de colisões e o acesso ao canal é justo.

#### 3.8.1.1 FDMA

FDMA é um tipo de protocolo MAC baseado em partição que divide a frequência de banda disponível em múltiplas sub-bandas e assinala cada sub-banda a um nó individual. Em geral, as sub-bandas precisam de bandas de guarda entre elas que compensem filtros imperfeitos, interferência de canais adjacentes e *spectral spreading* devido ao efeito Doppler. A maior vantagem do FDMA é sua simplicidade algorítmica e eficiência quando lidando com pequeno



número de nós em um tráfego constante uniforme. No entanto, este esquema não é aplicável em redes aquáticas porque a frequência de banda disponível no canal acústico aquático é pequena.

### 3.8.1.2 TDMA

Em protocolos TDMA, um intervalo de tempo é dividido em múltiplos períodos de tempo onde cada período é unicamente assinalado a um nó individual. Nesse esquema, os dados são colocados em um buffer até que chegue a vez de seu período de tempo, quando os dados são transmitidos. Isso pode levar a transmissões em rajadas. Infelizmente, transmissões em rajadas necessitam uma taxa alta de transmissão, quando comparados ao FDMA, e aumenta a interferência entre símbolos (ISI, do inglês Inter Symbol Interference). Por esse motivo, equalizadores adaptativos são usualmente necessários em sistemas TDMA (YACKOSKI; SHEN, 2008).

Como o modem de cada nó sensor é o mesmo, o número de períodos de tempo assinalados a um nó pode mudar sem precisar de hardware adicional. Isso provê ao sistemas TDMA uma flexibilidade. Dessa maneira, a taxa de transmissão dos nós sensores pode aumentar durante o ciclo ocioso (PAHLAVAN; LEVESQUE, 2005). Da mesma forma, o transmissor pode ser desligado por períodos de descanso, prolongando o tempo de vida da bateria. Infelizmente, o TDMA requer sincronização de tempo precisa entre todos os nós da rede. Isso implica que todos os nós devem ter mecanismos instantâneos de comunicação para permanecerem sincronizados durante falhas dos nós ou movimento dos nós para manter o escalonamento de transmissões. Entretanto, em comunicação acústica, devido à grande latência na propagação do sinal, cada período de tempo deve ser mantido longo o suficiente para evitar colisões. Isso pode levar a longos períodos de tempo sem comunicação, prejudicando severamente a vazão.

### 3.8.1.3 CDMA

Protocolos CDMA permitem que múltiplos nós operem concorrentemente sobre toda a frequência de banda. O acesso ao canal de nós diferentes é baseado em um código único que é usado para espalhamento. Existem duas técnicas básicas de espalhamento, DSSS (do inglês direct sequence spread spectrum) e FHSS (do inglês frequency hopped spread spectrum).

No caso do DSSS, sinais de informação são linearmente modulados usando códigos de banda larga, enquanto no FHSS as frequências portadoras dos nós são modificados de acordo com um padrão obtido a partir dos códigos (PAHLAVAN; LEVESQUE, 2005). Um único código é assinalado a cada nó para modulação. Enquanto o CDMA é robusto ao desvanecimento da frequência selecionada, beneficia de múltiplas recepções simultâneas, e pode compensar pelo efeito de múltiplos caminhos através de filtros Rake (POMPILI D.; AKYILDIZ, 2009).

Porém o CDMA é vulnerável ao problema perto–longe1 (também conhecido como near-far problem) (FLIKKEMA, 1997). Adicionalmente, um algoritmo de controle de potência é necessário para reduzir o nível da potência de saída de cada nó para estabelecer transferência de

pacotes confiável sem criar interferência excessiva e para aliviar o consumo excessivo de bateria em transmissões aquáticas.

(POMPILI D.; AKYILDIZ, 2009) propôs um algoritmo distribuído para um protocolo CDMA que obtém a transmissão ótima de potência e tamanho do código. Resultados de simulação mostram que o protocolo apresenta uma melhora em relação a protocolos MAC existentes sintonizados para águas rasas, em diferentes cenários de simulação. Entretanto, vários problemas precisam ser resolvidos, como separação de sinal e a atribuição dos códigos. De acordo com os autores, CDMA e sinalização de espalhamento de espectro são técnicas promissoras para múltiplos acessos em redes em águas rasas (profundidade menor que 100 m).

### 3.8.2 Protocolos Baseados em Acesso Aleatório

Estes tipos de protocolos são baseados em acesso aleatório. Este esquema não divide os recursos limitados do canal, mas permitem que nós acessem o meio baseado no princípio da contenção. Assim, esses tipos de protocolos aumentam a utilização do canal, mas não garantem acesso livre de colisão entre os nós

#### 3.8.2.1 ALOHA

O protocolo Aloha original é baseado em puro acesso aleatório ao meio. Quando um nó tem informação a enviar, ele transmite a informação imediatamente. Uma confirmação (ACK) é enviada de volta pelo receptor se o pacote for recebido corretamente.

Se a colisão ocorrer, devido à transmissão concorrente de dados ou ACKs no receptor e transmissor respectivamente, o transmissor retransmite o mesmo pacote. Devido à contenção e retransmissões, a vazão máxima alcançável pelo Aloha original é de 18 por cento (VIEIRA L. F. M.; GERLA, 2006)

(ROBERTS, 1975) propuseram dois protocolos distribuídos baseados no Aloha para comunicação aquática: Aloha com Collision Avoidance (Aloha-CA) e Aloha com Advance Notification (Aloha-AN). Nesses dois protocolos, cada nó tenta utilizar a informação sobre o transmissor e o receptor, escutando outras transmissões, para ajudar a evitar colisões e obter um desempenho de vazão melhor.

No Aloha-CA, cada pacote é diferenciado em dois segmentos distintos, um cabeçalho e um segmento de dados. Baseado na informação de transmissor-receptor obtida escutando a transmissão, cada nó calcula a duração que o meio estará ocupado da transmissão em andamento e decide se esse estado ocupado é causado pela transmissão, recepção ou escuta de pacote. Essa decisão ajuda cada nó a aumentar a vazão e reduzir a chance de colisão de pacotes. Esse esquema não requer sincronização porque cada nó mantém informação localmente na sua tabela de dados com respeito ao seu próprio relógio. Reduzindo o tamanho do cabeçalho, o protocolo Aloha-CA reduz o tempo necessário para obter informações úteis e diminui a possibilidade de colisão

de pacotes. Entretanto, colisões ainda são possíveis porque a tabela é mantida somente com a informação que o nó escutou anteriormente, que é meramente um subconjunto de toda a rede e que é necessário para decisões livres de colisões.

O protocolo Aloha-AN modifica o Aloha-CA. Cada nó primeiro transmite seu segmento de cabeçalho como um pequeno pacote de notificação avançada (NTF). O transmissor irá esperar por um período de tempo, denominado lag time, antes de enviar um pacote de dados. Cada nó na rede, que escuta o NTF, checa se o pacote de dados associado irá causar conflito com seu próprio pacote de dados escalonado para transmitir ou receber, e calcula o tempo esperado de transmissão. Se existe qualquer conflito com sua transmissão ou com nós vizinhos, o mecanismo de resolução é invocado. Se um conflito é detectado entre dois nós, o mecanismo escolhe o nó com o escalonamento mais cedo para transmitir primeiro e o outro aguarda. O nó que aguarda, tem que transmitir um novo pacote NTF.

A diferença chave entre os dois mecanismos é que o Aloha-AN já considera a transmissão de dados escalonada obtida via NTF quando escalonando a própria transmissão. Se um novo escalonamento entra em conflito com o escalonamento existente, o escalonamento pode ser posto em espera. Portanto, quando comparado com o Aloha-CA, o NTF dá ao nó um subconjunto maior do estado da rede, permitindo tomar uma decisão melhor para evitar colisões.

O custo adicional inserido na transmissão do pequeno pacote NTF é justificado pelos casos de colisão de pacotes que iriam ocorrer caso o mecanismo de informação adicional não existisse. Esse mecanismo não é totalmente livre de colisão. Ele também não garante justiça, visto que um nó pode continuamente enviar pacotes NTF para capturar o canal indefinidamente, enquanto tiver dados para transmitir.

Finalmente, esse algoritmo não pode realizar ciclos ociosos para economizar energia porque a informação adicional é coletada escutando o canal, requerendo que o nó permaneça ligado escutando os pacotes NTF.

### 3.8.2.2 CSMA

Recursos limitados do canal podem ser melhor utilizados se os nós sensoriam as portadoras antes de transmitir. O método de acesso ao meio é baseado nesta idéia e é chamado CSMA (carrier sense multiple access) (PAHLAVAN; LEVESQUE, 2005).

O método CSMA tenta evitar colisões ao escutar o meio na vizinhança do transmissor. Nesta abordagem, apenas colisões no lado do transmissor são tratadas, o receptor ainda pode sofrer colisões devido ao problema do terminal escondido (BHARGHAVAN V.; ZHANG, 1994). Detalhes e variantes desse método podem ser encontrados em (SMITH S.; NEEL, 1997).

O protocolo MACA proposto por (KARN, 1990) para detectar colisões no receptor é uma alternativa ao CSMA. Este protocolo introduz troca de mensagens ao adicionar dois pacotes de controle chamados RTS (request-to-send) e CTS (clear-to-send). Quando um nó quer

enviar mensagem a outro, primeiro ele envia um RTS que contém o tamanho da mensagem que será enviada. Se o destinatário receber o RTS, ele responde com um CTS que também tem o comprimento da mensagem. Logo que o transmissor recebe o CTS, ele inicia a transmissão de dados. Qualquer nó que escuta o CTS deixa de transmitir pelo tempo necessário para evitar colisão com o pacote sendo transmitido.

Se um nó vizinho escuta o RTS mas não o CTS, ele infere que está fora do alcance do receptor e transmite o próprio pacote. Este protocolo confia na simetria do canal; o CTS deve ser escutado por todos os nós dentro do alcance do nó receptor. Isso implica que o controle de potência não pode ser usado com estes tipos de protocolos.

Fullmer (FULLMER; GARCIA-LUNA-ACEVES, 1995) propôs FAMA, que estende a duração dos pacotes de controle RTS (maior que o máximo de atraso na propagação) e CTS (comprimento do RTS mais duas vezes o máximo da propagação, mais tempo de transição do hardware para transmitir/receber) para prevenir alguns tipos de colisões. Para adaptar este protocolo para redes aquáticas, (MOLINS, 2006) propôs o protocolo Slotted FAMA. O princípio do algoritmo Slotted FAMA é baseado no algoritmo FAMA. A diferença primária é que o canal é dividido em períodos de tempo e pacotes (RTS, CTS, DATA or ACK), que são enviados apenas por nós no começo de um intervalo de tempo. Se um nó quer enviar um pacote, ele tem que esperar o começo do próximo intervalo de tempo para iniciar o algoritmo FAMA. Esta técnica de divisão do tempo em intervalos é provada em (ROBERTS, 1975) que minimiza as chances de colisão e, portanto, aumenta a vazão. Ao contrário do FAMA, o receptor do pacote pode imediatamente encaminhar o pacote sem ter que esperar um tempo de contenção, permitindo o slotted FAMA alcançar uma melhor eficiência.

No entanto, como mencionado no caso do TDMA, sincronização precisa entre todos os nós da rede é difícil de ser implementada em redes aquáticas. Além disso, esse protocolo tem bom desempenho apenas com um pequeno número de nós, visto que a vazão diminui com o aumento do número de nós.

## 3.9 Protocolos abordados neste trabalho

Os protocolos utilizados nas redes acústicas submarinas devem ser robustos o suficiente para suportar os elevados níveis de interferência e ruído do meio. Alguns protocolos MAC tais como o Broadcast MAC e o R-MAC podem ser utilizados tanto em redes terrestres quanto em redes submarinas devido principalmente a simplicidade e eficiência de ambos.

### 3.9.0.1 R-MAC: Protocolo MAC Baseado em Reserva

R-MAC é um protocolo tipo CSMA baseado em escalonamento (XIE; CUI, 2007b). Ele alcança eficiência de energia usando escuta periódica e modos de dormir para reduzir a energia gasta em estados ociosos. Todos os nós usando este protocolo têm a mesma duração de

períodos para escutar e dormir, mas aleatoriamente selecionam seu escalonamento de tal forma a ter eficiência de energia e justiça de acesso ao meio. R-MAC compreende três fases: detecção de latência, anúncio do período, e operação periódica. As duas primeiras fases sincronizam cada nó com seus vizinhos e a terceira realiza operações de escutar/dormir e comunicação de dados. Um nó na fase de detecção de latência detecta a latência de propagação para todos os seus nós vizinhos. Na fase de anúncio do período, cada nó aleatoriamente escolhe seu próprio escalonamento para escutar/dormir e propaga essa informação. Os dados são transmitidos na fase periódica de operação.

Durante a transmissão de dados, os nós comunicam através de mensagens dos tipos REV/ACK-REV/DATA/ACK-DATA. Quando um nó possui dados para transmitir, primeiro ele envia um REV (pacote de reserva) para fazer a reserva do intervalo de tempo no receptor. Uma vez que o pretendido receptor está pronto para receber os dados, ele notifica seus vizinhos, assim como o transmissor, enviando uma mensagem do tipo ACKREV, reconhecendo que está pronto para receber. Ao receber o pacote ACK-REV, todos os outros, com exceção do transmissor, ficam em silêncio, enquanto o transmissor envia dados no intervalo de tempo. O receptor envia de volta uma mensagem ACK-DATA para o transmissor no fim da transmissão para liberar o meio.

Este mecanismo de escalonamento baseado em reserva permite aos nós dormirem, economizando energia e resolvendo o problema do terminal escondido presente em protocolos baseados em RTS/CTS. Entretanto, este protocolo falha em topologias dinâmicas e móveis. Embora este protocolo seja baseado em escalonamento, desvios no relógio do sistema não são considerados no trabalho original.

### 3.9.1 Protocolos de roteamento

Os protocolos de roteamento utilizados nas redes acústicas submarinas devem ser robustos, escaláveis e utilizar energia com eficiência. Estes requisitos são imprescindíveis para o funcionamento das redes. Protocolos de roteamento proativos e reativos, utilizados nas redes terrestres, não devem ser utilizados nas redes acústicas submarinas. Isto porque o primeiro tipo de protocolo de roteamento realiza atualizações constantes das rotas consumindo muita energia e o segundo tipo, devido ao atraso de propagação do meio, leva muito tempo para encontrar as rotas. Protocolos de roteamento geográfico que associam a localização geográfica do sensor a sua profundidade são os mais indicados para redes de sensores subaquáticas. Um nó sorvedouro, localizado na superfície ou a uma determinada profundidade, é a referência geográfica de toda a rede e todos os demais nós atualizam suas posições em relação a ele através da troca periódica de pacotes e do cálculo das distâncias a partir do atraso de propagação.

O DBR (Depth Based Routing) (YAN H., 2010), assim como a maioria dos protocolos de roteamento acústico, considera a arquitetura geral de uma rede submarina de sensores, composta por um nó sorvedouro (sink) localizado na superfície e por nós submersos. Baseando-se nas informações de profundidade de cada um dos nós sensores, o DBR realiza o encaminhamento dos

pacotes enviados pelos nós submersos até o nó sorvedouro. O pacote de dados do DBR possui um campo que registra as informações de profundidade do último transmissor do pacote, que são atualizadas a cada salto. Quando um nó recebe um pacote ele o encaminha se a profundidade do próprio nó for menor do que a profundidade registrada no pacote, caso contrário o pacote é descartado. Os nós localizados em maiores profundidades enviam para os nós localizados em menores profundidades até que as informações cheguem ao nó sorvedouro. Para a determinação das distâncias entre os nós utiliza-se o protocolo TOA (Time of Arrival), que calcula, através de algoritmos, a distância entre os nós considerando o atraso de propagação. O nó sorvedouro, localizado em uma boia na superfície, estabelece a referência geográfica da rede através de GPS.

O protocolo VBF (XIE P., 2006) opera de forma escalável e com eficiência energética e apresenta bons resultados em redes acústicas submarinas. No VBF cada pacote carrega as informações do nó fonte (source), do nó destino (target / sink) e do nó transmissor (forwarder). Uma rota de transmissão é estabelecida pelo vetor de roteamento entre o nó fonte e o nó de destino considerando a localização de cada um dos nós. Ao receber um pacote um nó calcula sua posição relativa ao transmissor medindo a distância e o ângulo de chegada do sinal. Se o nó verifica que está suficientemente próximo do vetor de roteamento predefinido ele inclui suas próprias informações de localização no pacote e o encaminha, passando a fazer parte da rota de roteamento para aquele destinatário, caso contrário simplesmente descarta o pacote. Pode-se considerar que o caminho de transmissão, estabelecido pelo vetor de roteamento, forma um tubo virtual de roteamento que vai do nó fonte ao destino. Os nós sensores localizados dentro da área do tubo são utilizados para realizar a transmissão, já os nós fora do tubo não transmitem.

## 4 METODOLOGIA

### 4.1 Recursos Utilizados

Esta seção descreve os materiais e métodos utilizados para o desenvolvimento do trabalho e será analisada a viabilidade de aplicações genéricas utilizando redes acústicas submarinas. O desempenho do protocolo R- MAC em três topologias básicas foi avaliado utilizando-se como métricas a vazão, atraso total e a média do consumo de energia. O objetivo principal foi definir qual topologia de rede é mais apropriado para cada tipo de aplicação.

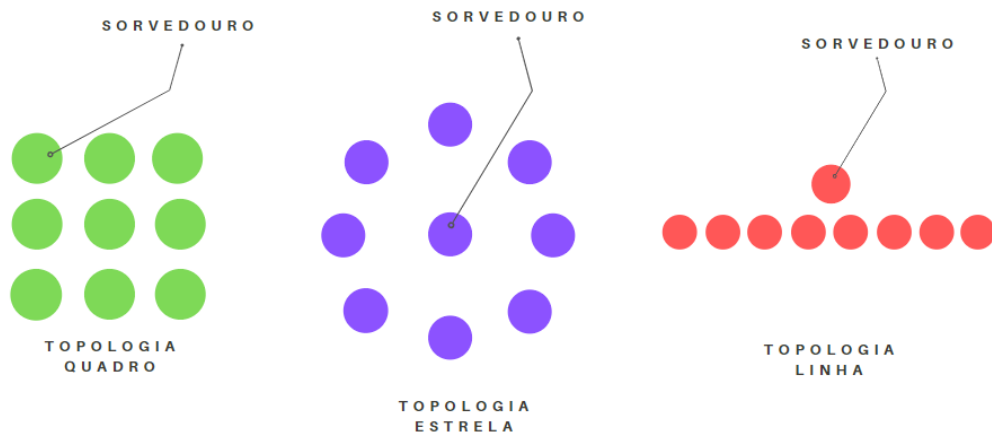
### 4.2 Implementação

Para a realização das simulações foi utilizado o simulador NS-2 e o módulo *Aqua-sim* (XIE P., 2009) que efetivamente simula a atenuação no canal acústico submarino, os atrasos e as colisões na rede. Este simulador suporta nova camada MAC, novos protocolos que são úteis no roteamento subaquático. Serão utilizadas diferentes topologias com o objetivo de auxiliar o desenvolvimento de aplicações baseadas neste tipo de rede. Além disso possui a capacidade de simular redes em duas ou três dimensões, reproduzindo a transmissão no meio subaquático com uma boa fidelidade. O *Aqua-sim* é um módulo independente desenvolvido para rodar no NS-2 (<http://isi.edu/nsnam/ns/>).

Neste estudo são utilizadas as topologias quadro, estrela e linha, propostas como topologias básicas para as redes acústicas submarinas, conforme mostrado na Figura (6). Cada topologia é composta por nove nós, sendo oito nós transmissores e um nó sorvedouro (*sink*), que realiza a concentração dos pacotes, a interface e o encaminhamento das informações para uma rede na superfície. Nas simulações consideramos que a conexão do nó sorvedouro com a superfície é realizada através de um cabo. O nó sorvedouro está localizado a uma profundidade de 50 metros em relação à superfície e a distância ( $d$ ) entre nós adjacentes é de 50 metros.

Devido às limitações do canal acústico, alguns requisitos devem ser observados na implementação de aplicações baseadas em redes de sensores com transmissão acústica. As análises se basearam em simulações realizadas nas três topologias propostas. Para o roteamento foi utilizado o protocolo *VBF* e para acesso ao meio foi avaliado o protocolo R-MAC. Os sensores foram configurados para transmitir pacotes de 60 bytes considerados suficientes para a transmissão das informações coletadas em diversas aplicações o tipo de tráfego utilizado foi o *CBR* (Constant Bit Rate). Cada simulação foi repetida dez vezes e a partir dos dados obtidos foram calculados os intervalos de confiança para a vazão, atraso e consumo total de energia das redes.

Figura 6 – Topologias



Fonte: Próprio Autor (2019)

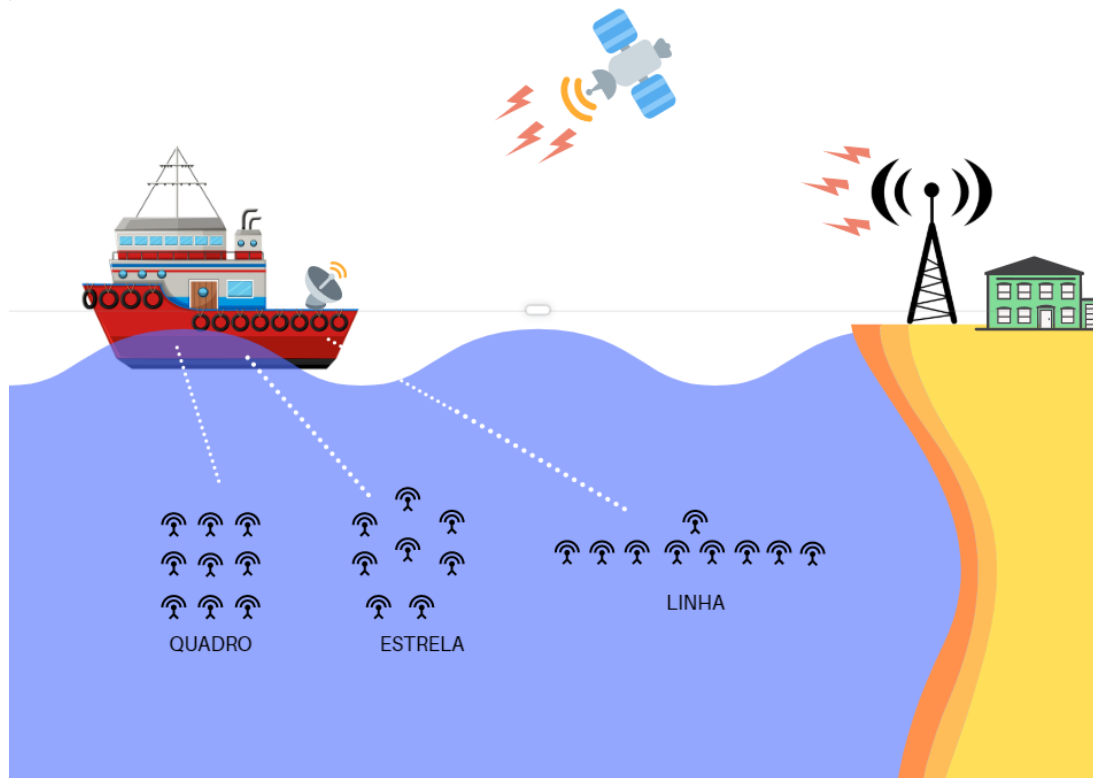
### 4.3 Cenário

Para o cenário (Figura 7), as aplicações são associadas a uma das três topologias básicas propostas e ao protocolo R-MAC, analisados com o objetivo de auxiliar os desenvolvedores das aplicações. A definição da melhor topologia de rede para uma determinada aplicação se baseou na área de cobertura necessária e na necessidade da implementação em duas ou três dimensões. Em uma aplicação de controle de robôs e veículos submarinos, por exemplo, a topologia mais indicada é a estrela que possibilita o controle dos dispositivos em uma determinada área e em três dimensões. Já para a aplicação de monitoramento de desastres (tsunamis), a topologia mais indicada é a linha devido a necessidade de se monitorar grandes distâncias a uma determinada profundidade, caracterizando uma topologia em duas dimensões. Já a topologia quadro é indicada para aplicações de monitoramento militar que necessitam monitorar grandes distâncias e diferentes profundidades.

As aplicações de monitoramento oceanográfico e ambiental e de exploração submarina, por exemplo, não geram informações constantemente já que os parâmetros oceanográficos monitorados não variam com muita rapidez, podendo levar dias ou horas. Apesar disso as informações coletadas devem ser transmitidas de forma confiável e os sensores devem ter uma grande autonomia de energia. Por isso neste caso o protocolo mais indicado é o R-MAC. O protocolo de roteamento *VBF* também adota um algoritmo de auto-adaptação localizado e distribuído que permite aos nós pesar o benefício dos pacotes de encaminhamento e, assim, reduzir o consumo de energia descartando os pacotes de baixo benefício. A tabela 3 mostra algumas aplicações com o protocolo R-MAC.



Figura 7 – Cenário



Fonte: Próprio Autor (2019)

Tabela 3 – Aplicações R-MAC

Aplicação	Características	Topologia	Autonomia de energia	Tolerância ao atraso
Monitoramento Oceanográfico e ambiental	Os parâmetros monitorados não variam com muita rapidez. Os dados precisam ser confiáveis	Quadro ou Linha	Alta	Alta
Exploração sub-marina	Os parâmetros monitorados não variam com muita rapidez. Os dados precisam ser confiáveis	Estrela	Alta	Alta
Monitoramento de reservatórios de água	Os parâmetros monitorados variam após longos intervalos de tempo. Os dados precisam ser confiáveis	Linha	Baixa	Alta

Fonte: Próprio Autor (2019)

## 5 RESULTADOS

Este capítulo é responsável por detalhar como foi realizada as simulações da arquitetura no NS-2, bem como descrever a avaliação de desempenho do protocolo R-MAC em diferentes tipos de topologias no simulador utilizado. A autonomia em termos de vazão, atraso e média de consumo de energia e foram definidos de acordo com as características e os tipos de dados gerados por cada aplicação.

### 5.1 Simulações

Para a avaliação de cada topologia é utilizada uma arquitetura baseada em duas dimensões (2D) com transmissão acústica. Cada topologia é composta por nove nós, sendo oito nós transmissores e um nó sorvedouro (sink), que coleta as informações dos pacotes, a interface e o encaminhamento das informações para uma rede na superfície. Nas simulações consideramos que a conexão do nó sorvedouro com a superfície é realizada através de um cabo. A duração do tempo de simulação total será de 1000 segundos (s).

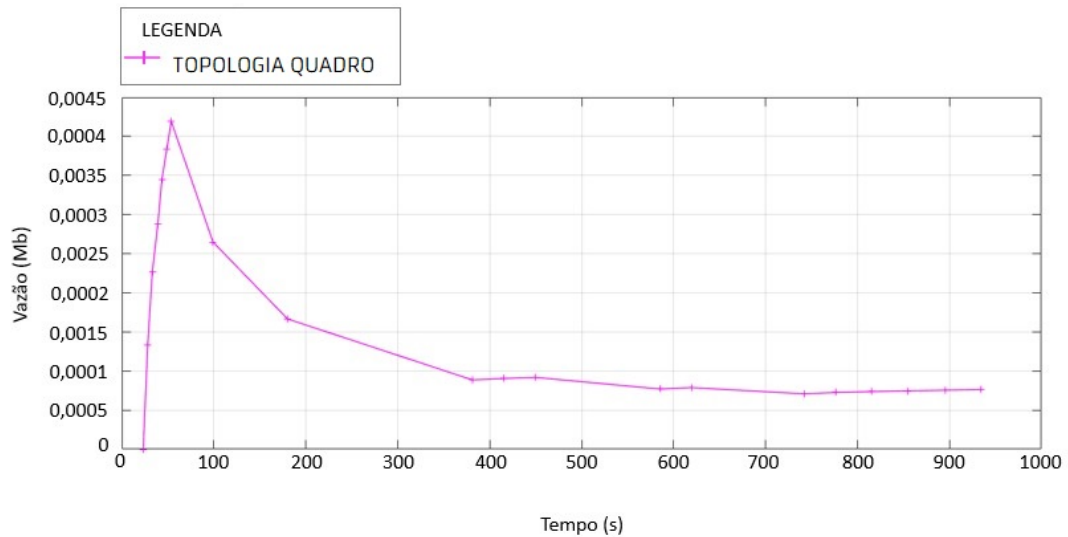
O R-MAC é um protocolo MAC baseado em reserva projetado para redes de sensores subaquáticos de longo atraso, proposto pelo laboratório UWSN na Universidade de Connecticut. No R-MAC, todos os nós são sincronizados e o R-MAC agenda a transmissão de pacotes de controle e pacotes de dados para cancelar completamente a colisão de pacotes de dados. O R-MAC pode alcançar alta eficiência energética e equidade.

O canal de comunicação que usaremos para esta simulação é o canal acústico subaquático *UnderwaterChannel*. O modelo de propagação aqui é *UnderwaterPropagation*, que simula a propagação do sinal acústico subaquático no ambiente subaquático. Tanto o longo atraso de propagação como a alta taxa de atenuação são considerados neste modelo. Você pode alterar os parâmetros neste modelo de propagação. Por exemplo, podemos definir a velocidade de propagação e a taxa de atenuação. O modelo de camada física para este exemplo é *UnderwaterPhy*, que é um tipo de modelo de camada física do *Aqua-Sim* onde é adotada tanto a propriedade half-duplex, quanto o modelo de energia. O poder de transmissão de cada nó subaquático é de 0.6 watt e a potência de recepção é de 0.3 watts. A antena que usamos é a *Antena-Omnidireccional* que pode transmitir sinal em todas as direções igualmente. Após especificar todos os parâmetros do protocolo R-MAC, construímos o nó subaquático, combinando todas as camadas de rede diferentes, todos os nós na rede usarão este modelo de nó.

Os resultados alcançados pela coleta da vazão na primeira simulação. Mostram-nos o valor total da vazão coletada pela topologia quadro, onde a rede atingiu seu pico máximo e recebeu uma taxa por volta de 0,0004 Mbps nos primeiros 50 segundos (s) da simulação, tendo uma queda a partir dos 100 segundos (s), e manteve-se constante dos 400 segundos (s) em diante com uma taxa de transmissão por volta de 0,0001 Mbps, isso é justificado pelo fato de que os nós

transmissores acabam ficando mais distantes do nó coletor principal, apresentando essa resultado. No gráfico 1, nos é mostrado o gráfico da vazão da topologia quadro no cenário proposto.

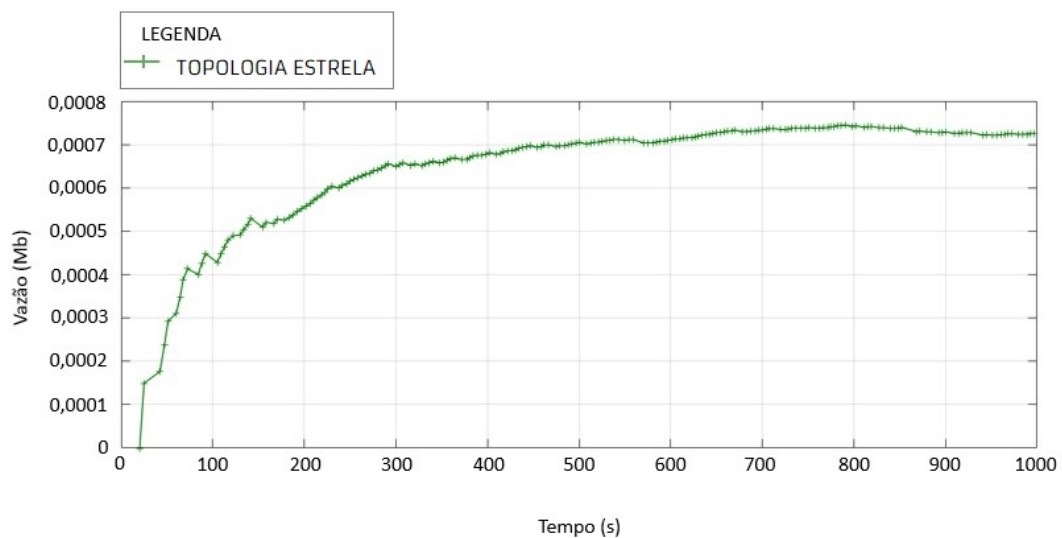
**Gráfico 1 – Gráfico - Topologia quadro**



**Fonte: Próprio Autor (2019)**

A topologia estrela foi a que apresentou um melhor resultado em termos de vazão, podemos observar que ela se manteve com uma taxa de transmissão constante por volta de 0,0006 a 0,0008 Mbps a partir dos 200 segundos (s) se mantendo constante até o fim da simulação. Isso é justificado pelo fato que nessa topologia os nós transmissores estão todos mais próximos ao seu nó coletor, apresentando assim uma melhor taxa de transmissão e consequentemente um melhor resultado.

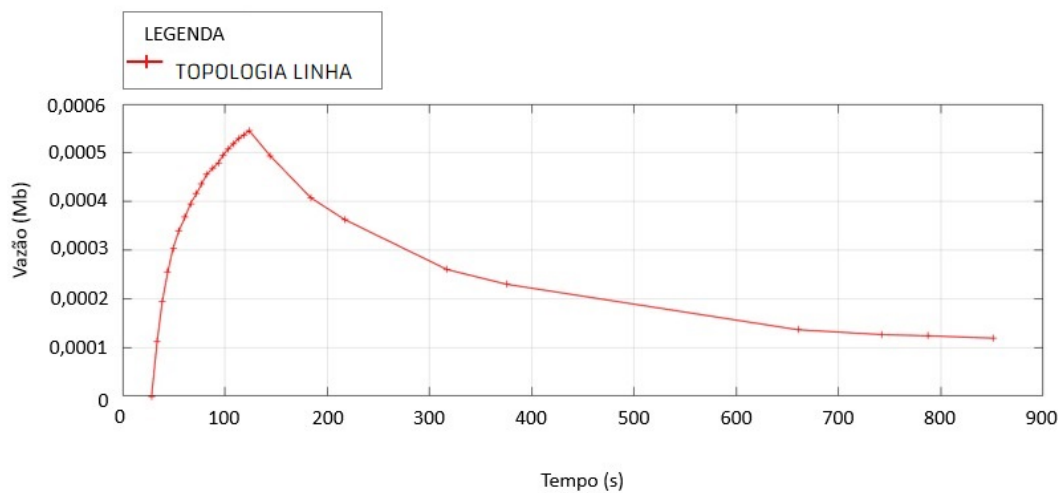
**Gráfico 2 – Gráfico - Topologia estrela**



**Fonte: Próprio Autor (2019)**

Os resultados alcançados pela coleta da vazão na terceira simulação. Mostram-nos o valor total da vazão coletada pela topologia linha, onde a rede teve um pico de 0,0005 Mbps nos primeiros 100 segundos (s) da simulação, tendo uma queda gradativa após os 150 segundos (s) e se manteve constante com uma taxa por volta de 0,0001 a 0,0002 Mbps se mantendo constante até o fim da simulação. No gráfico 3, nos é mostrado o gráfico da vazão da atuação da topologia linha no cenário proposto.

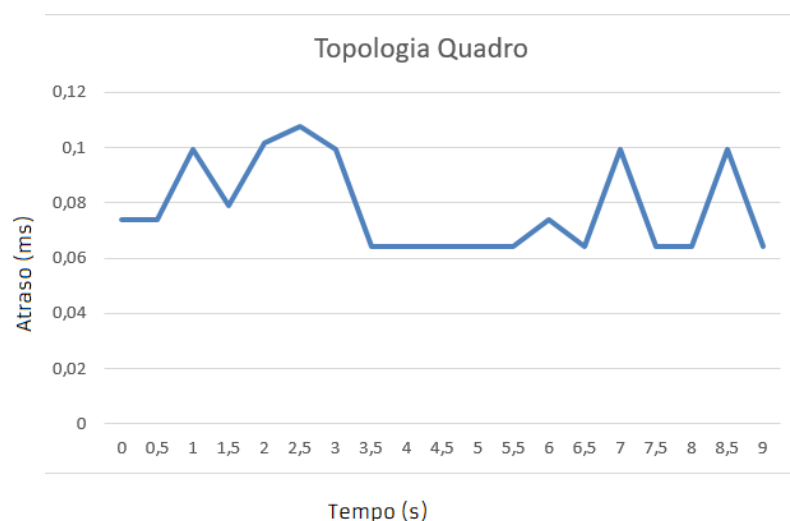
**Gráfico 3 – Gráfico - Topologia linha**



**Fonte: Próprio Autor (2019)**

No gráfico 4, nos é mostrado o resultado do atraso da topologia quadro. Analisando o gráfico podemos observar que essa topologia, obteve um atraso de aproximadamente 0.06 a 0.08 milissegundos. Visto que essa topologia não apresentou uma taxa de de aplicação muito alta, consequentemente não apresentou grandes variações de atraso.

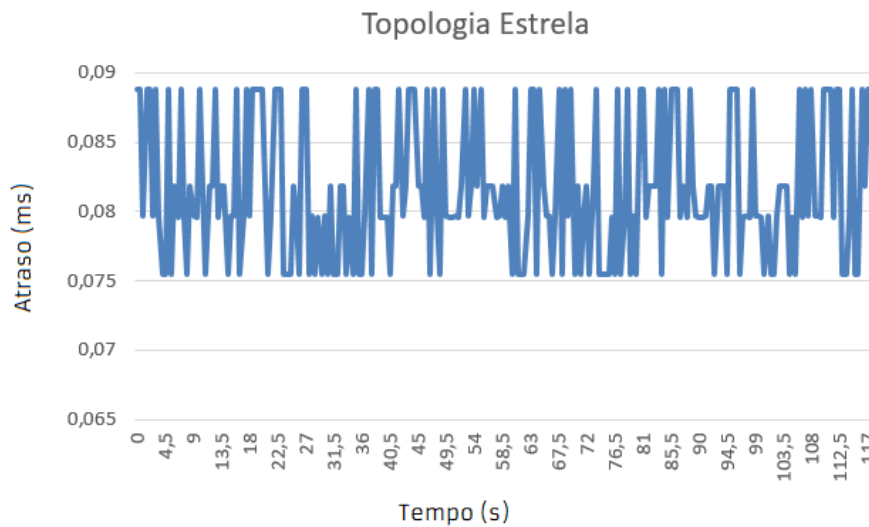
**Gráfico 4 – Atraso - Topologia Quadro**



**Fonte: Próprio Autor (2019)**

No gráfico 5, é mostrado o resultado do atraso da topologia estrela analisando o gráfico podemos observar que essa topologia apesar de ter tido um bom resultado em termos de vazão houve uma grande variação de atraso, isso é justificado pelo fato que na topologia estrela os pacotes necessitam realizar vários saltos para chegar no seu destino final.

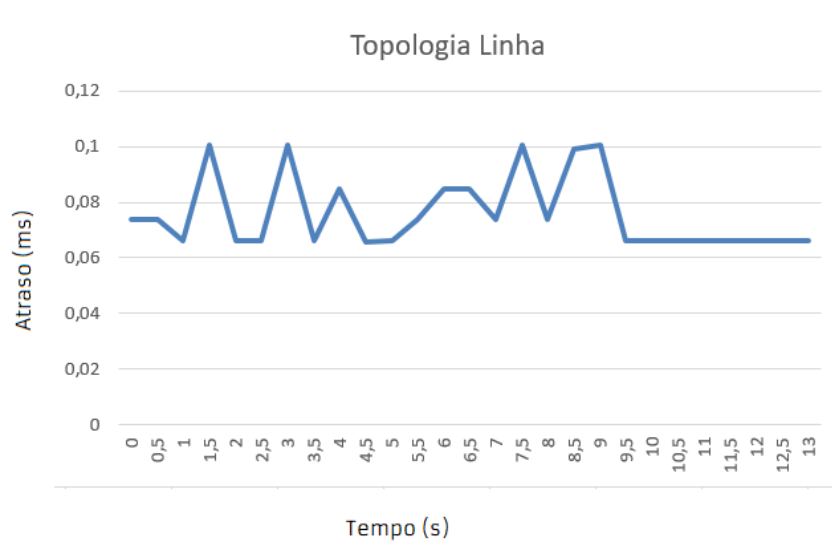
**Gráfico 5 – Atraso - Topologia Estrela**



**Fonte: Próprio Autor (2019)**

No gráfico 6, é mostrado o resultado do atraso da topologia linha. Analisando o gráfico podemos observar que essa topologia, obteve um atraso de aproximadamente 0.06 a 0.08 milissegundos. Seu resultado foi pouco similar a topologia quadro, não apresentando grandes variações de atraso.

**Gráfico 6 – Atraso - Topologia Linha**



**Fonte: Próprio Autor (2019)**

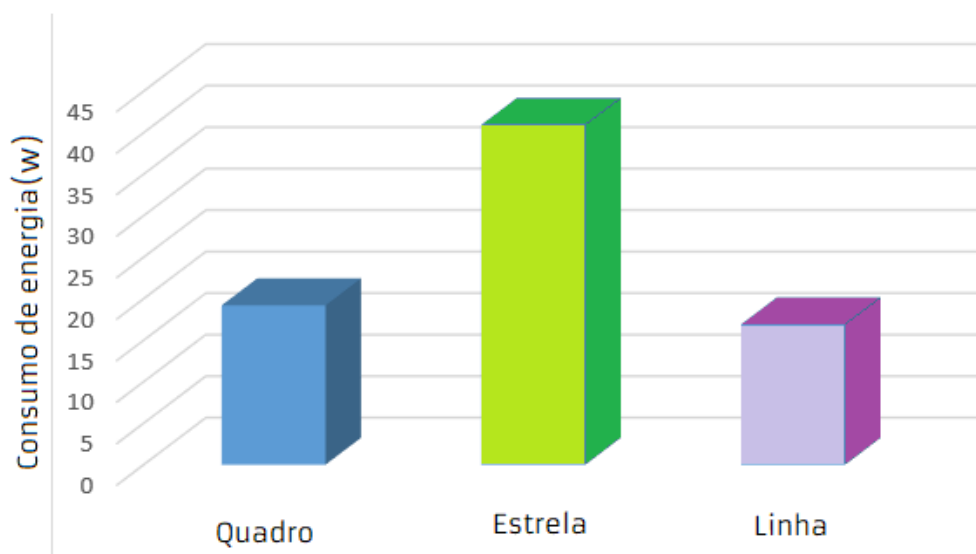
## 5.2 Média do consumo de energia

O R-MAC é um protocolo que alcança eficiência de energia. Esse comportamento é justificado pelo modelo de operação escutar/hibernar do R-MAC, onde os nós escutam periodicamente o meio para verificar se existe tráfego. Se o canal estiver livre e houver pacotes a transmitir, o transmissor enviará um pacote solicitando ao receptor a reserva do canal. Este processo reduz consideravelmente as colisões no meio e o consumo de energia. Usando escuta periódica e modos de dormir para reduzir a energia gasta em estados ociosos. Todos os nós usando este protocolo têm a mesma duração de períodos para escutar e dormir, mas aleatoriamente selecionam seu escalonamento de tal forma a ter eficiência de energia e justiça de acesso ao meio.

O R-MAC compreende três fases: detecção de latência, anúncio do período, e operação periódica. As duas primeiras fases sincronizam cada nó com seus vizinhos e a terceira realiza operações de escutar/dormir e comunicação de dados. Um nó na fase de detecção de latência detecta a latência de propagação para todos os seus nós vizinhos. Na fase de anúncio do período, cada nó aleatoriamente escolhe seu próprio escalonamento para escutar/dormir e propaga essa informação. Os dados são transmitidos na fase periódica de operação.

A autonomia em termos de consumo de energia foi definido de acordo com as características e os tipos de dados gerados por cada aplicação. As aplicações de monitoramento oceanográfico e ambiental e de exploração submarina, por exemplo, não geram informações constantemente já que os parâmetros oceanográficos monitorados não variam com muita rapidez, podendo levar dias ou horas. Apesar disso as informações coletadas devem ser transmitidas de forma confiável e os sensores devem ter uma grande autonomia de energia.

**Gráfico 7 – Média consumo de energia.**



**Fonte: Próprio Autor (2019)**

Apesar dos ganhos em vazão, o protocolo R-MAC exige mais energia para o seu correto funcionamento. Visto que quando se aumenta a taxa de transmissão eles podem desperdiçar energia, retransmitido o mesmo pacote de dados. O gráfico 7 mostra a média do consumo de energia das 3 topologias analisadas.

## 6 CONSIDERAÇÕES FINAIS

As RSAs são uma área de pesquisa importante. Há uma perspectiva de seu crescimento e utilização em diversas áreas nos próximos anos, como: oceanografia, biologia marinha, estudos da interação entre oceanos e atmosfera, estudos do clima, aquecimento global, arqueologia no fundo do mar, predições sísmicas, na área de saúde para detecção de poluentes e substâncias contaminantes, controlando a qualidade da água, e áreas importantes da economia como exploração e monitoração de campos de gás, óleo e petróleo.

A partir da análise das simulações realizadas neste estudo conclui-se que as aplicações de monitoramento e controle, baseadas em redes acústicas submarinas, são consideradas viáveis para aplicações práticas. Os desenvolvedores das aplicações devem atentar principalmente para o tipo de protocolo e a topologia que melhor se adequam a aplicação, ou seja, que irão proporcionar um melhor desempenho da rede e uma maior confiabilidade dos dados.

Este estudo pode ser considerado um guia inicial para auxiliar os desenvolvedores na definição destes parâmetros básicos, mas fundamentais ao sucesso de uma aplicação em redes acústicas submarinas. As RSAs deveriam se tornar um dos principais focos de pesquisadores. Se essas aplicações são devidamente exploradas e atendidas, muitas vidas, tempo e dinheiro podem ser salvos. Outra forma de se reduzir a energia consumida pelas redes de sensores sem fio é através de melhorias nos projetos de hardware, diminuindo a potência consumida por esses dispositivos ou melhorando o armazenamento de energia. Felizmente, avanços significativos estão ocorrendo nessa área.

Como trabalho futuro pretende-se avaliar outros protocolos MAC e de roteamento associados às topologias quadro, estrela e linha. Com isso será possível a elaboração de um guia de referência mais completo para os desenvolvedores.



## REFERÊNCIAS

- VIEIRA L., L. A. F. A. e. C. M. . Redes de sensores aquáticas. *XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Gramado, RS, Brasil, 2010.
- ACAR, G.; ADAMS, A. E. **ACMENet: an underwater acoustic sensor network protocol for real-time environmental monitoring in coastal areas**. [S.l.]: IEE Proceedings: Radar, Sonar and Navigation, vol. 153, no. 4, pp. 365–380, 2006.
- AKYILDIZ I. F., P. D.; MELODIA, T. **Underwater acoustic sensor networks: Research challenges**. [S.l.]: Ad Hoc Networks (Elsevier), 3:257–279., 2005.
- ALFOUZAN ALIREZA SHAHRABI, S. M. G. F.; BOUTALEB, T. **Efcient Depth-Based Scheduling MAC Protocolfor Underwater Sensor**. [S.l.]: Ninth International Conference on Ubiquitous and Future Networks (ICUFN), 2017.
- ALIPPI R. CAMPLANI, C. G. C.; ROVERI, M. **A robust, adaptive, solar-powered WSN framework for aquatic environmental monitoring**. [S.l.]: IEEE Sensors Journal, vol. 11, no. 1, pp. 45–55, 2011.
- BHARGHAVAN V., D. A. S. S.; ZHANG, L. **Macaw: a media access protocol for wireless lan's**. [S.l.]: In SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications, pages 212–225, New York, NY, USA. ACM., 1994.
- BREKHOVSKIKH L. M. E LYSANOV, Y. P. **Fundamentals of Ocean Acoustics**. [S.l.]: Springer-Verlag., 1991.
- CASEY, A. L. K.; DOZIER, G. **A sensor network architecture for tsunami detection and response**. [S.l.]: International Journal of Distributed Sensor Networks, vol. 4, no. 1, pp. 28–43, 2008.
- CAYIRCI E., T. H. D. Y. e. C. V. **Wireless sensor networks for underwater surveillance systems**. [S.l.]: Cd Hoc Networks vol. 4, no. 4, pp. 409-23, 2004.
- COUTINHO AZZEDINE BOUKERCHE, L. F. M. V. e. A. A. F. L. R. W. L. **Um Protocolo de Roteamento para o Consumo Balanceado de Energia em Redes de Sensores Aquáticas**. [S.l.]: XXXV, Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SRBC), 2017.
- FELEMBAN FAISAL KARIM SHAIKH, U. M. Q. A. A. S. S. B. Q. E. **Underwater Sensor Network Applications: A Comprehensive Survey**. [S.l.]: This article is part of the following special collection(s): Highly Cited Articles, Underwater Wireless Sensor Networks, 2015.
- FLIKKEMA, P. G. **Spread-spectrum techniques for wireless communication**. [S.l.]: IEEE Signal Processing Magazine, 14(3), 26-36., 1997.
- FULLMER, C. L.; GARCIA-LUNA-ACEVES, J. J. **Floor acquisition multiple access (fama) for packet-radio networks**. [S.l.]: In ACM SIGCOMM 95., 1995.
- HARRIS A., F. e. Z. M. **Modeling the Underwater Acoustic Channel in ns2**. [S.l.]: ValueTools '07: Proceedings of the 2nd International Conference on Performance Evaluation Methodologies and Tools, pp. 1–8, ICST, Bélgica., 2007.

- IWENDI, C. O.; ALLEN, A. R. **Wireless sensor network nodes: security and deployment in the niger-delta oil and gas sector.** [S.l.]: International Journal of Network Security Its Applications, vol. 3, no. 1, pp. 68–79, 2011.
- KARN, P. **Maca : A new channel access protocol for packet radio.** [S.l.]: In ARRL/CRRL Amateur Radio 9th Computer Networking Conference, pages 134–140., 1990.
- KHAN, A.; JENKINS, L. **Undersea wireless sensor network for ocean pollution prevention.** [S.l.]: in Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE '08), pp. 2–8, 2008.
- KONG J., C. J. W. D.; GERLA, M. **Building underwater ad-hoc networks and sensor networks for large scale real-time aquatic applications.** In IEEE MILCOM. [S.l.]: Atlantic City, NJ, USA., 2005.
- KUMAR, P. P. P.; SRIJA. **Underwater acoustic sensor network for early warning generation.** [S.l.]: Proceedings of the Oceans, pp. 1–6, IEEE, Hampton Roads, Va, USA, 2012.
- MOLINS, M. **Slotted fama: a mac protocol for underwater acoustic networks.** [S.l.]: In IEEE OCEANS 06, Singapore, pages 16–19., 2006.
- PAHLAVAN, K.; LEVESQUE, A. H. **Wireless Information Networks.** [S.l.]: Second Edition, 2005.
- PARTAN J., K. J.; LEVINE, B. N. **A survey of practical issues in underwater networks.** [S.l.]: In WUWNet'06, pages 17–24, Los Angeles, CA, USA., 2006.
- PASI, A. A.; BHAVE, U. **Flood detection system using wireless sensor network.** [S.l.]: International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5, no. 2, pp. 386–389, 2015.
- POMPILI D., M. T.; AKYILDIZ, I. F. **A cdma-based medium access control for underwater acoustic sensor networks.** [S.l.]: Trans. Wireless. Comm., 8(4):1899–1909., 2009.
- ROBERTS, L. G. **Aloha packet system with and without slots and capture.** [S.l.]: SIGCOMM Comput. Commun. Rev., 5(2):28–42., 1975.
- SMITH S., P. J.; NEEL, A. **A peer-to-peer communication protocol for underwater acoustic communication.** [S.l.]: In Oceans, pages 268–272., 1997.
- SOREIDE N.N., W. C. e. H. S. **Overview of ocean based buoys and drifters: Present applications and future needs.** [S.l.]: Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology., 2004.
- TUNA, O. A. G.; GULEZ, K. **Continuous monitoring of water quality using portable and low-cost approaches.** [S.l.]: International Journal of Distributed Sensor Networks, vol. 2013, Article ID 249598, 11 pages, 2013.
- TYAN, S.; OH, S.-H. **AUV-RM: underwater sensor network scheme for AUV based river monitoring.** [S.l.]: Research Trend in Computer and Applications, SERCE, vol. 24, pp. 53–55, 2013.
- URICK, R. J. **Principles of Underwater Sound.** [S.l.]: McGraw-Hill, 1983.

- VIEIRA L. F. M., K. J. L. U.; GERLA, M. **Analysis of Aloha Protocols for Underwater**. [S.l.]: In ACM International Workshop on Underwater Networks, WUWNet'06, Los Angeles, CA, 2006.
- VIEIRA L. F. M., V. M. A. M. N. J. A.; VIEIRA, A. B. **Autonomous wireless lake monitoring**. [S.l.]: n Computing in Science Engineering, 2018.
- XIE, J.-H. C. P.; LAO, L. **VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks**. [S.l.]: Part of the Lecture Notes in Computer Science book series (LNCS, volume 3976), 2006.
- XIE, P.; CUI, J.-H. **R-MAC: An Energy-Efficient MAC Protocol for Underwater Sensor Networks**. [S.l.]: International Conference on Wireless Algorithms, Systems and Applications, 2007.
- XIE, P.; CUI, J. H. **R-MAC: An Energy-Efficient MAC Protocol for Underwater Sensor Networks**. [S.l.]: In International Conference on Wireless Algorithms, Systems and Applications., 2007.
- XIE P., C.-J. H. e. L. L. **VBF: Vector-Based Forwarding Protocol for Underwater Sensor Networks**. [S.l.]: In IFIP Networking, Coimbra. Portugal., 2006.
- XIE P., Z.-Z. Z. P. H. Y. H. T. C. J.-H. S. Z. F. Y. e. Z. S. **Aqua-Sim: A NS-2 based simulator for underwater sensor networks**. [S.l.]: in OCEANS MTS/IEEE., 2009.
- XIE ZHONG ZHOU, Z. P. H. Y. T. H. J.-H. C. Z. S. Y. F. S. Z. P. **Aqua-Sim: An NS-2 Based Simulator for Underwater Sensor Networks**. [S.l.]: IEEE, 2009.
- YACKOSKI, J.; SHEN, C.-C. **Uw-flashr: achieving high channel utilization in a time-based acoustic mac protocol**. [S.l.]: In WuWNeT '08: Proceedings of the third ACM international workshop on Underwater Networks, pages 59–66, New York, NY, USA. ACM., 2008.
- YALCUK, A.; POSTALCIOGLU, S. **Evaluation of pool water quality of trout farms by fuzzy logic: monitoring of pool water quality for trout farms**. [S.l.]: International Journal of Environmental Science and Technology, vol. 12, no. 5, pp. 1503–1514, 2015., 2015.
- YAN H., S. Z. J. e. C. J. H. **DBR: Depth-Based Routing for Underwater Sensor Networks**. [S.l.]: Industrial Mechatronics and Automation (ICIMA), 2010 2nd International Conference., 2010.
- YANG ONG, D. Z. M. e. G. **Design of a wireless sensor network for long-term, in-situ monitoring of an aqueous environment**. [S.l.]: Sensors, vol. 2, pp. 455–472., 2002.
- ZHANG B, S. G. e. R. A. **Adaptive sampling for marine microorganism monitoring**. [S.l.]: IEEE/RSJ International Conference on Intelligent Robots and Systems., 2004.

## **Apêndices**

## APÊNDICE A – TOPOLOGIA QUADRO

```

set opt(prop)           Propagation / UnderwaterPropagation
set opt(netif)          Phy / UnderwaterPhy
set opt(mac)            Mac / UnderwaterMac / RMac
set opt(ifq)            Queue / DropTail / PriQueue
set opt(ll)             LL
set opt(energy)         EnergyModel
set opt(txpower)        0.6
set opt(rxpower)        0.3
set opt(initialenergy) 10000
set opt(idlepower)      0.01
set opt(ant)            Antenna / OmniAntenna
set opt(filters)        GradientFilter

set opt(bit_rate)       1.0e4
set opt(encoding_efficiency) 1
set opt(ND_window)     1
set opt(ACKND_window)  1
set opt(PhaseOne_window) 3
set opt(PhaseTwo_window) 1
set opt(PhaseTwo_interval) 0.5
set opt(IntervalPhase2Phase3) 1
set opt(duration)      0.1
set opt(PhyOverhead)   8
set opt(large_packet_size) 480
set opt(short_packet_size) 40
set opt(PhaseOne_cycle) 4
set opt(PhaseTwo_cycle) 2
set opt(PeriodInterval) 1
set opt(transmission_time_error) 0.0001

set opt(dz)             10
set opt(ifqlen)         50
set opt(nn)             9
set opt(layers)         1
set opt(x)              100
set opt(y)              100

```

```

set opt(z) [expr ($opt(layers)-1)*$opt(dz)
set opt(seed) 348.88
set opt(stop) 1000
set opt(prestop) 20
set opt(tr) "rmac1.tr"
set opt(nam) "rmac1.nam"
set opt(adhocRouting) Vectorbasedforward
set opt(width) 20
set opt(adj) 10
set opt(interval) 0.001

```

```

LL set mindelay_ 50us
LL set delay_ 25us
LL set bandwidth_ 0

```

```

Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Z_ 0.05
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

```

```

Mac/UnderwaterMac set bit_rate_ $opt(bit_rate)
Mac/UnderwaterMac set encoding_efficiency_ $opt(encoding_efficiency)
Mac/UnderwaterMac/RMac set ND_window_ $opt(ND_window)
Mac/UnderwaterMac/RMac set ACKND_window_ $opt(ACKND_window)
Mac/UnderwaterMac/RMac set PhaseOne_window_ $opt(PhaseOne_window)
Mac/UnderwaterMac/RMac set PhaseTwo_window_ $opt(PhaseTwo_window)
Mac/UnderwaterMac/RMac set PhaseTwo_interval_ $opt(PhaseTwo_interval)
Mac/UnderwaterMac/RMac set
IntervalPhase2Phase3_ $opt(IntervalPhase2Phase3)
Mac/UnderwaterMac/RMac set duration_ $opt(duration)
Mac/UnderwaterMac/RMac set PhyOverhead_ $opt(PhyOverhead)
Mac/UnderwaterMac/RMac set large_packet_size_ $opt(large_packet_size)
Mac/UnderwaterMac/RMac set short_packet_size_ $opt(short_packet_size)
Mac/UnderwaterMac/RMac set PhaseOne_cycle_ $opt(PhaseOne_cycle)
Mac/UnderwaterMac/RMac set PhaseTwo_cycle_ $opt(PhaseTwo_cycle)

```

```
Mac/UnderwaterMac/RMac set PeriodInterval_ $opt(PeriodInterval)
Mac/UnderwaterMac/RMac set
transmission_time_error_ $opt(transmission_time_error)
```

```
Phy/UnderwaterPhy set CPThresh_ 100
Phy/UnderwaterPhy set CStresh_ 0
Phy/UnderwaterPhy set RXThresh_ 0
```

```
Phy/UnderwaterPhy set Pt_ 0.2818
Phy/UnderwaterPhy set freq_ 25
Phy/UnderwaterPhy set K_ 2.0
```

```
remove-all-packet-headers
```

```
add-packet-header IP Mac LL ARP UWVB RMAC
```

```
set ns_ [new Simulator]
set topo [new Topography]
```

```
$topo load_cubicgrid $opt(x) $opt(y) $opt(z)
```

```
set tracefd [open $opt(tr) w]
$ns_ trace-all $tracefd
```

```
set nf [open $opt(nam) w]
$ns_ namtrace-all-wireless $nf $opt(x) $opt(y)
```

```
set phase1_time [expr $opt(PhaseOne_cycle)*$opt(PhaseOne_window)]
set phase2_time
[expr $opt(PhaseTwo_cycle)*($opt(PhaseTwo_window)
+$opt(PhaseTwo_interval))]
set start_time [expr $phase1_time+$phase2_time+
$opt(IntervalPhase2Phase3)]
```

```
puts "the start time is $start_time"
```

```
set total_number [expr $opt(nn)-1]
```

```
set god_ [create-god $opt(nn)]

set chan_1_ [new $opt(chan)]

global defaultRNG
$defaultRNG seed $opt(seed)

$ns_ node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -topoInstance $topo \
    -energyModel $opt(energy) \
    -txpower $opt(txpower) \
    -rxpower $opt(rxpower) \
    -initialEnergy $opt(initialenergy) \
    -idlePower $opt(idlepower) \
    -channel $chan_1_

set node_(0) [$ns_ node 0]
$node_(0) set sinkStatus_ 1
$node_(0) set passive 1

$ns_ at 0.0 "$node_(0) label COLECTOR"
$ns_ at 0.0 "$node_(0) add-mark n1 blue hexagon"

$god_ new_node $node_(0)
$node_(0) set X_ -38
$node_(0) set Y_ 45
$node_(0) set Z_ 0.0
```



```
$node_(0) set passive 1

set a_(0) [new Agent/UWSink] ;
$ns_ attach-agent $node_(0) $a_(0)
$a_(0) attach-vectorbasedforward $opt(width)
$a_(0) cmd set-range 20
$a_(0) cmd set-target-x -20
$a_(0) cmd set-target-y -10
$a_(0) cmd set-target-z -20

set node_(1) [$ns_ node 1]
$node_(1) set sinkStatus_ 1

$god_ new_node $node_(1)
$node_(1) set X_ 0
$node_(1) set Y_ 45
$node_(1) set Z_ 0
$node_(1) set-cx 0
$node_(1) set-cy -15
$node_(1) set-cz 0
$node_(1) set_next_hop 0

set a_(1) [new Agent/UWSink]
$ns_ attach-agent $node_(1) $a_(1)
$a_(1) attach-vectorbasedforward $opt(width)
$a_(1) cmd set-range 20
$a_(1) cmd set-target-x 0
$a_(1) cmd set-target-y 0
$a_(1) cmd set-target-z 0
$a_(1) set data_rate_ 0.05

set node_(2) [$ns_ node 2]
$node_(2) set sinkStatus_ 1

$god_ new_node $node_(2)
$node_(2) set X_ 38
$node_(2) set Y_ 45
```

```
$node_(2) set Z_ 0
$node_(2) set-cx 0
$node_(2) set-cy 40
$node_(2) set-cz 0
$node_(2) set_next_hop 0

set a_(2) [new Agent/UWSink]
$ns_ attach-agent $node_(2) $a_(2)
$a_(2) attach-vectorbasedforward $opt(width)
$a_(2) cmd set-range 20
$a_(2) cmd set-target-x 0
$a_(2) cmd set-target-y 0
$a_(2) cmd set-target-z 0
$a_(2) set data_rate_ 0.05

set node_(3) [$ns_ node 3]
$node_(3) set sinkStatus_ 1

$god_ new_node $node_(3)
$node_(3) set X_ -38
$node_(3) set Y_ 22
$node_(3) set Z_ 0
$node_(3) set-cx 0
$node_(3) set-cy -30
$node_(3) set-cz 0
$node_(3) set_next_hop 0

set a_(3) [new Agent/UWSink]
$ns_ attach-agent $node_(3) $a_(3)
$a_(3) attach-vectorbasedforward $opt(width)
$a_(3) cmd set-range 20
$a_(3) cmd set-target-x 0
$a_(3) cmd set-target-y 0
$a_(3) cmd set-target-z 0
$a_(3) set data_rate_ 0.05

set node_(4) [$ns_ node 4]
$node_(4) set sinkStatus_ 1
```

```
$god_ new_node $node_(4)
$node_(4) set X_ 0
$node_(4) set Y_ 22
$node_(4) set Z_ 0
$node_(4) set-cx 0
$node_(4) set-cy -50
$node_(4) set-cz 0
$node_(4) set_next_hop 0

set a_(4) [new Agent/UWSink]
$ns_ attach-agent $node_(4) $a_(4)
$a_(4) attach-vectorbasedforward $opt(width)
$a_(4) cmd set-range 40
$a_(4) cmd set-target-x 0
$a_(4) cmd set-target-y 0
$a_(4) cmd set-target-z 0
$a_(4) set data_rate_ 0.05

set node_(5) [$ns_ node 5]
$node_(5) set sinkStatus_ 1

$god_ new_node $node_(5)
$node_(5) set X_ 38
$node_(5) set Y_ 22
$node_(5) set Z_ 0
$node_(5) set-cx 0
$node_(5) set-cy -50
$node_(5) set-cz 0
$node_(5) set_next_hop 0

set a_(5) [new Agent/UWSink]
$ns_ attach-agent $node_(5) $a_(5)
$a_(5) attach-vectorbasedforward $opt(width)
$a_(5) cmd set-range 40
$a_(5) cmd set-target-x 0
$a_(5) cmd set-target-y 0
$a_(5) cmd set-target-z 0
$a_(5) set data_rate_ 0.05
```

```
set node_(6) [$ns_ node 6]
$node_(6) set sinkStatus_ 1

$god_ new_node $node_(6)
$node_(6) set X_ -38
$node_(6) set Y_ 0
$node_(6) set Z_ 0
$node_(6) set-cx 0
$node_(6) set-cy -50
$node_(6) set-cz 0
$node_(6) set_next_hop 0

set a_(6) [new Agent/UWSink]
$ns_ attach-agent $node_(6) $a_(6)
$a_(6) attach-vectorbasedforward $opt(width)
$a_(6) cmd set-range 40
$a_(6) cmd set-target-x 0
$a_(6) cmd set-target-y 0
$a_(6) cmd set-target-z 0
$a_(6) set data_rate_ 0.05

set node_(7) [$ns_ node 7]
$node_(7) set sinkStatus_ 1

$god_ new_node $node_(7)
$node_(7) set X_ 0
$node_(7) set Y_ 0
$node_(7) set Z_ 0
$node_(7) set-cx 0
$node_(7) set-cy -50
$node_(7) set-cz 0
$node_(7) set_next_hop 0

set a_(7) [new Agent/UWSink]
$ns_ attach-agent $node_(7) $a_(7)
$a_(7) attach-vectorbasedforward $opt(width)
```

```
$a_(7) cmd set-range 40
$a_(7) cmd set-target-x 0
$a_(7) cmd set-target-y 0
$a_(7) cmd set-target-z 0
$a_(7) set data_rate_ 0.05
```

```
set node_($total_number) [$ns_ node $total_number]
```

```
$god_ new_node $node_($total_number)
$node_($total_number) set X_ 38
$node_($total_number) set Y_ 0
$node_($total_number) set Z_ 0
$node_($total_number) set-cx 60
$node_($total_number) set-cy 0
$node_($total_number) set-cz 0
$node_($total_number) set_next_hop 0
```

```
set a_($total_number) [new Agent/UWSink]
$ns_ attach-agent $node_($total_number) $a_($total_number)
$a_($total_number) attach-vectorbasedforward $opt(width)
$a_($total_number) cmd set-range 20
$a_($total_number) cmd set-target-x 0
$a_($total_number) cmd set-target-y 0
$a_($total_number) cmd set-target-z 0
$a_($total_number) set data_rate_ 0.05
```

```
$ns_ at $start_time "$a_($total_number) exp-start"
$ns_ at $start_time "$a_(1) exp-start"
$ns_ at $start_time "$a_(2) exp-start"
$ns_ at $start_time "$a_(3) exp-start"
$ns_ at $start_time "$a_(4) exp-start"
$ns_ at $start_time "$a_(5) exp-start"
$ns_ at $start_time "$a_(6) exp-start"
$ns_ at $start_time "$a_(7) exp-start"
```

```

set node_size 11
for {set k 0} { $k<$opt(nn) } { incr k } {
    $ns_ initial_node_pos $node_($k) $node_size
}

puts "++++++AFTER ANNOUNCE++++++"

$ns_ at $opt(stop).001 "$a_(0) terminate"
$ns_ at $opt(stop).002 "$a_($total_number) terminate"
$ns_ at $opt(stop).002 "$a_(1) terminate"
$ns_ at $opt(stop).002 "$a_(2) terminate"
$ns_ at $opt(stop).002 "$a_(3) terminate"
$ns_ at $opt(stop).002 "$a_(4) terminate"
$ns_ at $opt(stop).002 "$a_(5) terminate"
$ns_ at $opt(stop).002 "$a_(6) terminate"
$ns_ at $opt(stop).002 "$a_(7) terminate"

$ns_ at $opt(stop).003 "$god_ compute_energy"
$ns_ at $opt(stop).004 "$ns_ nam-end-wireless $opt(stop)"
$ns_ at $opt(stop).005 "puts \"NS EXISTING...\"; $ns_ halt"

puts $tracefd "vectorbased"
puts $tracefd "M 0.0 nn $opt(nn) x $opt(x) y $opt(y) z $opt(z)"
puts $tracefd "M 0.0 prop $opt(prop) ant $opt(ant)"
puts "starting Simulation..."
proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam rmac2.nam &
    exit 0
}

$ns_ run

```

## APÊNDICE B – TOPOLOGIA ESTRELA

```

set opt(chan)           Channel/UnderwaterChannel
set opt(prop)          Propagation/UnderwaterPropagation

set opt(netif)         Phy/UnderwaterPhy
set opt(mac)           Mac/UnderwaterMac/RMac
set opt(ifq)           Queue/DropTail/PriQueue
set opt(ll)            LL
set opt(energy)        EnergyModel
set opt(txpower)       0.6
set opt(rxpower)       0.3
set opt(initialenergy) 10000
set opt(idlepower)     0.01
set opt(ant)           Antenna/OmniAntenna
set opt(filters)       GradientFilter

set opt(bit_rate)      1.0e4
set opt(encoding_efficiency) 1
set opt(ND_window)     1
set opt(ACKND_window) 1
set opt(PhaseOne_window) 3
set opt(PhaseTwo_window) 1
set opt(PhaseTwo_interval) 0.5
set opt(IntervalPhase2Phase3) 1
set opt(duration)      0.1
set opt(PhyOverhead)   8
set opt(large_packet_size) 480
set opt(short_packet_size) 40
set opt(PhaseOne_cycle) 4
set opt(PhaseTwo_cycle) 2
set opt(PeriodInterval) 1
set opt(transmission_time_error) 0.0001

set opt(dz)            10
set opt(ifqlen)        50
set opt(nn)            9
set opt(layers)        1

```

```

set opt(x) 100
set opt(y) 100
set opt(z) [expr ($opt(layers)-1)*$opt(dz)]
set opt(seed) 348.88
set opt(stop) 1000
set opt(prestop) 20
set opt(tr) "rmac2.tr"
set opt(nam) "rmac2.nam"
set opt(adhocRouting) Vectorbasedforward
set opt(width) 20
set opt(adj) 10
set opt(interval) 0.001

```

```

LL set mindelay_ 50us
LL set delay_ 25us
LL set bandwidth_ 0

```

```

Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Z_ 0.05
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

```

```

Mac/UnderwaterMac set bit_rate_ $opt(bit_rate)
Mac/UnderwaterMac set encoding_efficiency_ $opt(encoding_efficiency)
Mac/UnderwaterMac/RMac set ND_window_ $opt(ND_window)
Mac/UnderwaterMac/RMac set ACKND_window_ $opt(ACKND_window)
Mac/UnderwaterMac/RMac set PhaseOne_window_ $opt(PhaseOne_window)
Mac/UnderwaterMac/RMac set PhaseTwo_window_ $opt(PhaseTwo_window)
Mac/UnderwaterMac/RMac set PhaseTwo_interval_ $opt(PhaseTwo_interval)
Mac/UnderwaterMac/RMac set
IntervalPhase2Phase3_ $opt(IntervalPhase2Phase3)
Mac/UnderwaterMac/RMac set duration_ $opt(duration)
Mac/UnderwaterMac/RMac set PhyOverhead_ $opt(PhyOverhead)
Mac/UnderwaterMac/RMac set large_packet_size_ $opt(large_packet_size)
Mac/UnderwaterMac/RMac set short_packet_size_ $opt(short_packet_size)

```



```

Mac/UnderwaterMac/RMac set PhaseOne_cycle_ $opt(PhaseOne_cycle)
Mac/UnderwaterMac/RMac set PhaseTwo_cycle_ $opt(PhaseTwo_cycle)
Mac/UnderwaterMac/RMac set PeriodInterval_ $opt(PeriodInterval)
Mac/UnderwaterMac/RMac set
transmission_time_error_ $opt(transmission_time_error)

```

```

Phy/UnderwaterPhy set CPTHresh_ 100
Phy/UnderwaterPhy set CSTresh_ 0
Phy/UnderwaterPhy set RXThresh_ 0
Phy/UnderwaterPhy set Pt_ 0.2818
Phy/UnderwaterPhy set freq_ 25
Phy/UnderwaterPhy set K_ 2.0

```

```

remove-all-packet-headers
add-packet-header IP Mac LL ARP UWVB RMAC

```

```

set ns_ [new Simulator]
set topo [new Topography]

```

```

$topo load_cubicgrid $opt(x) $opt(y) $opt(z)
set tracefd [open $opt(tr) w]
$ns_ trace-all $tracefd

```

```

set nf [open $opt(nam) w]
$ns_ namtrace-all-wireless $nf $opt(x) $opt(y)

```

```

set phase1_time [expr $opt(PhaseOne_cycle)*$opt(PhaseOne_window)]
set phase2_time [expr $opt(PhaseTwo_cycle)*($opt(PhaseTwo_window)
+$opt(PhaseTwo_interval))]
set start_time [expr $phase1_time+$phase2_time+
$opt(IntervalPhase2Phase3)]

```

```

puts "the start time is $start_time"

```

```

set total_number [expr $opt(nn)-1]
set god_ [create-god $opt(nn)]

```

```

set chan_1_ [new $opt(chan)]

```

```
global defaultRNG
$defaultRNG seed $opt(seed)

$ns_ node-config -adhocRouting $opt(adhocRouting) \
    -llType $opt(ll) \
    -macType $opt(mac) \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -agentTrace OFF \
    -routerTrace OFF \
    -macTrace ON \
    -topoInstance $topo \
    -energyModel $opt(energy) \
    -txpower $opt(txpower) \
    -rxpower $opt(rxpower) \
    -initialEnergy $opt(initialenergy) \
    -idlePower $opt(idlepower) \
    -channel $chan_1_

set node_(0) [$ns_ node 0]
$node_(0) set sinkStatus_ 1
$node_(0) set passive 1

$ns_ at 0.0 "$node_(0) label COLECTOR"
$ns_ at 0.0 "$node_(0) add-mark n1 blue hexagon"

$god_ new_node $node_(0)
$node_(0) set X_ 0
$node_(0) set Y_ 0
$node_(0) set Z_ 0.0
$node_(0) set passive 1

set a_(0) [new Agent/UWSink] ;
$ns_ attach-agent $node_(0) $a_(0)
```

```
$a_(0) attach-vectorbasedforward $opt(width)
$a_(0) cmd set-range 20
$a_(0) cmd set-target-x -20
$a_(0) cmd set-target-y -10
$a_(0) cmd set-target-z -20

set node_(1) [$ns_ node 1]
$node_(1) set sinkStatus_ 1

$god_ new_node $node_(1)
$node_(1) set X_ 0
$node_(1) set Y_ -40
$node_(1) set Z_ 0
$node_(1) set-cx 0
$node_(1) set-cy -15
$node_(1) set-cz 0
$node_(1) set_next_hop 0

set a_(1) [new Agent/UWSink]
$ns_ attach-agent $node_(1) $a_(1)
$a_(1) attach-vectorbasedforward $opt(width)
$a_(1) cmd set-range 20
$a_(1) cmd set-target-x 0
$a_(1) cmd set-target-y 0
$a_(1) cmd set-target-z 0
$a_(1) set data_rate_ 0.05

set node_(2) [$ns_ node 2]
$node_(2) set sinkStatus_ 1

$god_ new_node $node_(2)
$node_(2) set X_ 0
$node_(2) set Y_ 40
$node_(2) set Z_ 0
$node_(2) set-cx 0
$node_(2) set-cy 40
$node_(2) set-cz 0
$node_(2) set_next_hop 0
```

```
set a_(2) [new Agent/UWSink]
$ns_ attach-agent $node_(2) $a_(2)
$a_(2) attach-vectorbasedforward $opt(width)
$a_(2) cmd set-range 20
$a_(2) cmd set-target-x 0
$a_(2) cmd set-target-y 0
$a_(2) cmd set-target-z 0
$a_(2) set data_rate_ 0.05
```

```
set node_(3) [$ns_ node 3]
$node_(3) set sinkStatus_ 1
```

```
$god_ new_node $node_(3)
$node_(3) set X_ -60
$node_(3) set Y_ 0
$node_(3) set Z_ 0
$node_(3) set-cx 0
$node_(3) set-cy -30
$node_(3) set-cz 0
$node_(3) set_next_hop 0
```

```
set a_(3) [new Agent/UWSink]
$ns_ attach-agent $node_(3) $a_(3)
$a_(3) attach-vectorbasedforward $opt(width)
$a_(3) cmd set-range 20
$a_(3) cmd set-target-x 0
$a_(3) cmd set-target-y 0
$a_(3) cmd set-target-z 0
$a_(3) set data_rate_ 0.05
```

```
set node_(4) [$ns_ node 4]
$node_(4) set sinkStatus_ 1
```

```
$god_ new_node $node_(4)
$node_(4) set X_ -39.1428
$node_(4) set Y_ 30.3023
$node_(4) set Z_ 0
$node_(4) set-cx 0
```

```
$node_(4) set-cy -50
$node_(4) set-cz 0
$node_(4) set_next_hop 0

set a_(4) [new Agent/UWSink]
$ns_ attach-agent $node_(4) $a_(4)
$a_(4) attach-vectorbasedforward $opt(width)
$a_(4) cmd set-range 40
$a_(4) cmd set-target-x 0
$a_(4) cmd set-target-y 0
$a_(4) cmd set-target-z 0
$a_(4) set data_rate_ 0.05

set node_(5) [$ns_ node 5]
$node_(5) set sinkStatus_ 1

$god_ new_node $node_(5)
$node_(5) set X_ 39.1428
$node_(5) set Y_ 30.3023
$node_(5) set Z_ 0
$node_(5) set-cx 0
$node_(5) set-cy -50
$node_(5) set-cz 0
$node_(5) set_next_hop 0

set a_(5) [new Agent/UWSink]
$ns_ attach-agent $node_(5) $a_(5)
$a_(5) attach-vectorbasedforward $opt(width)
$a_(5) cmd set-range 40
$a_(5) cmd set-target-x 0
$a_(5) cmd set-target-y 0
$a_(5) cmd set-target-z 0
$a_(5) set data_rate_ 0.05

set node_(6) [$ns_ node 6]
$node_(6) set sinkStatus_ 1
```

```
$god_ new_node $node_(6)
$node_(6) set X_ -33.2793
$node_(6) set Y_ -32.07
$node_(6) set Z_ 0
$node_(6) set-cx 0
$node_(6) set-cy -50
$node_(6) set-cz 0
$node_(6) set_next_hop 0

set a_(6) [new Agent/UWSink]
$ns_ attach-agent $node_(6) $a_(6)
$a_(6) attach-vectorbasedforward $opt(width)
$a_(6) cmd set-range 40
$a_(6) cmd set-target-x 0
$a_(6) cmd set-target-y 0
$a_(6) cmd set-target-z 0
$a_(6) set data_rate_ 0.05

set node_(7) [$ns_ node 7]
$node_(7) set sinkStatus_ 1

$god_ new_node $node_(7)
$node_(7) set X_ 33.5318
$node_(7) set Y_ -32.07
$node_(7) set Z_ 0
$node_(7) set-cx 0
$node_(7) set-cy -50
$node_(7) set-cz 0
$node_(7) set_next_hop 0

set a_(7) [new Agent/UWSink]
$ns_ attach-agent $node_(7) $a_(7)
$a_(7) attach-vectorbasedforward $opt(width)
$a_(7) cmd set-range 40
$a_(7) cmd set-target-x 0
$a_(7) cmd set-target-y 0
$a_(7) cmd set-target-z 0
```

```

$a_(7) set data_rate_ 0.05

set node_($total_number) [$ns_ node $total_number]

$god_ new_node $node_($total_number)
$node_($total_number) set X_ 60
$node_($total_number) set Y_ 0
$node_($total_number) set Z_ 0
$node_($total_number) set -cx 60
$node_($total_number) set -cy 0
$node_($total_number) set -cz 0
$node_($total_number) set_next_hop 0

set a_($total_number) [new Agent/UWSink]
$ns_ attach-agent $node_($total_number) $a_($total_number)
$a_($total_number) attach-vectorbasedforward $opt(width)
$a_($total_number) cmd set -range 20
$a_($total_number) cmd set -target -x 0
$a_($total_number) cmd set -target -y 0
$a_($total_number) cmd set -target -z 0
$a_($total_number) set data_rate_ 0.05

$ns_ at $start_time "$a_($total_number) exp-start"
$ns_ at $start_time "$a_(1) exp-start"
$ns_ at $start_time "$a_(2) exp-start"
$ns_ at $start_time "$a_(3) exp-start"
$ns_ at $start_time "$a_(4) exp-start"
$ns_ at $start_time "$a_(5) exp-start"
$ns_ at $start_time "$a_(6) exp-start"
$ns_ at $start_time "$a_(7) exp-start"

set node_size 11
for {set k 0} { $k<$opt(nn) } { incr k } {
    $ns_ initial_node_pos $node_($k) $node_size
}

puts "++++++AFTER ANNOUNCE++++++"

$ns_ at $opt(stop).001 "$a_(0) terminate"

```

```

$ns_ at $opt(stop).002 "$a_($total_number) terminate"
$ns_ at $opt(stop).002 "$a_(1) terminate"
$ns_ at $opt(stop).002 "$a_(2) terminate"
$ns_ at $opt(stop).002 "$a_(3) terminate"
$ns_ at $opt(stop).002 "$a_(4) terminate"
$ns_ at $opt(stop).002 "$a_(5) terminate"
$ns_ at $opt(stop).002 "$a_(6) terminate"
$ns_ at $opt(stop).002 "$a_(7) terminate"

$ns_ at $opt(stop).003 "$god_ compute_energy"
$ns_ at $opt(stop).004 "$ns_ nam-end-wireless $opt(stop)"
$ns_ at $opt(stop).005 "puts \"NS EXISTING...\"; $ns_ halt"

puts $tracefd "vectorbased"
puts $tracefd "M 0.0 nn $opt(nn) x $opt(x) y $opt(y) z $opt(z)"
puts $tracefd "M 0.0 prop $opt(prop) ant $opt(ant)"
puts "starting Simulation..."
proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam rmacl.nam &
    exit 0
}

$ns_ run

```



## APÊNDICE C – TOPOLOGIA LINHA

```

set opt(chan)                Channel/UnderwaterChannel
set opt(prop)                Propagation/UnderwaterPropagation

set opt(netif)              Phy/UnderwaterPhy
set opt(mac)                Mac/UnderwaterMac/RMac
set opt(ifq)                Queue/DropTail/PriQueue
set opt(ll)                 LL
set opt(energy)             EnergyModel
set opt(txpower)            0.6
set opt(rxpower)            0.3
set opt(initialenergy)      10000
set opt(idlepower)          0.01
set opt(ant)                Antenna/OmniAntenna
set opt(filters)            GradientFilter

set opt(bit_rate)           1.0e4
set opt(encoding_efficiency) 1
set opt(ND_window)          1
set opt(ACKND_window)       1
set opt(PhaseOne_window)    3
set opt(PhaseTwo_window)    1
set opt(PhaseTwo_interval)  0.5
set opt(IntervalPhase2Phase3) 1
set opt(duration)           0.1
set opt(PhyOverhead)        8
set opt(large_packet_size)  480
set opt(short_packet_size)  40
set opt(PhaseOne_cycle)     4
set opt(PhaseTwo_cycle)     2
set opt(PeriodInterval)     1
set opt(transmission_time_error) 0.0001

set opt(dz)                 10
set opt(ifqlen)             50
set opt(nn)                  9
set opt(layers)              1

```

```

set opt(x) 100
set opt(y) 100
set opt(z) [expr ($opt(layers)-1)*$opt(dz)]
set opt(seed) 348.88
set opt(stop) 1000
set opt(prestop) 20
set opt(tr) "rmac3.tr"
set opt(nam) "rmac3.nam"
set opt(adhocRouting) Vectorbasedforward
set opt(width) 20
set opt(adj) 10
set opt(interval) 0.001

```

```

LL set mindelay_ 50us
LL set delay_ 25us
LL set bandwidth_ 0

```

```

Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Z_ 0.05
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

```

```

Mac/UnderwaterMac set bit_rate_ $opt(bit_rate)
Mac/UnderwaterMac set encoding_efficiency_ $opt(encoding_efficiency)
Mac/UnderwaterMac/RMac set ND_window_ $opt(ND_window)
Mac/UnderwaterMac/RMac set ACKND_window_ $opt(ACKND_window)
Mac/UnderwaterMac/RMac set PhaseOne_window_ $opt(PhaseOne_window)
Mac/UnderwaterMac/RMac set PhaseTwo_window_ $opt(PhaseTwo_window)
Mac/UnderwaterMac/RMac set PhaseTwo_interval_ $opt(PhaseTwo_interval)
Mac/UnderwaterMac/RMac set
IntervalPhase2Phase3_ $opt(IntervalPhase2Phase3)
Mac/UnderwaterMac/RMac set duration_ $opt(duration)
Mac/UnderwaterMac/RMac set PhyOverhead_ $opt(PhyOverhead)
Mac/UnderwaterMac/RMac set large_packet_size_ $opt(large_packet_size)
Mac/UnderwaterMac/RMac set short_packet_size_ $opt(short_packet_size)

```

```

Mac/UnderwaterMac/RMac set PhaseOne_cycle_ $opt(PhaseOne_cycle)
Mac/UnderwaterMac/RMac set PhaseTwo_cycle_ $opt(PhaseTwo_cycle)
Mac/UnderwaterMac/RMac set PeriodInterval_ $opt(PeriodInterval)
Mac/UnderwaterMac/RMac set
transmission_time_error_ $opt(transmission_time_error)

```

```

Phy/UnderwaterPhy set CPThresh_ 100
Phy/UnderwaterPhy set CStresh_ 0
Phy/UnderwaterPhy set RXThresh_ 0
Phy/UnderwaterPhy set Pt_ 0.2818
Phy/UnderwaterPhy set freq_ 25
Phy/UnderwaterPhy set K_ 2.0

```

```

remove-all-packet-headers
add-packet-header IP Mac LL ARP UWVB RMAC

```

```

set ns_ [new Simulator]
set topo [new Topography]

```

```

$topo load_cubicgrid $opt(x) $opt(y) $opt(z)
set tracefd [open $opt(tr) w]
$ns_ trace-all $tracefd

```

```

set nf [open $opt(nam) w]
$ns_ namtrace-all-wireless $nf $opt(x) $opt(y)

```

```

set phase1_time [expr $opt(PhaseOne_cycle)*$opt(PhaseOne_window)]
set phase2_time [expr $opt(PhaseTwo_cycle)*($opt(PhaseTwo_window)+
$opt(PhaseTwo_interval))]
set start_time [expr $phase1_time+$phase2_time+$opt
(IntervalPhase2Phase3)]

```

```

puts "the start time is $start_time"

```

```

set total_number [expr $opt(nn)-1]
set god_ [create-god $opt(nn)]

```

```
set chan_1_ [new $opt(chan)]
```

```
global defaultRNG
```

```
$defaultRNG seed $opt(seed)
```

```
$ns_ node-config --adhocRouting $opt(adhocRouting) \  
                --llType $opt(ll) \  
                --macType $opt(mac) \  
                --ifqType $opt(ifq) \  
                --ifqLen $opt(ifqlen) \  
                --antType $opt(ant) \  
                --propType $opt(prop) \  
                --phyType $opt(netif) \  
                --agentTrace OFF \  
                --routerTrace OFF \  
                --macTrace ON\  
                --topoInstance $topo\  
                --energyModel $opt(energy)\  
                --txpower $opt(txpower)\  
                --rxpower $opt(rxpower)\  
                --initialEnergy $opt(initialenergy)\  
                --idlePower $opt(idlepower)\  
                --channel $chan_1_
```

```
set node_(0) [$ns_ node 0]
```

```
$node_(0) set sinkStatus_ 1
```

```
$node_(0) set passive 1
```

```
$ns_ at 0.0 "$node_(0) label COLECTOR"
```

```
$ns_ at 0.0 "$node_(0) add-mark n1 blue hexagon"
```

```
$god_ new_node $node_(0)
```

```
$node_(0) set X_ 14
```

```
$node_(0) set Y_ 44
```

```
$node_(0) set Z_ 0.0
```

```
$node_(0) set passive 1
```

```
set a_(0) [new Agent/UWSink] ;
$ns_ attach-agent $node_(0) $a_(0)
$a_(0) attach-vectorbasedforward $opt(width)
$a_(0) cmd set-range 20
$a_(0) cmd set-target-x -20
$a_(0) cmd set-target-y -10
$a_(0) cmd set-target-z -20
```

```
set node_(1) [$ns_ node 1]
$node_(1) set sinkStatus_ 1
```

```
$god_ new_node $node_(1)
$node_(1) set X_ -60
$node_(1) set Y_ 21
$node_(1) set Z_ 0
$node_(1) set-cx 0
$node_(1) set-cy -15
$node_(1) set-cz 0
$node_(1) set_next_hop 0
```

```
set a_(1) [new Agent/UWSink]
$ns_ attach-agent $node_(1) $a_(1)
$a_(1) attach-vectorbasedforward $opt(width)
$a_(1) cmd set-range 20
$a_(1) cmd set-target-x 0
$a_(1) cmd set-target-y 0
$a_(1) cmd set-target-z 0
$a_(1) set data_rate_ 0.05
```

```
set node_(2) [$ns_ node 2]
$node_(2) set sinkStatus_ 1
```

```
$god_ new_node $node_(2)
$node_(2) set X_ -35
$node_(2) set Y_ 21
$node_(2) set Z_ 0
$node_(2) set-cx 0
```

```
$node_(2) set-cy 40
$node_(2) set-cz 0
$node_(2) set_next_hop 0

set a_(2) [new Agent/UWSink]
$ns_ attach-agent $node_(2) $a_(2)
$a_(2) attach-vectorbasedforward $opt(width)
$a_(2) cmd set-range 20
$a_(2) cmd set-target-x 0
$a_(2) cmd set-target-y 0
$a_(2) cmd set-target-z 0
$a_(2) set data_rate_ 0.05

set node_(3) [$ns_ node 3]
$node_(3) set sinkStatus_ 1

$god_ new_node $node_(3)
$node_(3) set X_ -16
$node_(3) set Y_ 21
$node_(3) set Z_ 0
$node_(3) set-cx 0
$node_(3) set-cy -30
$node_(3) set-cz 0
$node_(3) set_next_hop 0

set a_(3) [new Agent/UWSink]
$ns_ attach-agent $node_(3) $a_(3)
$a_(3) attach-vectorbasedforward $opt(width)
$a_(3) cmd set-range 20
$a_(3) cmd set-target-x 0
$a_(3) cmd set-target-y 0
$a_(3) cmd set-target-z 0
$a_(3) set data_rate_ 0.05

set node_(4) [$ns_ node 4]
$node_(4) set sinkStatus_ 1
```

```
$god_ new_node $node_(4)
$node_(4) set X_ 2
$node_(4) set Y_ 21
$node_(4) set Z_ 0
$node_(4) set-cx 0
$node_(4) set-cy -50
$node_(4) set-cz 0
$node_(4) set_next_hop 0

set a_(4) [new Agent/UWSink]
$ns_ attach-agent $node_(4) $a_(4)
$a_(4) attach-vectorbasedforward $opt(width)
$a_(4) cmd set-range 40
$a_(4) cmd set-target-x 0
$a_(4) cmd set-target-y 0
$a_(4) cmd set-target-z 0
$a_(4) set data_rate_ 0.05

set node_(5) [$ns_ node 5]
$node_(5) set sinkStatus_ 1

$god_ new_node $node_(5)
$node_(5) set X_ 24
$node_(5) set Y_ 21
$node_(5) set Z_ 0
$node_(5) set-cx 0
$node_(5) set-cy -50
$node_(5) set-cz 0
$node_(5) set_next_hop 0

set a_(5) [new Agent/UWSink]
$ns_ attach-agent $node_(5) $a_(5)
$a_(5) attach-vectorbasedforward $opt(width)
$a_(5) cmd set-range 40
$a_(5) cmd set-target-x 0
$a_(5) cmd set-target-y 0
$a_(5) cmd set-target-z 0
$a_(5) set data_rate_ 0.05
```

```
set node_(6) [$ns_ node 6]
$node_(6) set sinkStatus_ 1

$god_ new_node $node_(6)
$node_(6) set X_ 44
$node_(6) set Y_ 21
$node_(6) set Z_ 0
$node_(6) set-cx 0
$node_(6) set-cy -50
$node_(6) set-cz 0
$node_(6) set_next_hop 0

set a_(6) [new Agent/UWSink]
$ns_ attach-agent $node_(6) $a_(6)
$a_(6) attach-vectorbasedforward $opt(width)
$a_(6) cmd set-range 40
$a_(6) cmd set-target-x 0
$a_(6) cmd set-target-y 0
$a_(6) cmd set-target-z 0
$a_(6) set data_rate_ 0.05

set node_(7) [$ns_ node 7]
$node_(7) set sinkStatus_ 1

$god_ new_node $node_(7)
$node_(7) set X_ 62
$node_(7) set Y_ 21
$node_(7) set Z_ 0
$node_(7) set-cx 0
$node_(7) set-cy -50
$node_(7) set-cz 0
$node_(7) set_next_hop 0

set a_(7) [new Agent/UWSink]
$ns_ attach-agent $node_(7) $a_(7)
$a_(7) attach-vectorbasedforward $opt(width)
$a_(7) cmd set-range 40
```



```

$a_(7) cmd set-target-x 0
$a_(7) cmd set-target-y 0
$a_(7) cmd set-target-z 0
$a_(7) set data_rate_ 0.05

set node_($total_number) [$ns_ node $total_number]

$god_ new_node $node_($total_number)
$node_($total_number) set X_ 86
$node_($total_number) set Y_ 21
$node_($total_number) set Z_ 0
$node_($total_number) set-cx 60
$node_($total_number) set-cy 0
$node_($total_number) set-cz 0
$node_($total_number) set_next_hop 0

set a_($total_number) [new Agent/UWSink]
$ns_ attach-agent $node_($total_number) $a_($total_number)
$a_($total_number) attach-vectorbasedforward $opt(width)
$a_($total_number) cmd set-range 20
$a_($total_number) cmd set-target-x 0
$a_($total_number) cmd set-target-y 0
$a_($total_number) cmd set-target-z 0
$a_($total_number) set data_rate_ 0.05

$ns_ at $start_time "$a_($total_number) exp-start"
$ns_ at $start_time "$a_(1) exp-start"
$ns_ at $start_time "$a_(2) exp-start"
$ns_ at $start_time "$a_(3) exp-start"
$ns_ at $start_time "$a_(4) exp-start"
$ns_ at $start_time "$a_(5) exp-start"
$ns_ at $start_time "$a_(6) exp-start"
$ns_ at $start_time "$a_(7) exp-start"

set node_size 11
for {set k 0} {$k<$opt(nn)} {incr k} {
    $ns_ initial_node_pos $node_($k) $node_size
}

```

```

puts "++++++AFTER ANNOUNCE++++++"

$ns_ at $opt(stop).001 "$a_(0) terminate"
$ns_ at $opt(stop).002 "$a_($total_number) terminate"
$ns_ at $opt(stop).002 "$a_(1) terminate"
$ns_ at $opt(stop).002 "$a_(2) terminate"
$ns_ at $opt(stop).002 "$a_(3) terminate"
$ns_ at $opt(stop).002 "$a_(4) terminate"
$ns_ at $opt(stop).002 "$a_(5) terminate"
$ns_ at $opt(stop).002 "$a_(6) terminate"
$ns_ at $opt(stop).002 "$a_(7) terminate"

$ns_ at $opt(stop).003 "$god_ compute_energy"
$ns_ at $opt(stop).004 "$ns_ nam-end-wireless $opt(stop)"
$ns_ at $opt(stop).005 "puts \"NS EXISTING...\"; $ns_ halt"

puts $tracefd "vectorbased"
puts $tracefd "M 0.0 nn $opt(nn) x $opt(x) y $opt(y) z $opt(z)"
puts $tracefd "M 0.0 prop $opt(prop) ant $opt(ant)"
puts "starting Simulation..."
proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam rmac3.nam &
    exit 0
}

$ns_ run

```