

UNIVERSIDADE FEDERAL DE ALAGOAS

INSTITUTO DE COMPUTAÇÃO

Coordenação de Pós-Graduação em Modelagem Computacional do Conhecimento

CÉSAR FELIPE GONÇALVES DA SILVA

Análise de otimização de roteamento IoT em redes LLN's para cidades inteligentes

Um estudo empírico sobre a utilização do método de enxame ACO em ambientes IoT para cidades inteligentes

Maceió/AL

Dezembro de 2020

CÉSAR FELIPE GONÇALVES DA SILVA

Análise de otimização de roteamento IoT em redes LLN's para cidades inteligentes

Um estudo empírico sobre a utilização do método de enxame ACO em ambientes IoT para cidades inteligentes

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Modelagem Computacional do Conhecimento da Universidade Federal de Alagoas, como requisito para obtenção do grau de mestre.

Orientador: Prof. Dr. Rafael de Amorim Silva

Maceió/AL

Dezembro de 2020

Catálogo na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico

Bibliotecário: Marcelino de Carvalho Freitas Neto – CRB-4 – 1767

S586a Silva, César Felipe Gonçalves da.
Análise de otimização de roteamento Iot em redes LLN's para cidades inteligentes :
um estudo empírico sobre a utilização do método de enxame ACO em ambientes Iot para
cidades inteligentes / César Felipe Gonçalves da Silva. – 2020.
98 f. : il.

Orientadora: Rafael de Amorim Silva.
Dissertação (mestrado em Modelagem Computacional do Conhecimento) –
Universidade Federal de Alagoas. Instituto de Computação. Maceió, 2019.

Inclui glossário.
Bibliografia: f. 52-56.
Apêndices: f. 59-98.

1. Internet das coisas. 2. Colônia de formigas (Algoritmo). 3. Cidades inteligentes. 4.
Eficiência energética. 5. Sistemas de comunicação sem fio. I. Título.

CDU: 004.73



Ata da defesa de dissertação do aluno

César Felipe Gonçalves da Silva

Realizou-se no dia 20 de Dezembro de 2019, a partir das 10 h 00, na Sala de reuniões do Instituto de Computação da Universidade Federal de Alagoas, a defesa de dissertação de Mestrado em Modelagem Computacional de Conhecimento, intitulada "Análise de Otimização de Roteamento IoT em redes LLN's para Cidades Inteligentes", apresentada por César Felipe Gonçalves da Silva, Graduado em Gestão de Infraestrutura de Redes, como requisito parcial para a obtenção do grau de Mestre em Modelagem Computacional de Conhecimento, perante a seguinte comissão examinadora:

Professor e orientador Rafael de Amorim Silva
Instituto de Computação – UFAL

Professor Diego Dermeval Medeiros da Cunha Matos
Faculdade de Medicina – UFAL

Professor Bruno Almeida Pimentel
Instituto de Computação – UFAL



Após a apresentação, o aluno foi arguido pelos examinadores que, consideraram a dissertação:

Aprovada () Reprovada.

Observações:

CONSIDERAR AS SUGESTÕES DA BANCA EXAMINADORA

Nada mais havendo a tratar, a sessão foi encerrada às 12 h 20 dela sendo lavrado a presente ata, que segue assinada pela Banca Examinadora.

Maceió, 20 de Dezembro de 2019.

Rafael de Amorim Silva

Rafael de Amorim Silva
Doutor em Engenharia Eletrônica – Instituto Tecnológico de Aeronáutica/Brasil

Diego Dermeval Medeiros da Cunha Matos

Diego Dermeval Medeiros da Cunha Matos
Doutor em Ciências da Computação – Universidade Federal de Campina Grande/Brasil

Bruno Almeida Pimentel

Bruno Almeida Pimentel
Doutor em Ciências da Computação – Universidade Federal de Pernambuco/Brasil

RECEBIDO COM O ORIGINAL

DATA: 20 / 03 / 2020

Andre Luiz de Aguiar
Andre Luiz de Aguiar
Sicopel 1647956

À DEUS e toda sua criação.

À minha toda Família com muito amor, por ter suportado e compreendido minha ausência ao longo destes últimos anos.

Aos meus filhos, alunos e toda humanidade um exemplo de que a esperança de dias melhores somente é construída através dos estudos, dedicação, disciplina, abdicção, amor, respeito e trabalho honesto.

AGRADECIMENTOS

À DEUS pelo dom da vida.

Aos meus amados pais Nilson Cleomar e Marizilda Peixoto pelos valores que me ensinaram e pelo amor que sempre recebi, sacrifício por meu sustento e minha educação.

À minha amada irmã Flávia Cristina pelo respeito e amizade.

À minha amada esposa Daiane Oliveira pela compreensão pelo longo período de minha presente ausência

Aos meus amados filhos, com os quais tanto me preocupo em dar-lhes bons exemplos e a esperança de dias melhores

Às minhas amadas crianças de quatro patas: tetezinha (*in memoriam*), Luna, Tom, Mia, Melody e os dois baixinhos por tornarem nossas vidas mais leves ao longo de todos estes anos.

A todos colegas do Núcleo de tecnologia da informação da Justiça Federal de Alagoas pela compreensão pelos meus momentos de ausência para os estudos e por dar-me a honra de desfrutar de suas ricas e agradáveis companhias diariamente.

Ao meu orientador Prof. Dr. Rafael de Amorim Silva por me acompanhar, indicar o melhor caminho ao longo desta trajetória e mostra-se um grande incentivador.

Aos Profs. Drs. Bruno de Almeida Pimentel e Diego Dermeval Medeiros da Cunha Matos não só pela educação e respeito durante a etapa de qualificação, mas, também, pelas ricas dicas que abrilhantaram este trabalho.

A todos os professores deste programa de pós-graduação pelos ensinamentos repassados ao longo desta jornada.

Durante esta caminhada, cada linha de código e cada “enter” tornavam-se um obstáculo. Todas vezes eu caía, todas eu me levantava. Os obstáculos viram que eu seria imbatível e desistiram. Eu venci!

RESUMO

Internet das Coisas é uma rede de dispositivos que possuem sensores e que tem poder de processamento, armazenamento, capacidade de interconexão e alguma inteligência. Inúmeras tecnologias IoT têm possibilitado a redução significativa de tarefas repetitivas, um tráfego rápido de dados em uma rede de comunicação, um melhor armazenamento e organização de informações, um monitoramento efetivo de ambientes diversos, entre outros benefícios a sociedade. Quando aplicada a centros urbanos, dispositivos IoT permitem o monitoramento individual de indicadores de saúde, qualidade e umidade do ar, temperatura ambiente, pressão atmosférica, tráfego terrestre e aéreo, monitoramento do solo, da agricultura e pecuária, serviços de infraestrutura pública, emissão de alertas de perigo, dentre outros. A aplicação de dispositivos IoT nestes centros urbanos é essencial para caracterizar as Smart Cities (cidades inteligentes), um conceito moderno que busca otimizar o uso destas tecnologias e a gestão do espaço urbano. Normalmente, tais dispositivos têm escassa capacidade energética, de processamento, armazenamento e conectividade e um dos principais desafios científicos da atualidade é tentar otimizar o uso destes recursos de forma inteligente, aplicando técnicas como ACO (Ant Colony Optimization), baseadas no micromundo das formigas. Portanto, a proposta deste trabalho é realizar uma análise detalhada da otimização do uso energético dos nós de uma rede IoT provida pelo método de enxame ACO, com o emprego de simulações, coleta de dados e respectiva análise para entender se sua aplicação proporciona aos dispositivos investigados um aumento efetivo no tempo de vida útil destes dispositivos em uma rede IoT. Os resultados obtidos deste estudo servirão como base para melhoria dos mecanismos empregados nos eventos de roteamentos presentes em cenários IoT.

Palavras-Chave: Internet das Coisas, Otimização por Colônia de Formigas, Cidades Inteligentes, Eficiência Energética, Rede de Sensor Sem Fio.

ABSTRACT

Internet of Things is a network of sensor-enabled devices that have processing power, storage, interconnectivity, and some intelligence. Numerous IoT technologies have made it possible to significantly reduce repetitive tasks, fast data traffic on a communication network, better storage and organization of information, effective monitoring of diverse environments, and other benefits to society. When applied to urban centers, IoT devices allow individual monitoring of indicators of health, air quality and humidity, ambient temperature, atmospheric pressure, land and air traffic, soil, agriculture and livestock monitoring, public infrastructure services, hazard alerts, among others. The application of IoT devices in these urban centers is essential to characterize Smart Cities, a modern concept that seeks to optimize the use of these technologies and the management of urban space. Typically, such devices have scarce power, processing, storage, and connectivity capabilities, and one of today's key scientific challenges is trying to intelligently optimize the use of these resources by applying techniques such as Ant Colony Optimization (ACO), based on ant microworlds. Therefore, the purpose of this work is to perform a detailed analysis of the energy use optimization of the nodes of an IoT network provided by the ACO swarm method, using simulations, data collection and respective analysis to understand if its application provides the investigated devices. an effective increase in the lifetime of these devices on an IoT network. The results obtained from this study will serve as a basis for improving the mechanisms employed in routing events present in IoT scenarios.

Keywords: Internet of Things, Ant Colony Optimization, Smart Cities, Energy Efficiency, Wireless Sensor Networks.

LISTA DE FIGURAS

Figura 1 - Smart city and services	17
Figura 2 - Funcionamento de um protocolo de roteamento	22
Figura 3 – Colônia de formigas ilustrativa (fonte: Wikipédia)	30
Figura 4 - Colônia de formigas reais (fonte: Wikipédia)	31
Figura 5 - Lógica de seleção roulette wheel	34
Figura 6 - Primeiro estado da rede para simulação sem ACO	43
Figura 7 - 17º e último estado da rede para simulação sem ACO	44
Figura 8 - Resumo de conexões da rede.....	45
Figura 9 - Primeiro estado da rede para simulação com ACO	45
Figura 10 - Pacotes transmitidos totais sem ACO X primeira indisponibilidade ACO	46
Figura 11 - Pacotes transmitidos antes da primeira indisponibilidade sem e com ACO.....	47

LISTA DE TABELAS

Tabela 1 – Cálculo de probabilidade ACO.....	32
Tabela 2 - Exemplificação da seleção roulette wheel	34
Tabela 3 - Log de eventos da rede de 70 nós sem ACO.....	43
Tabela 4 - Totais energéticos.....	100

LISTA DE SIGLAS E ABREVIATURAS

ACO - Ant Colony Optimization ou Otimização de colônia de formigas

IP – Internet Protocol

IoT - Internet of Things ou Internet das coisas

LLN - Low-Energy and Loozy Network ou Redes lentas e com pouca energia

MANET - Mobile Ad-hoc Network ou Redes móveis

PDR - Packet Delivery Ratio ou Taxa de entrega de pacotes

PLR - Packet Loss Ratio ou Taxa de perda de pacotes

QoS - Quality of Service ou Qualidade do serviço

WSN - Wireless Sensor Network ou Rede de sensores sem fio

SUMÁRIO

1. INTRODUÇÃO	14
1.1 Contextualização	14
1.2 Problemática	14
1.3 Proposta	15
1.4 Metodologia	16
1.5 Estrutura do Trabalho	16
2. FUNDAMENTAÇÃO TEÓRICA	17
2.1 Roteamento em Rede IoT	18
2.2 Eficiência Energética	20
3. RELATO DO PROBLEMA	21
3.1 Trabalhos Relacionados	25
4. PROPOSTA	27
4.1 Inteligência de Enxame	28
4.2 Algoritmo ACO	29
4.3 Relevância da proposta	35
5. SIMULAÇÃO	36
5.1 Cenário	36
5.2 Metodologia	37
5.3 Parâmetros	38
6. RESULTADOS	43
6.1 Discussão dos Resultados	47
6.2 Ameaças à validade	48
6.3 Ameaças internas	48
6.4 Ameaças externas	49
7. CONSIDERAÇÕES FINAIS	50
REFERÊNCIAS BIBLIOGRÁFICAS	52
GLOSÁRIO	55
Apêndice A	57
SIMULAÇÃO EM MATLAB	57

Arquivo main.m	57
Arquivo aco.m.....	65
Arquivo createColony.m	72
Arquivo rouletteWheel.m.....	72
Arquivo fitnessFunction.m.....	72
Apêndice B	74
ASPECTOS DA SIMULAÇÃO	74
Apêndice C	100
RESULTADOS ENERGÉTICOS DA SIMULAÇÃO	100

1. INTRODUÇÃO

1.1 Contextualização

A urbanização é um fenômeno em expansão, responsável por inserir mais da metade da população global em centros urbanos. Durante o último século, a população mundial quadruplicou e houve forte migração de pessoas da área rural para a urbana. Atualmente, 50% da população mundial vive nas cidades e a expectativa é de que este número chegue a 70% por volta de 2050 [1]. Impulsionados pela globalização, a melhora da qualidade de vida, gestão dos serviços públicos e a otimização da tecnologia, tais centros devem prover moradia, serviços de saneamento, cuidados com a saúde, educação e segurança a esta população [2]. Em paralelo, os últimos anos têm testemunhado o aparecimento das cidades inteligentes, com desafios que podem ser resolvidos com uso de tecnologia.

A aplicação de tecnologia IoT baseada em dispositivos independentes e interconectados que executam tarefas específicas já é realidade em centros urbanos desenvolvidos. Entretanto, o crescimento de equipamentos inteligentes de rede que possuam capacidade de interconexão para transmissão de dados tem se tornado um dos maiores desafios científicos da atualidade. Podemos citar, por exemplo, o uso de WSN para estabelecer uma rede de comunicação entre tais dispositivos. Desta forma, componentes de uma WSN são providos de baterias e o uso eficiente do tempo de vida destes componentes é um assunto largamente discutido no meio científico[3]. Com o número crescente de dispositivos interconectados e o aumento dos requisitos para estas conexões, aspectos como tempo de bateria, custo de envio e cobertura são cruciais para garantir a comunicação eficiente ponta-a-ponta em um cenário IoT [4].

1.2 Problemática

Neste contexto de cidades inteligentes, ambientes IoT apresentam problemas que precisam ser pesquisados e resolvidos [5]. Por exemplo, estes ambientes geralmente são compostos por uma gama de dispositivos heterogêneos[6], equipados com sensores utilizados

para captar dados do mundo exterior e, também, capazes de interconectar-se [7] para que os dados sejam encaminhados de um ponto a outro com o mínimo de intervenção humana[8]. No entanto, devido à restrição de recursos dos dispositivos IoT [9], é necessário estender o máximo possível o tempo de vida energética destes para garantir a perpetuação do fluxo de dados entre origem e destino através de tarefas de encaminhamento [10]. Tais tarefas devem utilizar informações de roteamento construídas de acordo com informações de rotas resultantes dos protocolos em uso na rede [9].

Durante o andamento da pesquisa, foram encontrados relatos referentes à preocupação em proporcionar à rede o aumento de sua vida útil, principalmente no que diz respeito à otimização do consumo de energia ocasionado pelos eventos de roteamento, atividade que utiliza mecanismo arbitrário de definição da rota a ser utilizada, pois os mecanismos de decisão mais utilizados atualmente não são adequados para redes do tipo WSN, pois apresentam desperdício de energia do conjunto para execução das tarefas de roteamento [11], sendo este desgaste patrocinado pelo mal gerenciamento dos recursos da rede, principalmente o energético, que pode ser atenuado com a utilização de algum mecanismo de balanceamento da utilização dos recursos [12].

1.3 Proposta

Esta dissertação apresenta um estudo baseado na simulação de uma transmissão oportunística de dados de emergência entre um ponto emissor “a”, para um receptor “b” através de uma rede de nós IoT. São considerados, como nós desta rede, dispositivos IoT diversos como celulares, tablets, veículos e similares, que estariam disponíveis no momento da ocorrência. Implementa-se um cenário de IoT contrastando a utilização de protocolo de roteamento vetor-distância embarcado no MatLab (conhecido como *ShortestPath*), com a utilização deste protocolo vetor-distância em conjunto com a técnica de inteligência de enxame conhecida como ACO. A finalidade deste estudo é provar que a técnica ACO é capaz de diminuir a quantidade de nós mortos pelo exaurimento de sua capacidade energética, através de mudança de rota temporal e conveniente, resultante da análise do estado global de cada rota, mudando a rota para outra em melhor condição energética.

1.4 Metodologia

Neste trabalho, desenvolve-se uma simulação que utiliza codificação nativa do MATLAB para gerar uma rede baseada em parâmetros de entrada definidos pelo usuário, simulando assim o posicionamento de possíveis dispositivos IoT quando uma transmissão de emergência é necessária. Após a criação desta rede com base no alcance dos nós, o protocolo *ShortestPath* encontra a melhor rota, inicia a transmissão dos dados, subtraindo dos nós envolvidos energia relativa à tarefa de encaminhamento e acrescentando a todos nós existentes o montante energético proveniente de uma hipotética fonte de recarga, sempre menor que aquele gasto pela tarefa de encaminhamento.

Existindo a ocorrência de esgotamento da bateria do dispositivo envolvido, o nó é identificado e as vizinhanças anteriormente estabelecidas com este nó deixam de existir. Neste momento, o protocolo de roteamento busca rota alternativa e continua a transmissão, assim sucessivamente até que não existam mais rotas disponíveis, momento em que a simulação recomeça para o mesmo cenário, anteriormente analisado, agora com uso do *ShortestPath* para busca das rotas possíveis e do ACO para escolha de qual destas utilizar. Cada simulação é executada até que não existam mais possibilidades de caminho.

1.5 Estrutura do Trabalho

O restante desta dissertação está organizado da seguinte forma: o Capítulo 2 apresenta a fundamentação teórica necessária para a compreensão deste trabalho. O Capítulo 3 descreve os possíveis problemas encontrados no roteamento de dispositivos IoT. O Capítulo 4 apresenta a proposta desta dissertação, detalhando como o algoritmo ACO funciona e que ajustes foram feitos para melhorar o desempenho do roteamento destes dispositivos. O Capítulo 5 define os aspectos utilizados para a elaboração da simulação. O Capítulo 6 apresenta e discute os resultados obtidos na simulação. O Capítulo 7 apresenta as considerações finais desta dissertação. Por fim, o código fonte da simulação em MATLAB e o comportamento dos dados durante a simulação é apresentado nos Apêndices A e B.

2. FUNDAMENTAÇÃO TEÓRICA

Para muitas pessoas, um dispositivo IoT representa um *Smartphone*, um *Tablet*, um *Smartwatch*, um *SmartTV*, um *SmartCar*, um *Notebook*, um *Drone*, etc, e não passam de objetos para uso pessoal. No contexto de cidades inteligentes, estes dispositivos estão no núcleo das soluções para estes centros urbanos e tem diversos papéis importantes [13]. Dentre várias opções de aplicabilidade, estão aquelas com caráter de emergência como guia para evacuação rápida, guia de saída durante *blackouts* causados por incêndios, inundações, terremotos e outros [14].

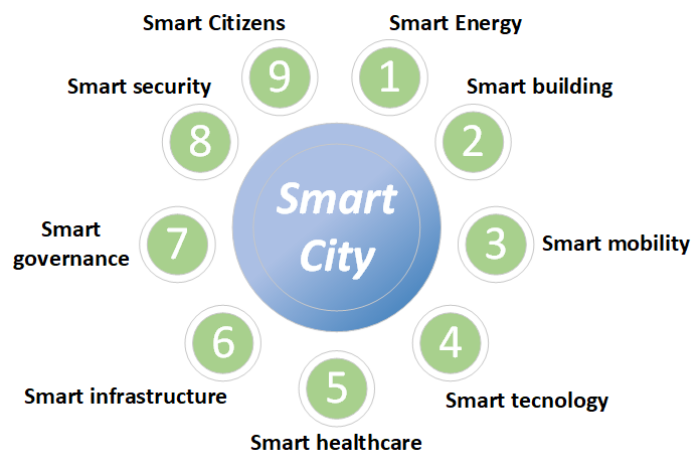


Figura 1 - Smart city and services (autoria própria)

O protótipo de IoT está diretamente relacionado a objetos inteligentes e autoconfiguráveis que se conectam uns aos outros através de uma rede mundial, ou através de uma rede ad-hoc, criando uma infraestrutura de rede de comunicação que possibilitará a troca de dados entre diversos sistemas [15]. O principal conceito de IoT é a presença universal de objetos que podem ser medidos, inferidos, compreendidos e que podem modificar o ambiente. Com base no que foi dito, IoT é a chave para o desenvolvimento de novos equipamentos e novas tecnologias de comunicação [15].

Dispositivos IoT podem ser integrados através de uma WSN. Esta rede é composta de nós sensores dispersos que analisam condições físicas e ambientais como temperatura e pressão e encaminham estas informações através de outros nós sensores até uma central recepção dos dados [14]. Podemos citar, como exemplo, os veículos aéreos não tripulados (UAVs) ou drones, dispositivos capazes de estabelecer uma infraestrutura de rede mais robusta e segura tais como redes ad hoc multidirecionais [16].

IoT cria uma rede de objetos físicos com as seguintes características [9]: (i) os nós sensores IoT conversam uns com os outros sem sistema físico de conexão, através de radiofrequência; (ii) objetos IoT são autoconfiguráveis e podem se adaptar a diversos cenários automaticamente; (iii) os dispositivos desta rede têm recursos limitados como memória, raio de alcance e bateria; e (iv) IoT é relevante para redes ad-hoc. A era da tecnologia da Internet inconscientemente migrou para a era da tecnologia de dados, com a proliferação de dispositivos móveis como celulares, tablets, dispositivos vestíveis, etc, com um total previsto de 50 bilhões de dispositivos em 2020 capazes de prestar os mais variados serviços [17].

Maximizar a vida útil da rede e otimizar o uso desta rede é um grande desafio [18]. Por exemplo, infraestruturas de utilidade pública geralmente apresentam falhas ou se tornam indisponíveis durante emergências ou desastres. Uma estrutura de comunicação torna-se crucial para um eficiente e bem-sucedido sistema de resposta. Um bom exemplo inclui o furacão Sandy (Outubro de 2012), que causou falta de energia e completo colapso do sistema de comunicação durante sua passagem. Assim, a existência de uma infraestrutura de comunicação alternativa disponibilizada através de dispositivos IoT disponíveis e operantes na área são de grande utilidade para proteção de civis e pessoal do corpo estratégico, tático e operacional, se tornando equipamentos capazes de oferecer transmissão de voz e dados através das áreas atingidas [19].

2.1 Roteamento em Rede IoT

O advento da Internet das coisas e suas aplicações têm atraído a atenção dos pesquisadores. Entretanto, as particularidades inerentes às limitações de recursos de *hardware* associadas a estes dispositivos têm indicado sérios desafios, principalmente relacionados aos

protocolos de roteamento [20]. As redes ad-hoc móveis, por exemplo, são a fundação do futuro da Internet devido à crescente demanda por mobilidade. Em paralelo, aplicações para *Smartphones* relacionadas à mobilidade, tais quais, rastreamento, sensoriamento, assistência pós-desastre e casas inteligentes, tem se tornado populares, pois tais aplicações utilizam o conceito de IoT.

Uma das grandes preocupações da comunidade científica recai sobre as questões referentes a eficiência de roteamento [9][9] Pacotes de dados são perdidos durante evento de convergência de rede devido à indisponibilidade de um caminho [9]. Neste sentido, esforços estão sendo direcionados pela literatura científica para interconectar a infraestrutura IoT, aumentando todo potencial que pode ser oferecido por esta tecnologia. A rede IoT é diferente das redes tradicionais e tem como forte característica recursos limitados como energia. Isto impõe desafios para o design de novas soluções para IoT, uma vez que as soluções atuais que atendem as redes atuais são inaplicáveis e este novo cenário [12]. WSN's tem diversas aplicações, como indústria e uso militar. As características particulares deste tipo de rede, como o tamanho pequeno dos dispositivos, baixo preço e conectividade, têm estimulado seu uso em todos os lugares. Entretanto, as características limitadas destes equipamentos, como baixa energia, memória e poder de processamento requerem que os protocolos de roteamento atuais sejam repensados [21].

De acordo com a escritório de reduções de risco da ONU (UNISDR), o impacto financeiro resultante de desastre natural ou produzido pelo homem é gigantesco. Segundo o relatório, até 2030 a média anual de perdas resultantes de catástrofes pode alcançar 415 bilhões de dólares. Estas perdas certamente seriam minimizadas com estratégia de comunicação a ser utilizada durante estas ocorrências. Como exemplo, duas crises recentes mostraram a importância de meios alternativos de comunicação podem ter durante operações de emergência. Em março de 2011, um terremoto de magnitude 9 causou um tsunami de 30 pés no Japão, o quarto maior da história. Seis reatores nucleares da usina de Fukushima foram afetados, 2 explodiram. A agência de desastre e fogo do Japão confirmaram 22 mil pessoas mortas ou desaparecidas. Durante as operações de resgate, equipes de emergência e o exército utilizaram tradicionais equipamentos de radiofrequência sem transmissão de dados e táxis locais foram autorizados a utilizar a faixa de frequência reservada para emergência para que auxiliassem nas

buscas. Em janeiro de 2010 um terremoto de magnitude 7.0 atingiu o Haiti e o mesmo cenário de resgate de repetiu. Durante as 24 horas seguintes ao desastre, as equipes de resgate ainda não tinham uma visão clara do todo. Em ambos os casos, as operações de resgate foram ineficientes sem um meio alternativo de transmissão de dados rapidamente disponível e operacional [22].

2.2 Eficiência Energética

A Internet vem sendo facilmente transformada de “Internet individual” para Internet das Coisas (IoT). Assim, é estimado que nos próximos anos mais de 50 bilhões de aparelhos estejam conectados com a Internet e interconectados. Entretanto, fazer com que estes dispositivos tenham utilização otimizada, de forma a aumentar seu tempo de presença na rede tem tido destaque, visto que a capacidade de armazenamento de energia destes dispositivos é pequena [11]. A natureza dos dispositivos IoT lhes impõe sérias restrições energéticas. Em geral, supõe-se que estes são remotamente monitorados, autônomos e vestíveis, com ou sem alta velocidade de mobilidade. Estas características são o motivo para sua baixa capacidade energética [23].

A conservação de energia é um dos problemas mais críticos de design em redes de sensores (WSN). Estas redes são compostas por diversos sensores cuja característica é a escassez de recursos. Em WSNs, os nós são obrigados a operar sob um cenário de recursos restritos. Especificamente, existe pouca energia para transmitir informações entre nós, que deve ser preservada para estender a vida útil da rede o máximo possível. Estes dispositivos são disponibilizados em áreas distantes para atividades diversas. Quando dispersos aleatoriamente em áreas sem infraestrutura, enfrentam diversos desafios como a rápida drenagem de sua carga energética, por exemplo [24]. A necessidade de recargas frequentes é um obstáculo e a solução para este problema é o desenvolvimento de um protocolo de roteamento mais eficiente [25].

3. RELATO DO PROBLEMA

Conforme fundamentações apresentadas na seção anterior, WSN's são redes compostas por dispositivos com recursos limitados, sendo o principal problema o recurso energético, devido à mobilidade dos equipamentos, o que impossibilita que sejam recarregados sempre que necessário. Assim, é oportuno que sejam estudadas soluções que otimizem a vida útil da rede. Pesquisas científicas geralmente relatam que a otimização do processo de consumo energético de um conjunto WSN é desejável, tendo em vista que a movimentação de vértices acarreta convergência da rede, ou seja, reorganização de rotas de acordo com o novo desenho formado por algum evento. Tal evento pode ser resultante da movimentação dos nós, que saem do alcance de vizinhos atuais, se aproximando com outros, formando novas vizinhanças e estabelecendo novos vértices de conexão, ou a indisponibilidade do nó na rede, ocasionada por esgotamento de seu recurso energético.

O esgotamento energético de um nó resulta em sua indisponibilidade para a rede e consequente diminuição de possíveis enlaces que ele formaria com dispositivos próximos, os quais poderiam vir a ser rotas alternativas em algum momento. Dentre vários motivos referentes ao esgotamento, está o consumo de energia referente à atividade de roteamento que um dispositivo está apto a executar. Em linhas gerais, os processos de roteamento seguem as seguintes etapas: 1) Através de sua antena, um dispositivo recebe um pacote para destino para o qual o equipamento tem caminho; 2) Todo pacote é recebido; 3) processado; 4) armazenado em buffer; 5) Sua tabela de roteamento é consultada para que saiba, dentre as rotas que tem conhecimento, qual daquelas que conhece é a melhor opção; 6) Novamente repassa o pacote para antena; 6) Retransmite o pacote para o próximo nó.

A tabela de roteamento de cada equipamento é construída pelo protocolo de roteamento em uso na rede, cuja sua finalidade é fazer com que os dispositivos informem uns aos outros quais destinos lhes é conhecido, acrescentando a cada rota conhecida um custo, que é o valor observado para a escolha da rota. Tal valor é gerado utilizando informações como número de saltos/hops (nós) até o destino e demais informações, como velocidade do link, etc [26].

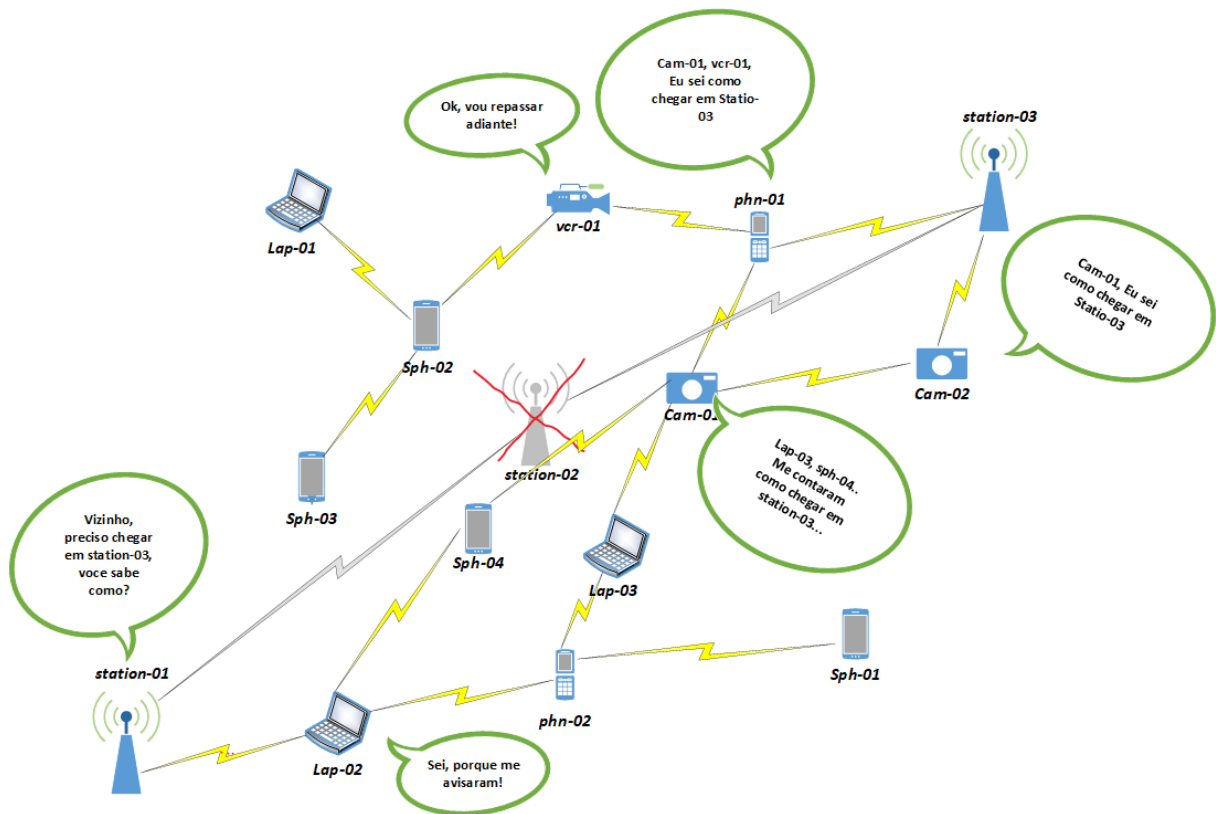


Figura 2 - Funcionamento de um protocolo de roteamento (autoria própria)

O processo descrito acima gera consumo de energia e o processo de consulta à tabela de roteamento para escolha do próximo salto segue, muitas vezes, regras arbitrárias e inflexíveis de escolha. Neste ponto, um protocolo vetor-distância, que usa quantidade de saltos, atribuirá baixo custo a uma rota com menos nós e esta informação não será modificada, fazendo com que ela seja utilizada arbitrariamente até que se torne indisponível pelo esgotamento de um dos roteadores. Abaixo, a tabela de roteamento dinamicamente construída de um celular motoG 5G2Plus com sistema operacional android 8.1.0 obtida com o comando `ip route show table 0`

```
default via 192.168.15.1 dev wlan0 table 1045 proto static
192.168.15.0/24 dev wlan0 table 1045 proto static scope link
default dev dummy0 table 1002 proto static scope link
192.168.15.0/24 dev wlan0 proto kernel scope link src 192.168.15.4
broadcast 127.0.0.0 dev lo table local proto kernel scope link src 127.0.0.1
local 127.0.0.0/8 dev lo table local proto kernel scope host src 127.0.0.1
```

```

local 127.0.0.1 dev lo table local proto kernel scope host src 127.0.0.1
broadcast 127.255.255.255 dev lo table local proto kernel scope link src 127.0.0.1
broadcast 192.168.15.0 dev wlan0 table local proto kernel scope link src 192.168.15.4
local 192.168.15.4 dev wlan0 table local proto kernel scope host src 192.168.15.4
broadcast 192.168.15.255 dev wlan0 table local proto kernel scope link src
192.168.15.4
unreachable default dev lo proto kernel metric 4294967295 error -101
2804:1b1:210b:db12::/64 dev wlan0 table 1045 proto kernel metric 256 expires
43165sec mtu 1460
2804:1b1:210b:db12::/64 dev wlan0 table 1045 proto static metric 1024
fe80::/64 dev wlan0 table 1045 proto kernel metric 256 mtu 1460
fe80::/64 dev wlan0 table 1045 proto static metric 1024
default via fe80::dafb:5eff:fe9a:69b6 dev wlan0 table 1045 proto ra metric 1024
expires 85sec mtu 1460 hoplimit 64
unreachable default dev lo proto kernel metric 4294967295 error -101
fe80::/64 dev dummy0 table 1002 proto kernel metric 256
default dev dummy0 table 1002 proto static metric 1024
unreachable default dev lo proto kernel metric 4294967295 error -101
local ::1 dev lo table local proto none metric 0
local 2804:1b1:210b:db12:d663:c6ff:fe82:c356 dev lo table local proto none metric 0
local 2804:1b1:210b:db12:d96c:ca46:d3af:4647 dev lo table local proto none metric
0
local fe80::14c7:b7ff:febd:efc3 dev lo table local proto none metric 0
local fe80::d663:c6ff:fe82:c356 dev lo table local proto none metric 0
ff00::/8 dev dummy0 table local metric 256
ff00::/8 dev wlan0 table local metric 256 mtu 1460

```

A tabela de roteamento de um notebook com sistema operacional Windows 10, obtida com o comando route print.

Tabela de rotas IPv4

Rotas ativas:

Endereço de rede	Máscara	Ender. gateway	Interface	Custo
0.0.0.0	0.0.0.0	192.168.15.1	192.168.15.20	10
0.0.0.0	0.0.0.0	192.168.15.1	192.168.15.5	35
127.0.0.0	255.0.0.0	No vínculo	127.0.0.1	331
127.0.0.1	255.255.255.255	No vínculo	127.0.0.1	331
127.255.255.255	255.255.255.255	No vínculo	127.0.0.1	331
192.168.15.0	255.255.255.0	No vínculo	192.168.15.5	291
224.0.0.0	240.0.0.0	No vínculo	127.0.0.1	331
224.0.0.0	240.0.0.0	No vínculo	192.168.56.1	281
224.0.0.0	240.0.0.0	No vínculo	192.168.15.5	291

Rotas persistentes:

Endereço de rede	Máscara	Ender. gateway	Custo
10.203.3.0	255.255.255.0	10.203.3.10	2
10.203.3.0	255.255.255.0	10.203.3.11	1

Assim, por exemplo, para as saídas de rotas persistentes (manualmente informadas) obtidas no *notebook*, onde as interfaces de rede cabeada e *wireless* estão conectadas, respectivamente com os IP's 10.203.3.10 e 10.203.3.11, a rota de saída para a rede de destino 10.203.3.0 terá prioridade pela interface 10.203.3.11, que pode estar interligada a um outro dispositivo IoT com baixa carga energética. Devido ao custo mais baixo que o da outra interface, a transmissão será feita pela interface *wireless* até que seu nó vizinho tenha sua energia esgotada pelo caráter arbitrário e inflexível do mecanismo de escolha da rota, que sempre optará pela de menor custo. Pelo exposto, um mecanismo de seleção de rota com visão geral sempre atualizada do contexto, que se sobreponha aos atuais mecanismos de seleção, é uma opção interessante a ser estudada.

3.1 Trabalhos Relacionados

Algoritmos bio-inspirados tem larga aplicação para escalabilidade, adaptabilidade, robustez e otimização de problemas complexos em WSN's. Otimização está presente em todos os setores da atividade humana, como organizar rotas aéreas para análises de negócios e finanças, e seleção de caminhos em redes *wireless* para a engenharia de arquitetura. Cada atividade de pesquisa na engenharia e ciência da computação considera o reconhecimento de padrões, análise de dados, simulações e otimização. A seleção do melhor algoritmo de otimização é um passo importante [5]. A escolha do algoritmo mais adequado necessita de poucos detalhes como comportamento do algoritmo, tipo de problema, soluções desejadas, recursos disponíveis e limites de tempo. Vários processos de decisão podem ser utilizados em conjunto com algoritmos bio-inspirados tais quais *Artificial Bee Colony* (ABC), *Genetic Algorithm* (GA), *Particle Swarm Optimization* (PSO) e muitos outros [27].

Modernos sistemas de defesa são desenvolvidos com uso de inteligência, como veículos terrestres não tripulados (UGVs), veículos aéreos não tripulados (UAVs), veículos de superfície não-tripulados (USVs) e veículos de submersão não tripulados (UUVs). Estes vêm se apresentando gradualmente como uma nova e inteligente rota de pesquisa. Sua grande vantagem é evitar o risco de casualidades. Estes equipamentos têm muitas vantagens como tamanho pequeno, resposta rápida e alta inteligência. Podem ser utilizados em operações de busca, resgate, observação, coleta de dados, transmissão e outras. Entretanto, devido à limitação imposta pela individualidade, o uso de colaboração multi-agente é necessário. Partindo desta perspectiva, de solução militar de defesa, tem importante valor o estudo do trabalho colaborativos entre equipamentos IoT para operações de emergência [28].

Recentes avanços nas tecnologias *wireless* têm atraído a atenção dos pesquisadores. Especificamente as redes móveis ad-hoc quando utilizadas em emergências, desastre e operações táticas. Devido à natureza flexível e escalável do IoT, ele executa algumas atividades apresentando degradação de performance. Roteamento é a principal função destas redes e o desenvolvimento de soluções de roteamento torna-se vital para as redes ad-hoc [29]. Recursos e agendamentos de tarefas para sistemas IoT tem sido estudados para minimizar o custo de processamento de dados imposto aos dispositivos. Para atingir este objetivo, é proposto

um mecanismo que mescla as funcionalidades de um algoritmo não preemptivo com algoritmo de enxame baseado em colônia de formigas (ACO) para priorizar as tarefas e encontrar os melhores caminhos [30].

As redes de sensores sem fio têm sido amplamente empregadas em cenários de IoT nos últimos anos. Estender o tempo de vida que todo este sistema tem sido um desafio significativo devido à exígua capacidade energética destes dispositivos. As estruturas de roteamento baseadas em clusters têm sido as soluções atualmente utilizadas e, efetivamente contribuem para a redução do consumo energético, aumentando a confiabilidade da rede. O estudo presente neste artigo introduz um algoritmo híbrido baseado em PSO e ACO e a pesquisa afirma que os testes executados demonstraram melhora na redução do consumo energético de todo o conjunto e aumentou o tempo de vida dos dispositivos [31].

4. PROPOSTA

A presente dissertação propõe o estudo de um mecanismo de análise constante do estado das opções de rotas existentes para a criação de uma visão geral do contexto que influencie na tomada de decisão sobre qual a melhor opção a ser escolhida. É sugerível a adoção de um mecanismo adaptativo, ou seja, que possa ser adequado para a análise em conjunto dos fatores que forem julgados importantes para a tomada de decisão e que não esteja atrelado ao mecanismo de escolha arbitrário presente em grande parte dos sistemas. Este mecanismo pode ser baseado em um modelo matemático capaz de suportar a definição de tantos critérios quantos forem os necessários para a escolha da melhor opção, com um fator de contrabalanceamento com finalidade de indicar o grau de relevância de cada um dos fatores indicados. Esta necessidade recai sobre o fato de que alguns requisitos devem ter sua razão inversamente proporcional ao que se deseja atingir, o que justifica o uso de um fator de contrabalanço.

Ante o exposto, é possível utilizar como exemplo um contador que informe quantas vezes uma rota vem sendo utilizada. A elevação constante deste contador indica uso excessivo da rota, o que indica um aumento na possibilidade de comprometimento energético dos nós presentes na rota devido à constante tarefa de roteamento. Assim, justificando o uso de um fator de contrabalanceamento para indicação de relevância de um valor dentro de um cálculo, a seguinte comparação é oferecida abaixo:

$$\text{Exemplo A: } R = \text{Valor}X^2 + \text{Valor}Y^3 > R = 2^2 + 3^2 > R = 4 + 9 > R = 13$$

$$\text{Exemplo B: } R = \text{Valor}X^2 + \text{Valor}Y^3 > R = 4^2 + 3^2 > R = 16 + 9 > R = 25$$

$$\text{Exemplo C: } R = \text{Valor}X^2 + \text{Valor}Y^3 > R = 2^{-2} + 3^2 > R = 0,25 + 9 > R = 9,25$$

$$\text{Exemplo D: } R = \text{Valor}X^2 + \text{Valor}Y^3 > R = 4^{-2} + 3^2 > R = 0,0625 + 9 > R = 9,0625$$

$$\text{Exemplo E: } R = \text{Valor}X^2 + \text{Valor}Y^3 > R = 4^{-3} + 3^2 > R = 0,0156 + 9 > R = 9,0156$$

Com os exemplos acima, fica matematicamente comprovado que uma das maneiras de identificar valores que são inversamente proporcionais ao resultado que se pretende obter é elevando estes a expoentes negativos que diminuirão suas relevâncias e, conseqüentemente, sua influência no resultado de uma equação qualquer. Nesta dissertação, a proposta é o uso de um modelo matemático adaptativo bio-inspirado em micromundo, especificamente o das formigas, conhecido como ACO (*Ant Colony Optimization*), classificada como técnica de inteligência de enxame, onde a estigmergia é fator preponderante.

Durante o levantamento do estado da arte, inicialmente os indicativos de que a eficiência energética era uma forte vertente de pesquisa chamou a atenção. O aprofundamento na pesquisa revelou que a utilização com visão unitária dos recursos de um ambiente levavam ao esgotamento antecipado do tempo de vida e que o uso de mecanismos que proporcionassem visão global dos recursos para a tomada de decisão é uma solução bastante pesquisada pela comunidade científica atual, tais engenhos são conhecidos como técnicas de inteligência de enxame, que utilizam modelos matemáticos com recebem dados do ambiente para execução de cálculos que retornam resultados que apontam, em tempo real, valores que podem ser utilizados para tomada de decisão.

4.1 Inteligência de Enxame

Algoritmos de inteligência de enxame são um conjunto de técnicas baseadas no comportamento social e coletivo de organismos organizados em forma de colônias, matilhas, enxames etc. Tais técnicas vem chamando a atenção das frentes de pesquisas para testes de eficiência na resolução automática de problemas sem a intervenção humana.

Inteligência de enxame se refere a um tipo de habilidade em solucionar problemas que nasce da interação entre simples unidades de processamento de informações. O conceito de enxame sugere multiplicidade, estocasticidade (resultado dependente de uma variável), aleatoriedade e diversidade de opções, onde o conceito de inteligência sugere um método que, de alguma forma, é bem-sucedido. As unidades de processamento que compõem um enxame podem ser animadas, mecânicas, computacionais ou matemáticas. Podem ser insetos, pássaros, humanos, conjuntos de elementos, robôs ou estações unitárias, também podendo ser reais ou

imaginárias. A união destes elementos pode conter uma grande variedade de características, mas deve haver interação entre eles [32].

Desenvolver múltiplos sistemas autônomos que coordenam uma ação de enxame no campo de batalha não é mais ficção. Como novas tecnologias têm modificado as características da guerra, as forças armadas americanas pesquisam algoritmos que possibilitem dispositivos independentes de coordenar táticas de enxame para cenários e fins diversos. Entretanto, os atuais *frameworks* em desenvolvimento utilizam táticas que são dependentes de um controle centralizado. Estes *frameworks* limitam a velocidade e flexibilidade do enxame, fazendo com que a eficiência do conjunto dependa de atributos que superam a capacidade humana de controle. Para contornar este obstáculo, o modo de guerra deve ser adaptável, as pesquisas das forças armadas devem explorar novas estratégias que possam rapidamente treinar os dispositivos de forma a torná-los eficientes e preparados para atuação descentralizada sem a intervenção humana, se livrando da lentidão que seria imposta pela dependência de ação humana [33].

Os existentes protocolos de roteamento para redes de sensores sem fio se preocupam, com eficiência energética, QoS ou segurança, entretanto, uma visão mais holística das WSN's é necessária, pois assim como as aplicações necessitam de garantias de QoS e segurança, a rede necessita de melhorias em seu consumo energético. Para resolver estes problemas, um estudo que mescla as funcionalidades do protocolo QEBSR (*QoS aware Energy Balancing Secure Routing*) com a inteligência do ACO, comparando os resultados com os obtidos pelo uso de outros dois protocolos (*distributed energy balanced routing and energy efficient routing with node compromised resistance*), indicaram que a combinação proposta neste estudo apresentaram melhores resultados do que os obtidos pelos dois outros protocolos [34].

4.2 Algoritmo ACO

O uso de inteligência de enxame é, relativamente, um assunto novo para se encontrar a solução para problemas dos mais diversos e é inspirado no comportamento social de insetos e outros animais. Particularmente, existem diversos métodos inspirados no micromundo dos

insetos, especificamente no das formigas, com diversas frentes de pesquisa referentes ao assunto, sendo o de maior aceitação e versatilidade o *Ant Colony Optimization* (ACO) [35].

O algoritmo em questão foi modelado com base na vida das formigas que procuram comida. Estas formigas deixam um rastro químico por onde passam, conhecido como feromônio que, com o decorrer do tempo, vai perdendo sua força devido ao processo de evaporação. Esta marca química é detectada por outras formigas e as atrai por um processo conhecido como estigmergia (atração). Assim, conforme um caminho, inicialmente escolhido de forma aleatória, começa a ser conhecido por ser o melhor caminho até a comida, a marca de feromônio se torna mais forte com o tempo e não consegue ser superado pelo evento de evaporação, a depender da taxa de evaporação e frequência com que as formigas passam por ali.

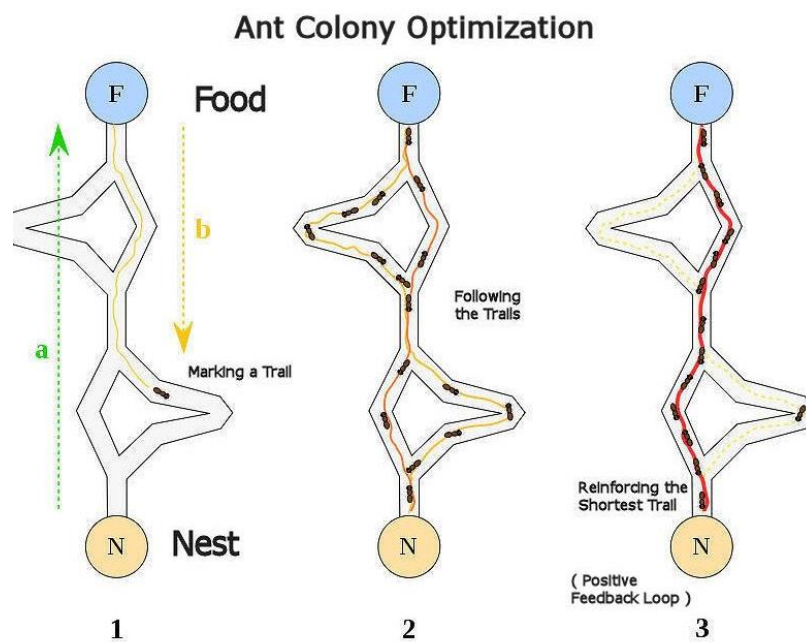


Figura 3 – Colônia de formigas ilustrativa (fonte: Wikipédia)



Figura 4 - Colônia de formigas reais (fonte: Wikipédia)

O sistema adotado pelo algoritmo do ACO é composto de três fases, sendo: 1) O cálculo de probabilidade referente a cada objeto observado, como uma rota, por exemplo; 2) Uma rodada de escolha de um objeto dentre o universo de todos os objetos, com base na probabilidade resultante do cálculo ocorrido no passo 1; 3) A definição do *fitness* de acordo com a informação definida como sendo a chave para seleção da melhor escolha. Durante cada ciclo, há a ação da evaporação do feromônio, cuja finalidade é atuar como agente diminuidor do valor cumulativo de cada visita à taxa de proporção estipulada pelo pesquisador.

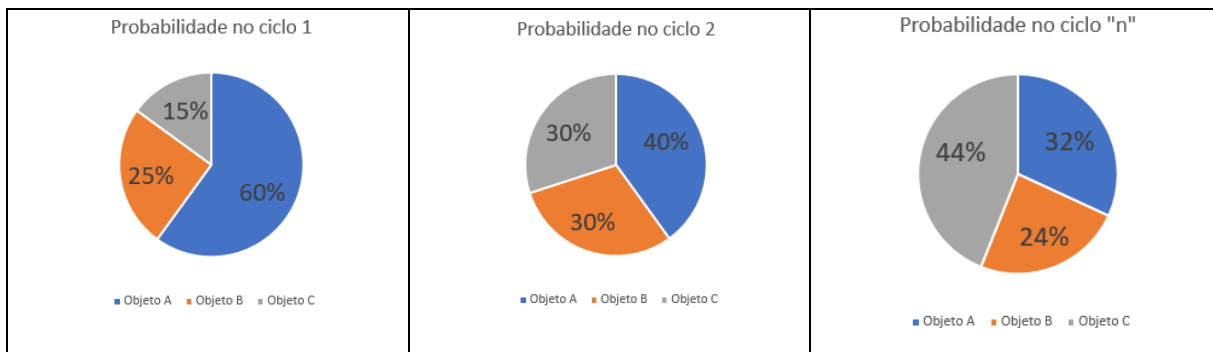
Esta tese propõe o estudo do ACO como mecanismo de escolha para transmissão de pacotes entre um ponto emissor e um receptor baseado nas informações de possíveis caminhos definidos pelo protocolo de roteamento *ShortestPath*, nativo do MATLAB. Na simulação, o mecanismo baseado em ACO se sobrepõe à detalhes de rota informadas pelo protocolo nativo, como o custo, e envia o pacote por rota distinta definida com base no resultado matemático resultante do cálculo dos requisitos previamente definidos que, para este caso, são o feromônio e a energia disponível nos caminhos disponíveis. A fórmula analítica básica do ACO (fase 1) se apresenta como segue:

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in j_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta}$$

A equação acima representa o feromônio deixado no caminho entre dois pontos por cada formiga durante sua passagem pelo que lhe foi indicado como opção de visita e η_{ij} indicam demais valores inerentes a este mesmo caminho, caso tenham influência na decisão, sendo oportuno informar que outros tantos valores podem ser inseridos na fórmula, desde que respeitada sua estrutura. Os valores correspondentes aos símbolos α e β são parâmetros de ajuste da equação que controlam o peso que seus atributos terão na tomada de decisão. O resultado desta equação informará qual a probabilidade que determinado objeto analisado terá de ser escolhido.

A fase 2 (escolha de um dos objetos observados dentre o universo de objetos) utiliza o mecanismo de roleta, que escolherá aleatoriamente qualquer dos objetos, mas o resultado dos cálculos de probabilidade obtidos na fase um não são ignorados, conforme ilustração criada para melhor entendimento, a seguir

Tabela 1 – Cálculo de probabilidade ACO



Assim, o valor estipulado pelo pesquisador será acrescido à soma cumulativa de cada visita feita por uma formiga aos objetos disponíveis cuja fórmula analítica é:

$$\Delta\tau_{i,j}^k = \begin{cases} 1 \\ L_k \\ 0 \end{cases}$$

Onde Δ representa a quantidade de feromônio que uma formiga deposita entre os pontos i e j , enquanto 0 é o valor computado se a formiga não visitar o caminho i,j . L_k é o peso, ou valor do feromônio a ser depositado, no caminho entre i, j , que pode ser inversamente proporcional em alguns casos.

Ao mesmo tempo, o resultado da fórmula acima sofrerá perda resultante do evento de evaporação, com fórmula:

$$\tau_{i,j}^k = (1 - \rho)\tau_{ij} + \sum_{k=1}^m \Delta\tau_{i,j}^k$$

Ante o exposto, a lógica adotada para este experimento levou em conta que o indicador de visitação da rota eleva sua probabilidade, o que resultaria na utilização elevada dos nós envolvidos na rota, incidindo em pane energética de alguns dos dispositivos com consequente perda de seus vértices com vizinhos e respectiva perda de possíveis rotas alternativas. Assim, o expoente α utilizado no processamento do feromônio foi definido como negativo, gerando efeito inverso, ou seja, quanto maior o feromônio, menor a chance daquela rota ser novamente escolhida.

Efetuada o cálculo, das probabilidades, os valores são informados ao sistema de escolha aleatória, conhecido como *Roulette Wheel*, cuja lógica é: dadas as possibilidades de 3 objetos, proceder com o sorteio de um número aleatório entre 0 e 1 que se encaixará dentro das possibilidades apresentadas, indicando a escolhida, conforme demonstra ilustração abaixo

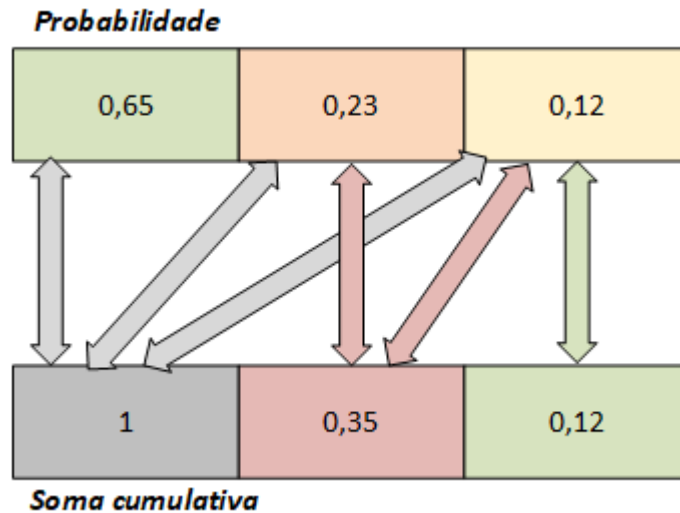


Figura 5 - Lógica de seleção roulette whell (autoria própria)

Após o sorteio do valor aleatório, para o caso ilustrado e número aleatório “x” sorteado 0,32768954, a regra de escolha é:

Tabela 2 - Exemplificação da seleção roulette whell

Escolher a rota...	Se “x”...
Rota 1	$1 > x > 0,35$
Rota 2	$0,35 > x > 0,12$
Rota 3	$x < 0,12$

Como etapa conclusiva, o *fitness* consiste em uma fórmula personalizada e definida especificamente pelo pesquisador para alcançar os fins pretendidos. Assim, por exemplo, se o objetivo é utilizar as informações de acumulado para a tomada de decisão, define-se matematicamente como sendo a melhor escolha, aquela rota com o menor valor, maior, média, mediana etc, pois o *fitness* não tem formula definida ou rígida e deve representar o cálculo que informe qual o melhor valor que atenda o projeto definido.

Ante o exposto, para simulação em pauta foi utilizado o valor 1 para feromônio a cada visita de formiga a uma rota e -0,1 para seu exponencial α . Quanto à energia, por serem valores por volta, e acima, de 800 (soma energética de todos nós de uma rota), adotou-se 0,1 como valor de sua exponencial β . O coeficiente de evaporação foi definido em 0,2 (20%) de evaporação a cada ciclo. As definições informadas fizeram com que as probabilidades para aquelas rotas que apresentassem baixa quantidade de feromônio, indicando baixo uso, e alta quantidade de energia, oriunda de consecutivas recargas energéticas sem o consumo proporcionado pela atividade de roteamento tivessem o melhor *fitness* para serem eleitas como as melhores rotas do momento, substituindo aquelas que vinham apresentando alto uso e consecutivo risco de pane energética.

4.3 Relevância da proposta

O estudo presente nesta dissertação tem por finalidade analisar a extensão na vida útil de uma rede de sensores sem fio, interesse originado de informações obtidas de diversos artigos científicos onde é informado que um dos maiores desafios da atualidade é evitar que uma rede de sensores sem fio se torne indisponível, ante o esgotamento energético ocasionado gradualmente a cada nó da rede pelo uso arbitrário e incondicional dos nós. Assim, a inclusão de um mecanismo inteligente e autônomo baseado em um modelo matemático flexível e personalizável, capaz de analisar fatores diversos e relevantes, é uma séria candidata a ser a solução para evitar a falência da rede de sensores sem fio. Os testes realizados pela simulação neste trabalho indicam que a aplicação deste mecanismo de escolha inteligente de rotas baseada em análise de contexto poderá perpetuar indefinidamente o tempo de vida de uma WSN. Os detalhes desta simulação e seus resultados serão descritos a seguir.

5. SIMULAÇÃO

A simulação foi executada utilizando o programa MATLAB (suíte R2019b) para criação de um ambiente WSN, com a possibilidade de variação na quantidade de nós e alcance de cada um destes (raio de alcance do sinal) para a criação de vizinhança. Neste sentido, analisa-se quantos pacotes a rede gerada consegue transmitir enquanto existirem nós “vivos” (com energia) capazes de gerar rotas e encaminhamento de pacotes entre uma torre emissora e uma torre receptora, utilizando-se oportunisticamente de dispositivos IoT disponíveis no caminho para que possam trocar dados de forma emergencial.

5.1 Cenário

Dispositivos com capacidade de interação estão disponíveis em todos os lugares. São dispositivos com alguma capacidade de processamento, armazenamento, conectividade wireless e energia. Atualmente, grande parte destes equipamentos são os celulares que, pela distribuição geográfica, proporcionam uma grande malha de conectividade por onde fluxos de dados podem trafegar de um ponto a outro, seja em um cenário previsto e esperado, seja em um de emergência, onde os dispositivos disponíveis na localidade podem servir de roteadores para transmissões de emergência entre dois pontos distintos de serviços de socorro.

Para o cenário definido para simulação, adotou-se que cada nó é equipado com algum dispositivo de recarga energética, como uma placa de energia solar e/ou está conectado em alguma fonte de energia, que lhe recupera energia em uma quantidade menor do que a gasta pela atividade de roteamento, quando o nó estiver envolvido em alguma rota emergencial em uso. As coordenadas de posicionamento de cada nó pode ser criada de forma aleatória para cada rodada dentro de uma área equivalente a 1Km de largura por 1Km de comprimento para as coordenadas de GPS -9.6605228 (Latitude), -35.7068941 (Longitude) e -9.662319(Latitude), -35.700399 (longitude), cuja figura ilustrativa de fundo, adequada às coordenadas, corresponde à cidade de Maceió, estado de Alagoas/Brasil, visualizando parte dos bairros da ponta verde e pajuçara.

Para a criação de vizinhança não foram levados em conta possíveis obstáculos entre nós dentro do raio de alcance que pudessem impedir a criação do enlace, cujo raio adotado para esta

simulação é de 50 metros. A energia de cada um dos nós também pode ser definida de forma sempre aleatória a cada rodada de simulação. A quantidade de nós, raio, energia máxima e mínima, fator de ganho e consumo energético de cada nó são personalizáveis. A simulação conta com um parâmetro de reprodutibilidade que permite sempre gerar os mesmos números aleatórios para que, para uso de mesmos valores de quantidade de nós, raio, energia máxima e mínima valores idênticos possam ser gerados para fins comparativos.

5.2 Metodologia

O processo de simulação consiste nos seguintes passos:

1. A posição dos nós é gerada aleatoriamente;
2. Cada nó inicia no cenário com uma quantidade entre 80 e 100 unidades de energia;
3. A distância euclidiana entre cada nó é calculada;
4. O enlace é estabelecido entre nós cuja distância euclidiana seja igual ou menor ao raio definido, o que indica que os equipamentos estão dentro do alcance um do outro;
5. O Protocolo de roteamento vetor-distancia embarcado no MATLAB, conhecido como *ShortestPath* (Menor caminho) busca rota com menor quantidade de saltos (nós) entre origem e destino, caso não exista rota, por coincidência do sorteio de posicionamento dos nós não fornecer caminhos entre origem e destino, a simulação para e notifica o pesquisador via mensagem no console (“*No route between source and target*”). Havendo, clona todos parâmetros do cenário montado para uso pelo ACO e segue para o item 6;
6. Identifica todos os nós envolvidos no roteamento e todos aqueles não envolvidos no roteamento;
7. Inicia-se uma rodada de consumo e recuperação energética para cada nó envolvido na rota indicada pelo mecanismo vetor-distância *ShortestPath*;
8. Inicia-se uma rodada apenas de ganho energético para cada nó do cenário não envolvido em rota para aquele momento. Não sendo computado gasto algum por adotar-se que nós não envolvidos apresentam consumo irrelevante por estarem em *standby*;

9. Para cada rodada de consumo para todos nós envolvidos na rota em uso, computa-se a adição de um pacote transportado com sucesso entre origem e destino;
10. Quando um nó envolvido tem sua energia esgotada, o cenário detecta a indisponibilidade da rota, reativa o shortestpath para busca de nova rota alternativa e, encontrando, volta ao passo 6, não encontrando mais rotas disponíveis devido ao esgotamento energético de todos nós possíveis, alerta o pesquisador e inicia a chamada da rotina de simulação do ACO;
11. A rodada de simulação ACO chama os dados clonados do primeiro cenário gerado (Item 5), garantindo que a inteligência de enxame seja aplicada sob o mesmo cenário e todos valores originalmente gerados;
12. O ACO inicia a rodada de análise de todas as rotas possíveis encontradas pelo shortestpath e deposita 1 unidade de feromônio para cada 1;
13. Com base na quantidade de feromônio e soma energética da rota, escolhe aquela com valor Fitness mais adequado (Stigmergia);
14. ACO deposita feromônio na rota selecionada;
15. Inicia rodada idêntica à descrita dos itens de 6 a 9;
16. Retorna à rotina descrita nos itens 13 e 14;
17. A simulação para caso um nó seja detectado como “morto” (sem energia).

A lógica utilizada pelo ACO é que, aquelas rotas com maior feromônio depositado, apontam para possível risco de exaustão energética de qualquer dos nós envolvidos, assim, migrando para outra rota que, provavelmente, contenha outros nós, dando possibilidade de que aqueles candidatos à exaustão possam recuperar suas energias.

5.3 Parâmetros

Abaixo, seguem os parâmetros utilizados para as rodadas de simulação com e sem o ACO, com respectivos comentários.

Parâmetros utilizados para a simulação sem ACO (Arquivo main.m):

1. dataset.nodeNo = 70; %Number of nodes

2. `ACOnodeNo = dataset.nodeNo;`
3. `dataset.nodePosition(1,:) = [1 50 50];` % (Sender node fixed position)
4. `dataset.nodePosition(2,:) = [2 900 900];` % (Receiver node fixed position)
5. `dataset.NeighborsNo = 5;`
6. `dataset.range = 250;` % Tolerance distance to become neighbor of one node (Euclidean distance based)
7. `dataset.atenuationFactor = 1.8;` % Atenuation factor in freespace - ranges from 1.8 to 4 due environment
8. `dataset.minEnergy = 80;` % Mw - Miliwatts (70% energy)
9. `dataset.maxEnergy = 100;` % Mw - Miliwatts (Full energy (100%) - 1 mAh charge capacity within 1 Volt energy)
10. `dataset.energyconsumptionperCicle = 0.75;`
11. `dataset.energyrecoveryperCicle = 0.2;`
12. `dataset.minenergyfactor = 0.18;`
13. `dataset.maxenergyfactor = 0.2;`
14. `STenergy=10000;`
15. `packet=0;`
16. `iterationcounter=1;`
17. `plotgraphs=1;` % Choose 1 for "yes" or 0 for "no" if you want to plot graphs or no (Better performance if no)
18. `reproduitibily = 0;` % 1 = yes (always generate same random numbers) (0) for no reproductibility (Different random numbers every code execution);

Comentários:

Linha 1) Quantidade de nós no cenário

Linha 2) Clone apenas do número de nós. Sem utilidade prática ou no código

Linhas 3 e 4) Definição de posições fixas para os nós 1 e 2 (remetente e destinatário)

Linha 5) Sem uso

Linha 6) Alcance do raio de cada nó, definido em metros

Linha 7) Fato de atenuação para cálculo do pathloss.

Linhas 8 e 9) Definição dos valores mínimo e máximo de unidade de energia a ser sorteada para cada nó

Linha 10) Valor do consumo energético de encaminhamento de um pacote, para cada nó envolvido em uma rota em uso.

Linha 11) Valor do ganho energético (recarga) de cada nó do cenário

Linhas 12 e 13) Pequeno índice randômico de variação de ganho e consumo energético aplicado a cada nó

Linha 14) Sem uso

Linha 15) Valor inicial de pacotes transmitidos sem ACO

Linha 16) Contador de iterações do bloco de loop principal sem ACO

Linha 17) Define se os gráficos deverão ser plotados, ou não. (A escolha da plotagem dos gráficos diminui cerca de 40% a velocidade de processamento.)

Linha 18) Fator de reprodutibilidade para que sempre sejam gerados mesmos valores aleatórios, ou diferentes.

Parâmetros utilizados para a simulação com ACO (Arquivo aco.m):

1. ACOsourcnode = 1;
2. ACOtargetnode = 2;
3. ACOmaxAntiteration = 100;
4. ACOantsNo = 21;
5. ACOpheromone = 1; %tau - pheromone concentration - most papers use 0 as its value
6. % ACOpathloss = 0.2; %eta - pathloss of an edge for aco calc
7. % ACOedgeenergy = 0.5; %Energy weight for aco calc
8. ACOevapcoefficient = 0.2; % 5 percent - (rho) pheromone evaporation rate
9. ACOalpha = -0.1; %Pheromone exponent
10. ACObeta = 0.1; %Energy exponent
11. ACOomega = 1; %Pathloss expoent
12. ACOpacket=0;
13. ACOiterationcounter=1;
14. ACObestfitness=inf;
15. ACOenergycycles=100;
16. ACObesttourhistory=[];

Comentários:

Linhas 1 e 2) Definição da identificação dos IDs do emissor e receptor

Linha 3) Número máximo de iterações das formigas, indicando o número de passeios que cada uma fará

Linha 4) Número de formigas que serão utilizadas

Linha 5) Valor que será depositado em cada item visitado por uma formiga

Linhas 6 e 7) Sem uso

Linha 8) Coeficiente de evaporação. Este valor indica que, a cada novo ciclo, a soma de todo feromônio depositado em um item irá desaparecer à taxa definida

Linha 9) Valor do expoente de relevância do feromônio. Neste caso é um valor negativo porque, a quantidade de feromônio depositado é indiretamente proporcional à sua relevância na próxima escolha, pois quanto mais feromônio, mais uso, mais risco de exaustão energética, menos relevância a rota deve ter na próxima escolha.

Linha 10) Valor do expoente de relevância da energia. Quanto mais energia a rota tiver, maior deve ser sua relevância, pois está mais longe de um estado de energia crítico.

Linha 11) Sem uso

Linha 12) Valor inicial de pacotes transmitidos com ACO

Linha 13) Contador de iterações do bloco de loop principal com ACO

Linha 14) Índice de busca do melhor fitness. Não há valor fixamente definidos porque os valores não param de modificar

Linha 15) Sem uso

Linha 16) parâmetro utilizado para armazenar o histórico de mudanças de rotas

Para o desenvolvimento desta pesquisa foram utilizadas simulações com uso do programa MatLab/SimuLink (suíte R2019b), por ser ferramenta respeitada e amplamente utilizada pela comunidade científica, com os seguintes plugins adicionais:

- Antenna toolbox: Disponibiliza design e análise de diversos tipos de antenas e seus respectivos padrões de funcionamento.
- Communications toolbox: Algoritmos para análise e modelagem de comunicações fim-a-fim.
- Database toolbox: Disponibiliza importação e exportação de dados com outros bancos de dados.

- MATLAB report generator: Gerador de relatórios do Matlab integrado em tempo real com as principais ferramentas de texto do mercado
- Phased Array System Toolbox: Algoritmos e aplicativos para design, simulação e análise em comunicações wireless
- RF blockset: Disponibiliza biblioteca para simulação de redes de rádio frequência com captura de ganhos, perdas, ruídos e outros tipos de distorções.
- RF toolbox: Prove funções e objetos para modelagem e análise de intensidade de rádio frequência
- Robotics System Toolbox: Simulação de movimento om verificação de colisão e geração de trajetória.
- Sensor Fusion and Tracking Toolbox: Inclui algoritmos e ferramentas para simulação para ambientes que necessitam de dados de posicionamento, orientação e gatilhos de eventos.
- Signal processing toolbox
- SimEvents: Disponibiliza mecanismo de simulação e biblioteca de componentes apara analisar eventos dirigidos e otimizar latência, throughput, perda de pacotes, QoS e análise de roteamento
- Simulink code generator: Compatibilidade com linguagem C e C++
- Simulink Control Design: Controla e monitora processamento
- Simulink Design Optimization: Definição de sensibilidade do ambiente e dos objetos para resultados mais confiáveis e reais.
- Simulink desktop realtime: Prove acesso mais rápido ao processador do computador de teste
- Simulink report generator: Criação de diagramas de blocos, gráficos, dicionário de dados, códigos, documentação e resultados de testes.
- Wavelet Toolbox: Análise de espectro da rádio frequência ao longo do tempo.

O computador utilizado no experimento foi um Dell inspiron 14 7460, I7 G7 com 4MB de cache e 2.7~3.5Ghz, 16 GB de memória ram 2.400Mhz e dois discos de estado solido de 500Gbs de velocidade de gravação.

6. RESULTADOS

A simulação em questão contemplou 70 nós aleatoriamente distribuídos em uma área de 1Km x 1Km, considerando todos os nós com alcance de 250 metros para estabelecimento de enlace com nó dentro do raio, gerando o seguinte gráfico:

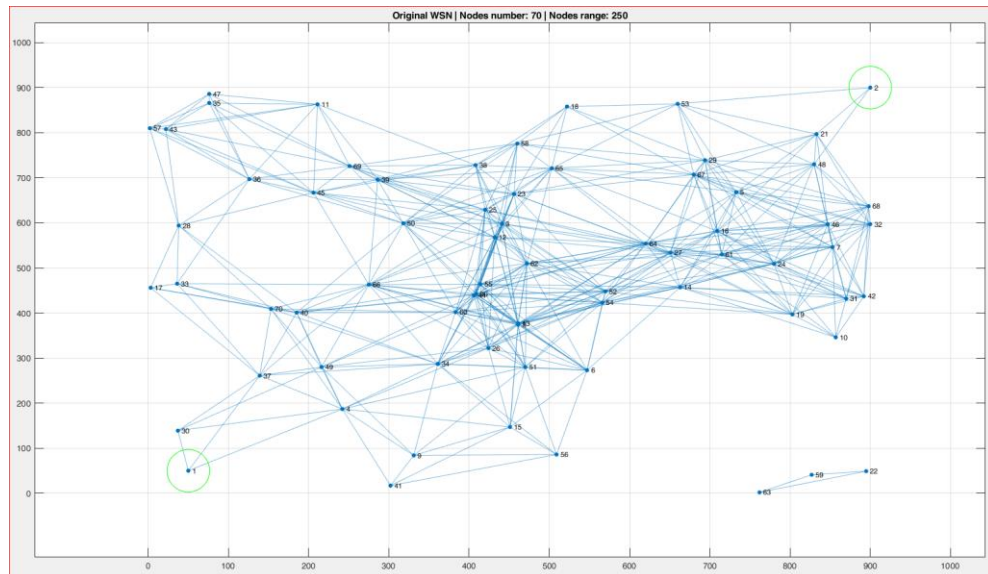


Figura 6 - Primeiro estado da rede para simulação sem ACO (autoria própria)

A ilustração acima apresenta o cenário aleatoriamente montado para a simulação, onde o ponto 1, presente na parte inferior esquerda, necessita transmitir dados para o ponto 2, parte superior direita. A transmissão é oportunística, ou seja, as torres precisam trocar dados em caráter emergencial.

Tabela 3 - Log de eventos da rede de 70 nós sem ACO

No ACO Scene	Hops	Packets sent	Dead node
2	7	5189	58
3	7	5304	26
4	7	6104	4
5	7	11481	21
6	7	12034	16
7	7	17829	48
8	7	24157	65
9	8	30450	18
10	8	30607	23

11	8	30861	30
12	8	37015	14
13	8	37169	5
14	8	43176	29
15	8	43441	27
16	8	49149	67
17	Sem Rotas entre Origem e Destino		

O protocolo de roteamento encontra a melhor rota e inicia a transmissão, sendo possível observar no log abaixo apresentado os detalhes dos eventos ocorridos na rede em questão do momento em que iniciam as transmissões até o colapso total das transmissões por inexistência de caminhos, ocasionado pelo consumo excessivo e arbitrário de energia.

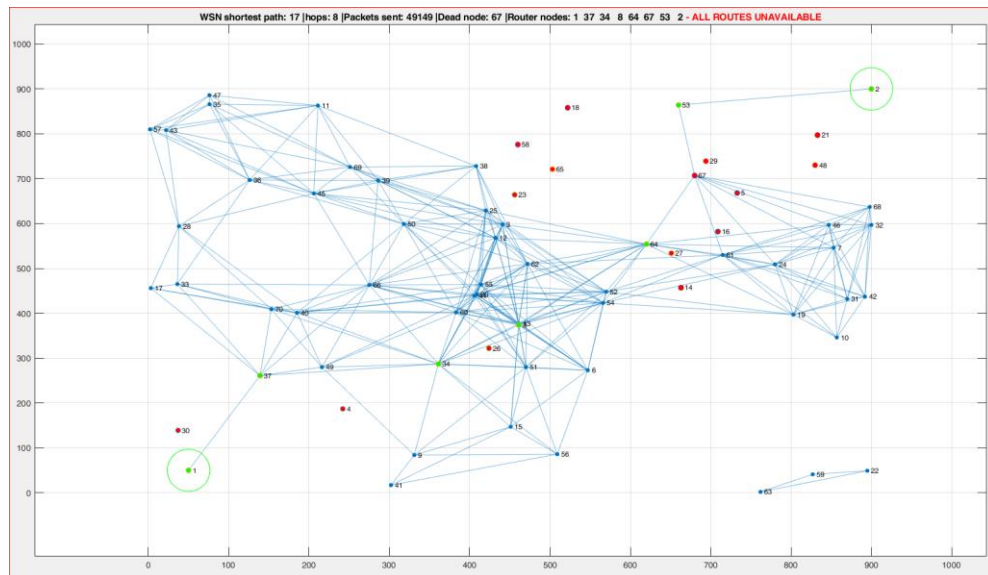


Figura 7 - 17º e último estado da rede para simulação sem ACO (autoria própria)

Com a indisponibilidade de comunicação pela extinção de todas as possíveis rotas, resultante da perda de 225 conexões das 948 existentes, conforme comprovado pela foto de estado acima e representado pelo gráfico abaixo.

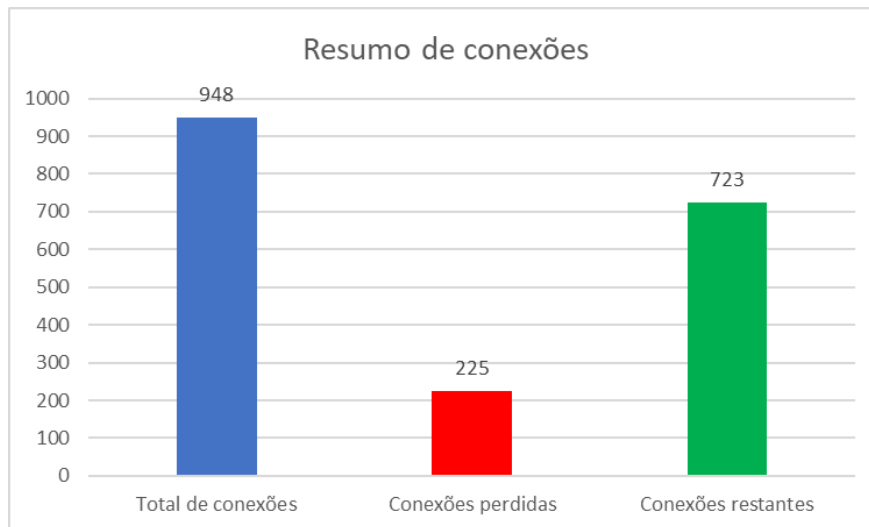


Figura 8 - Resumo de conexões da rede (autoria própria)

Neste ponto, inicia-se a simulação de transmissão para esta mesma rede, com parâmetros idênticos à gerada inicialmente (“Figura 5 - Primeiro estado da rede para simulação sem ACO”), conforme apresenta ilustração comprobatória que segue.

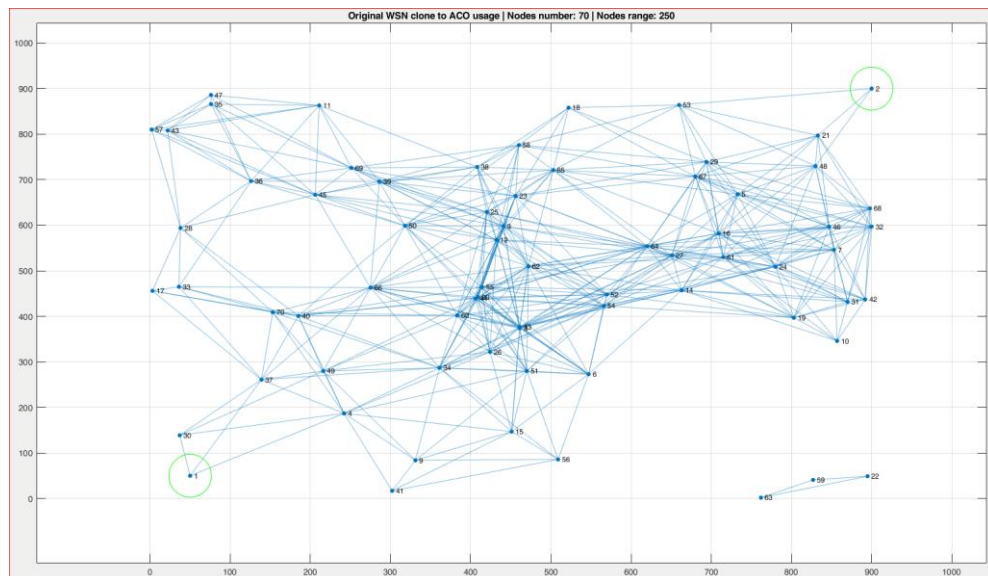


Figura 9 - Primeiro estado da rede para simulação com ACO (autoria própria)

Nesta etapa com ACO, o cenário é o mesmo: As rotas são descobertas e o ACO inicia sua primeira verificação para aquisição do conhecimento necessário para geração de seus ciclos decisórios. Com objetivo de abreviar a simulação, cada ciclo ACO ocorre a cada 100 iterações

de transmissão, o que, obviamente, compromete sua eficiência. Mesmo com o mecanismo ACO penalizado, o primeiro evento de nó indisponível por pane energética ocorre quando a rede já havia transmitido 38.201 pacotes contra 49.149 totais da mesma rede sem aplicação de ACO.

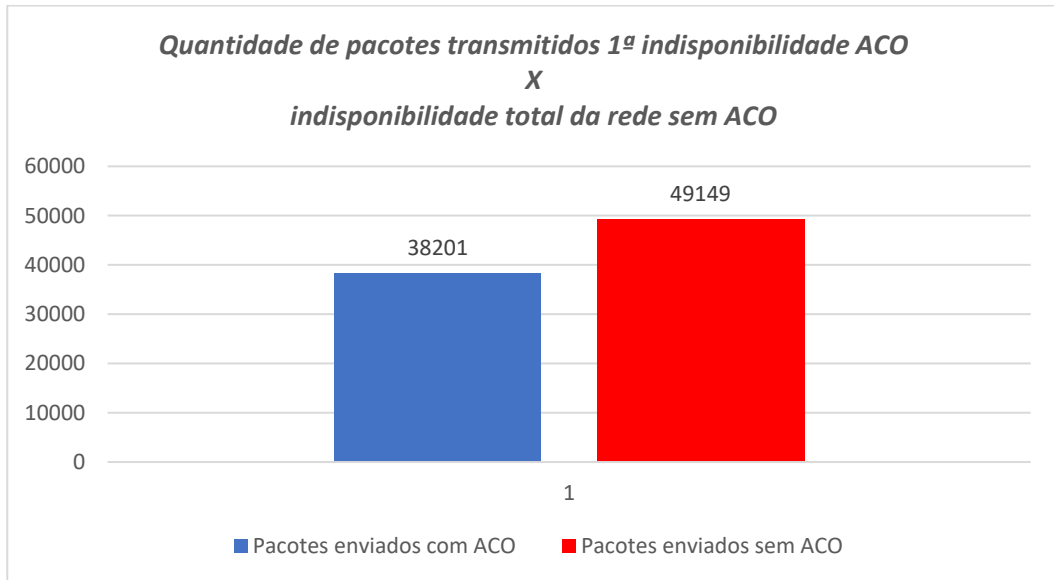


Figura 10 - Transmissões totais sem ACO X primeira indisponibilidade ACO (autoria própria)

Na simulação com ACO, pela imprevisibilidade da duração que a simulação poderia demorar, ela para a execução assim que o primeiro nó tem esgotamento total de energia, cujos totais comparativos estão disponíveis no apêndice C.

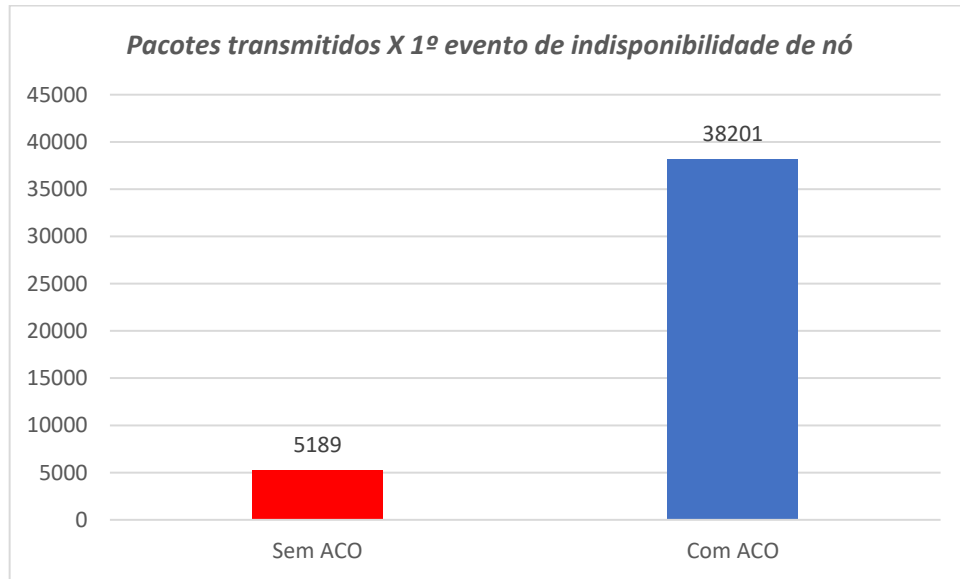


Figura 11 - Transmissões antes da primeira indisponibilidade sem e com ACO (autoria própria)

O gráfico acima demonstra o total de pacotes transmitidos antes do primeiro evento de indisponibilidade de nó da rede.

6.1 Discussão dos Resultados

Do estudo do estado da arte inferiu-se a necessidade de mecanismos de decisão que propiciem à rede maior durabilidade energética. No cenário da simulação utilizada neste estudo, foi adotado que todos os nós são equipados com pequeno dispositivo de recarga como luz solar e/ou estão, em algum momento, conectados à fonte de energia com potência de recarga sempre menor que o consumo referente à tarefa de roteamento, quando em execução. A definição deste cenário ofereceu à simulação proximidade com a realidade necessária para gerar dados confiáveis e coerentes.

Observando os dados obtidos de todas as simulações, a tabela 1 apresentada revela que a rede atinge o fim de sua vida útil de forma rápida, evento resultante do mecanismo arbitrário de escolha da rota pela não existência de engenho capaz de analisar todo contexto, assim perdendo a capacidade de prolongar a vida da rede por não ter dados referentes ao conjunto, apresentando perda rápida e constante de nós e relações de vizinhança e prematura

chegada ao fim de sua utilidade pelo esgotamento de todos os possíveis vértices de conexão por onde os dados pudessem ser transmitidos.

O gráfico comparativo da quantidade de pacotes transmitidos antes da pane energética do primeiro nó da rede ACO frente a quantidade total de pacotes transmitidos durante a vida útil da rede legada deixam evidente a eficiência proporcionada pelo ACO ao ambiente, uma vez que ACO garantiu o envio de 38.201 pacotes antes da perda de seu primeiro nó, enquanto que a rede de tecnologia de roteamento legada transmitiu apenas 5.189 antes de seu primeiro evento de perda e 49.149 totais, ou seja, ACO transmitiu 736% mais pacotes antes de sua primeira perda e 77,72% do total de pacotes transmitidos em todo tempo de vida útil da rede legada, perdendo apenas um nó neste intervalo, sendo oportuno lembrar que, conforme informado anteriormente, a rotina de estigmergia do ACO foi penalizado em mais de 1000% para efeitos de velocidade de processamento, certamente não perdendo nenhum nó caso esta restrição não lhe tivesse sido imposta.

6.2 Ameaças à validade

Nesta seção são apresentados possíveis fatores que podem vir a influenciar os resultados obtidos neste trabalho, divididos em ameaças internas e externas. A principal ameaça interna é a codificação da ferramenta e a ameaça externa compreende os projetos analisados a partir dos critérios de seleção.

6.3 Ameaças internas

A ferramenta utilizada para este experimento pode apresentar resultados não tão realísticos. Isto se deve ao fato da presença da plataforma de sistema operacional empregada, que funciona como intermediária entre o MatLab e o hardware, bem como demais aplicações e serviços executados em segundo plano, que concorrem para utilização da CPU. Este tipo de comportamento é esperado e deve ser levado em consideração. Para diminuir a intensidade da interferência citada os plugins Simulink Design Optimization e Simulink desktop realtime foram utilizados.

6.4 Ameaças externas

A principal ameaça externa está no ambiente natural ao qual, porventura, uma solução IoT para cidades inteligentes seja aplicada. Acredita-se que um ambiente real apresente mais agentes impactantes à comunicação entre os dispositivos do que aquele que é criado em ambiente de simulação, como barreiras físicas de maior ou menor influência, como a presença de prédios com paredes muito espessas, a presença de significativa quantidade de material ferroso, mecanismos eletromagnéticos e mecânicos dos mais variados e a presença de sinais de radiofrequência transmitindo na mesma faixa dos dispositivos empregados. Assim, devido à impossibilidade de criar cenário com valores médios relativos ao cenário real de uma cidade, este trabalho abstrai quaisquer possíveis fatores modificantes externos, utilizando um cenário livre de ruídos.

7. CONSIDERAÇÕES FINAIS

Durante o levantamento do estado da arte referente à aplicação de IoT, foi detectado que existe um problema relacionado ao tempo de vida de uma rede IoT, devido a sua característica de recursos limitados e ineficiência de mecanismos de gerência e ausência de métodos autônomicos com visão heurística da rede e, como é tendência da humanidade convergir para a utilização de dispositivos móveis interconectados com capacidade de prover grande gama de serviços como sensoriamento e fluxo de dados diversos através de WNS's é inquestionável. Em cenários de emergência, como a indisponibilidade dos serviços primários de comunicação devido a catástrofes como terremotos, ataques militares, erupções vulcânicas e outras, a rede IoT ocasionalmente presente no ambiente pode ser utilizada, de forma oportunística, pelos serviços governamentais para o transporte de dados entre suas bases de controle de operações, desde que confiáveis. Assim, frentes de pesquisa sobre a manutenção do tempo de vida das redes de sensores se mostram preocupadas com manutenção do tempo de vida útil da rede, que se mostra comprometida devido ao exaurimento energético dos nós, ocasionado pelo excesso de uso ineficiente patrocinado pela inflexibilidade na decisão do uso de rotas e a inexistência de um mecanismo capaz de criar uma visão global do contexto de uso, que seja capaz de desviar o fluxo dos dados por caminho que apresente melhores condições energéticas de conduzir a comunicação, e este foi fator motivante para o desenvolvimento deste trabalho.

O estudo presente nesta tese propôs a aplicação de inteligência baseada em técnicas de enxame de formigas e os testes apontaram que a utilização do ACO surtiu os efeitos esperados, quando, para redes idênticas em todos os sentidos, iniciou-se transmissão oportunística entre uma torre emissora e outra receptora através da rede IoT disponível sem e com a utilização do ACO. Os dados coletados apontam a completa desintegração dos enlaces resultando em impossibilidade de interligar as torres, enquanto a aplicação de ACO manteve a rede operacional, sem a perda de caminho ocasionada pela perda de nó participante por tempo muito superior e com mais pacotes transmitidos por evento de perda de nó em diversas simulações. Este estudo conclui que a aplicação do ACO relevante eficiência na gestão do uso de rotas em uma WSN com base na análise energética dos caminhos, frente ao algoritmo de roteamento *ShortestPath*, prolongando o tempo de vida da rede e transmitindo quantidade relevantemente maior de pacotes antes do colapso da rede.

Como trabalhos futuros, diversas frentes se abrem, visto que a aplicabilidade de técnicas de enxame se apresentou como solução para o cenário de IoT pela inquestionável melhora que apresentou em todas as simulações em relação ao protocolo de roteamento utilizado para comparação. Ante o exposto, a análise deste mesmo ambiente utilizando técnicas de enxame diferentes da *Ant colony Optimization*, como *Bee colony optimization*, *Grey Wolf Optimization*, *Firefly Optimization*, dentre outros trará uma visão mais ampla sobre a eficácia de todos estes métodos. Outra frente de pesquisa seria o estudo de valores de parâmetros melhores aplicados a mesma rede para análise de qual é o valor ideal para atingir o máximo de performance do ACO, bem como a adoção de novas variáveis ao cálculo, como pathloss no núcleo de cálculo do ACO para análise comparativa dos resultados e uma análise de relação referente à quantidade de vértices de vizinhança e a influência que estes têm no computo geral de performance da técnica de enxame aplicada.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] A. Castro Lundin, A. G. Ozkil, and J. Schuldt-Jensen, “**Smart Cities: A Case Study in Waste Monitoring and Management**,” *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, pp. 1392–1401, 2017, doi: 10.24251/hicss.2017.167.
- [2] D. Feldner, *Redesigning Organizations*. IEEE Networks, 2018 .
- [3] S. Abbasian Dehkordi, K. Farajzadeh, J. Rezazadeh, R. Farahbakhsh, K. Sandrasegaran, and M. Abbasian Dehkordi, “**A survey on data aggregation techniques in IoT sensor networks**,” *Wirel. Networks*, vol. 0123456789, 2019, doi: 10.1007/s11276-019-02142-z.
- [4] F. Gu, J. Niu, L. Jiang, X. Liu, and M. Atiquzzaman, “**Survey of the low power wide area network technologies**,” *J. Netw. Comput. Appl.*, vol. 149, p. 102459, 2020, doi: 10.1016/j.jnca.2019.102459.
- [5] L. Farhan, O. Kaiwartya, L. Alzubaidi, W. Gheth, E. Dimla, and R. Kharel, “**Toward Interference Aware IoT Framework: Energy and Geo-Location-Based-Modeling**,” *IEEE Access*, vol. 7, pp. 56617–56630, 2019, doi: 10.1109/ACCESS.2019.2913899.
- [6] S. Y. Hashemi and F. Shams Aliee, “**Dynamic and comprehensive trust model for IoT and its integration into RPL**,” *J. Supercomput.*, vol. 75, no. 7, pp. 3555–3584, 2019, doi: 10.1007/s11227-018-2700-3.
- [7] A. Dhumane, S. Guja, S. Deo, and R. Prasad, “**Context Awareness in IoT Routing**,” *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, pp. 1–5, 2019, doi: 10.1109/ICCUBEA.2018.8697685.
- [8] G. Shabbir, A. Akram, M. M. Iqbal, S. Jabbar, M. Alfawair, and J. Chaudhry, “**Network Performance Enhancement of Multi-sink Enabled Low Power Lossy Networks in SDN Based Internet of Things**,” *Int. J. Parallel Program.*, 2018, doi: 10.1007/s10766-018-0620-8.
- [9] C. Kapoor, “**Routing Table Management using Dynamic Information with Routing Around Connectivity Holes (RACH) for IoT Networks**,” *2019 Int. Conf. Autom. Comput. Technol. Manag.*, pp. 174–177, 2019, doi: 10.1109/icactm.2019.8776676.
- [10] J. N. Al-Karaki and a E. Kamal, “**Wireless Sensor Networks Routing Techniques in Wireless Sensor Networks : a Survey**,” *Ieee Wirel. Commun.*, vol. 11, no. December, pp. 6–28, 2004, doi: 10.1109/MWC.2004.1368893.
- [11] S. Khara, A. Saxena, S. K. Sinha, and S. K. Gupta, “**A Survey on IOT Issues and Data Analysis Techniques to Increase Life of Wireless Network**,” *Proc. 3rd Int. Conf. Commun. Electron. Syst. ICCES 2018*, no. Icces, pp. 176–182, 2018, doi: 10.1109/CESYS.2018.8724083.
- [12] F. Kiani, “**A survey on management frameworks and open challenges in IoT**,” *Wirel. Commun. Mob. Comput.*, vol. 2018, 2018, doi: 10.1155/2018/9857026.

- [13] H. Base and M. Will, *Demystifying Smart Cities* :, vol. 7, no. 7. 2017.
- [14] C. S. Ryu, “**IoT-based intelligent for fire emergency response systems,**” *Int. J. Smart Home*, vol. 9, no. 3, pp. 161–168, 2015, doi: 10.14257/ijsh.2015.9.3.15.
- [15] H. Arasteh *et al.*, “**IoT-based smart cities: A survey,**” *EEEIC 2016 - Int. Conf. Environ. Electr. Eng.*, pp. 2–7, 2016, doi: 10.1109/EEEIC.2016.7555867.
- [16] F. Al-Turjman, M. Abujubbeh, A. Malekloo, and L. Mostarda, “**UAVs assessment in software-defined IoT networks: An overview,**” *Comput. Commun.*, 2019, doi: 10.1016/j.comcom.2019.12.004.
- [17] L. Peng, A. R. Dhaini, and P. H. Ho, “**Toward integrated Cloud–Fog networks for efficient IoT provisioning: Key challenges and solutions,**” *Futur. Gener. Comput. Syst.*, vol. 88, no. 2018, pp. 606–613, 2018, doi: 10.1016/j.future.2018.05.015.
- [18] N. Mahakalkar and R. Pethe, “**Review of Routing Protocol in a Wireless Sensor Network for an IOT Application,**” *Proc. 3rd Int. Conf. Commun. Electron. Syst. ICCES 2018*, no. Icces, pp. 21–25, 2018, doi: 10.1109/CESYS.2018.8723935.
- [19] I. Gkotsis, A.-C. Kousouraki, G. Eftychidis, P. Kolios, and M. Terzi, “**Swarm of UAVs as an emergency response technology,**” in *Risk Analysis Based on Data and Crisis Response Beyond Knowledge: Proceedings of the 7th International Conference on Risk Analysis and Crisis Response (RACR 2019), October 15-19, 2019, Athens, Greece*, 2019, p. 353.
- [20] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rabêlo, J. Al-Muhtadi, and V. Korotaev, “**Routing protocols for low power and lossy networks in internet of things applications,**” *Sensors (Switzerland)*, vol. 19, no. 9, pp. 1–40, 2019, doi: 10.3390/s19092144.
- [21] I. Zaatouri, F. Sailhan, S. Rovedakis, A. Guiloufi, N. Alyaoui, and A. Kachouri, “**Performance Evaluation of Mobility-Aware Routing Protocol for Low power and Lossy Networks,**” *2019 16th Int. Multi-Conference Syst. Signals Devices*, pp. 636–641, 2019, doi: 10.1109/ssd.2019.8893187.
- [22] D. Ben Arbia, M. M. Alam, A. Kadri, E. Ben Hamida, and R. Attia, “**Enhanced IoT-based end-to-end emergency and disaster relief system,**” *J. Sens. Actuator Networks*, vol. 6, no. 3, pp. 1–18, 2017, doi: 10.3390/jsan6030019.
- [23] H. Aksu, L. Babun, M. Conti, G. Tolomei, and A. S. Uluagac, “**Advertising in the IoT Era: Vision and Challenges,**” *IEEE Commun. Mag.*, vol. 56, no. 11, pp. 138–144, 2018, doi: 10.1109/MCOM.2017.1700871.
- [24] H. Agarwal, F. Husain, and P. Saini, *Advances in Computing and Data Sciences*, vol. 1046. 2019.
- [25] G. R. Tsouri, A. Prieto, and N. Argade, “**On increasing network lifetime in body area networks using global routing with energy consumption balancing,**” *Sensors (Switzerland)*, vol. 12, no. 10, pp. 13088–13108, 2012, doi: 10.3390/s121013088.
- [26] C. F. G. Silva, *Configurando switches e roteadores CISCO*, 1st ed. São Paulo:

Brasport, 2013.

- [27] D. Prashar and D. Kumar, “**Performance Evaluation of the Optimized Error Correction Based Hop Localization Approach in a Wireless Sensor,**” *Wirel. Pers. Commun.*, no. 0123456789, 2019, doi: 10.1007/s11277-019-07001-w.
- [28] D. Science and S. G. Germany, “**Collaborative Defense with Multiple USVs and UAVs Based on,**” *J. Shanghai Jiaotong Univ.*, 2019, doi: 10.1007/s12204-019-2142-y.
- [29] L. Dupak and S. Banerjee, “**Role of bio-inspired algorithms for designing protocols in MANET-review,**” *Proc. - Int. Carnahan Conf. Secur. Technol.*, vol. 2019-October, 2019, doi: 10.1109/CCST.2019.8888407.
- [30] A. Prasanth, J. A. George, and P. Surendram, “**Optimal Resource and Task Scheduling for IoT,**” in *2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, 2019, pp. 1–4.
- [31] Q. Wang, W. Liu, H. Yu, S. Zheng, S. Gao, and F. Granelli, “**CPAC: Energy-efficient algorithm for IoT sensor networks based on enhanced hybrid intelligent swarm,**” *C. - Comput. Model. Eng. Sci.*, vol. 121, no. 1, pp. 83–103, 2019, doi: 10.32604/cmes.2019.06897.
- [32] P. Swarms, “**Chapter 6 SWARM INTELLIGENCE,**” *Springer*, vol. 6, 2004, doi: 10.1007_0-387-27705-6_6.
- [33] C. W. Schuety and L. E. Will, “**The American Way of Swarm: A Machine Learning Strategy for Training Autonomous Systems,**” *Def. Tech. Inf. Center/USA*, no. December, 2018.
- [34] M. Rathee, S. Kumar, A. H. Gandomi, S. Member, and K. Dilip, “**Ant Colony Optimization Based Quality of Service Aware Energy Balancing Secure Routing Algorithm for Wireless Sensor Networks,**” *IEEE Trans. Eng. Manag.*, vol. PP, pp. 1–13, 2019, doi: 10.1109/TEM.2019.2953889.
- [35] A. Yaghoubi and F. Akrami, “**Heliyon Proposing a new model for location - routing problem of perishable raw material suppliers with using meta-heuristic algorithms,**” *Heliyon*, vol. 5, no. June, p. e03020, 2019, doi: 10.1016/j.heliyon.2019.e03020.

GLOSÁRIO

4G – Termo utilizado para se referir à quarta geração da tecnologia de transmissão de dados em redes sem fio compostas por dispositivos móveis

5G – Melhoramento da tecnologia 4G

Ad-Hoc – Termo latim que significa *para isto*, para *determinado fim*

Estigmergia – Método de comunicação indireta e auto organizado onde diversos agentes colaboram entre si para criação de um ótimo global.

IP – Endereço, ou identificador, único e exclusivo utilizado para representar um dispositivo dentro de uma rede

IoT - Sigla utilizada para fazer referência a ambientes compostos por dispositivos tecnológicos capazes de trocar informações entre si. Na literatura também pode ser referenciado como IoE (Internet of Everything – Internet de tudo)

Jitter – Variação do valor da latência

Latência – Tempo que um pacote transmitido demora para percorrer o caminho entre o dispositivo emissor e receptor

LLN - Utilizada para se referir a ambientes compostos por dispositivos com recursos de hardware escassos como: energia, poder de processamento, memória, armazenamento, potência de antena e possibilidade de transmissão de dados

Link ou enlace – Termo utilizado para referenciar o estabelecimento de conexão entre dois, ou mais, dispositivos

MANET (Mobile Ad-hoc Network – Redes móveis) – Termo misto de inglês com latim para fazer referência a redes compostas de dispositivos móveis com características e/ou finalidades específicas.

Omni-direcional – Característica de antena que espalha/transmite seu sinal 360°

Pacote – Termo utilizado para referenciar a unidade mínima portadora de dados que são transmitidos entre dispositivos

PDR – Termo utilizado para se referir à quantidade de pacotes entregues entre origem e destino em um dado intervalo de tempo.

PLR – Taxa de perda de pacotes) - Termo utilizado para se referir à quantidade de pacotes perdidos entre origem e destino em um dado intervalo de tempo.

Protocolo – É uma padronização ou estrutura que define como os dados serão montados para que sejam compreendidos por todo e qualquer equipamento. Existem diversos protocolos para diversas finalidades, cada um com sua estrutura específica.

QoS - Refere-se à mecanismos de controle que priorizam a transmissão de dados classificados como mais importantes ou sensíveis que outros.

Range – Palavra do idioma inglês que significa alcance

Ruído – Termo técnico que se refere à interferências externas que podem influenciar negativamente no sinal de conexão entre dispositivos.

Smart cities - Termo do idioma inglês que significa cidades espertas (cidades inteligentes) utilizado para fazer referência a centros urbanos que utilizam tecnologia para prover, gerenciar e melhorar diversos serviços e a qualidade de vida

Throughput - Palavra do idioma inglês que significa taxa de transferência, geralmente medida em bits por segundo.

WSN – Ambiente composto por dispositivos equipados com sensores e que se comunicam via sinais de ondas de rádio, micro-ondas, infravermelho ou outra tecnologia similar.

Apêndice A

SIMULAÇÃO EM MATLAB

Arquivo main.m

```
clear all
close all
clc
```

```
%% GPU config
% gpu = gpuDevice;
% gpu(1);
```

```
tic
%% Main configuration values for this simulation
```

```
dataset.nodeNo = 70; %Number of nodes
ACOnodeNo = dataset.nodeNo;
dataset.nodePosition(1,:) = [1 50 50]; %(Sender node fixed position)
dataset.nodePosition(2,:) = [2 900 900]; %(Receiver node fixed position)
dataset.NeighborsNo = 5;
dataset.range = 250; %Tolerance distance to became neighbor of one node (Euclidean
distance based)
dataset.atenuationFactor = 1.8; %Atenuation factor in freespace - ranges from 1.8 to 4 due
environment
dataset.minEnergy = 80; % Mw - Miliwatts (70% energy)
dataset.maxEnergy = 100; % Mw - Miliwatts (Full energy (100%) - 1 mAh charge capacity
within 1 Volt energy)
dataset.energyconsumptionperCicle = 0.75;
dataset.energyrecoveryperCicle = 0.2;
dataset.minenergyfactor = 0.18;
dataset.maxenergyfactor = 0.2;
STenergy=10000;
packet=0;
iterationcounter=1;
plotgraphs=1; %Choose 1 for "yes" or 0 for "no" if you want to plot graphs or no (Better
performance if no)
reproduitibily = 0; % 1 = yes (always generate same random numbers) (0) for no reproductibility
(Different random numbers every code execution);

% Node position sortition
if reproductibily == 0
    rng('shuffle');
else
```

```

    rng('default');
end
for a = 3 : dataset.nodeNo

    dataset.nodeId = a;
    garbage.x = randi([1 900]); % Xpos sortition
    garbage.y = randi([1 900]); % Ypos sortition
    dataset.nodePosition(a,:) = [dataset.nodeId garbage.x garbage.y]; %NodeID, X and Y
    position into nodePosition table

end

% Euclidean Distance calc from one node to all others

for i = 1 : dataset.nodeNo
    for j = 1: dataset.nodeNo
        garbage.x1 = dataset.nodePosition(i,2);
        garbage.x2 = dataset.nodePosition(j,2);
        garbage.y1 = dataset.nodePosition(i,3);
        garbage.y2 = dataset.nodePosition(j,3);

        dataset.euclidiana(i,j) = sqrt( (garbage.x1 - garbage.x2) ^2 + (garbage.y1 -
garbage.y2)^2 );

    end
end

% Edges matrix definition due "range" variable value

dataset.weights = lt(dataset.euclidiana,dataset.range);

% Graph construction

G=graph(dataset.weights,'omitselfloops'); %Graph creation based on adjacency matrix (Edges
matrix) built above

% Euclidean distance extraction for all existente end-to-end formed by
% "distance tolerance" (range variable value)

for a = 1 : height(G.Edges)
    garbage.s = G.Edges.EndNodes(a,1);
    garbage.t = G.Edges.EndNodes(a,2);
    garbage.Z(a,:) = dataset.euclidiana(garbage.s,garbage.t);
end
G.Edges.Euclidiana = garbage.Z(:,1);

```

```

%Initial energy sortition (from 70% to 100% - minEnergy and maxEnergy variable valeu)

[dataset.nodePosition(:,4)] = dataset.maxEnergy -(dataset.maxEnergy-
dataset.minEnergy)*rand(dataset.nodeNo,1);
dataset.nodePosition(1:2,4)=STenergy;

%All "G" (Graph object) based nodes degree to use as "node processing
%status overload" (more connections, busier!)

for a = 1: length(dataset.nodePosition(:,1))

    dataset.nodePosition(a,5) = degree(G,dataset.nodePosition(a,1));

end

% Pathloss calc of each Edges based in a freespace (1.8 factor)

[G.Edges.Pathloss] = (10*dataset.attenuationFactor)*log10(G.Edges.Euclidian);

%End points coordinates and extra data migration to G object

for a = 1 : height(G.Edges)
    garbage.Sourcenode = G.Edges.EndNodes(a,1);
    garbage.Targetnode = G.Edges.EndNodes(a,2);
    G.Edges.SourcenodeXpos(a) = dataset.nodePosition(garbage.Sourcenode,2);
    G.Edges.SourcenodeYpos(a) = dataset.nodePosition(garbage.Sourcenode,3);
    G.Edges.TargetnodeXpos(a) = dataset.nodePosition(garbage.Targetnode,2);
    G.Edges.TargetnodeYpos(a) = dataset.nodePosition(garbage.Targetnode,3);
    G.Edges.ActiveEdge(a) = 1;
end

%G Edges cumulative energy calc
for a= 1: height(G.Edges)

G.Edges.Edgeenergy(a,:)=dataset.nodePosition(G.Edges.EndNodes(a,1),4)+dataset.nodePosit
ion(G.Edges.EndNodes(a,2),4);

end

%Cumulative edges energy matrix creation
dataset.edgesenergymatrix=zeros(dataset.nodeNo);
for a = 1 : height(G.Edges)

dataset.edgesenergymatrix(G.Edges.EndNodes(a,1),G.Edges.EndNodes(a,2))=1/G.Edges.Edg
eenergy(a);

```

```

dataset.edgesenergymatrix(G.Edges.EndNodes(a,2),G.Edges.EndNodes(a,1))=1/G.Edges.Edgeenergy(a);
end
ACOedgesenergymatrix=dataset.edgesenergymatrix;

% Graph objects plot

if plotgraphs == 1
    figure('units','normalized','innerposition',[0 0 1 1],'MenuBar','none')
    subplot(1,1,1) % 1,3,1 (Line number,collumn number, graph id) - if you want to show more
    than 1 graph in same windows
    garbage.Xmax = 1500;
    garbage.Xmin = 0;
    garbage.Ymax = 1500;
    garbage.Ymin = 0;
    p = plot(G,'XData',(dataset.nodePosition(:,2)),'YData',(dataset.nodePosition(:,3)));

    line(dataset.nodePosition(1:2,2),dataset.nodePosition(1:2,3),'color','green','marker','o','linestyl
    e','none','markersize',50)
    garbage.ax = gca;
    garbage.ax.XAxis.TickValues = 0:100:1000;
    garbage.ax.YAxis.TickValues = 0:100:1000;
    grid on
    hold on
    title(['Original WSN | ', 'Nodes number: ', num2str(dataset.nodeNo), ' | Nodes range: ',
    num2str(dataset.range)])
    figname=[strcat('FIG-Original-NO-ACO-simulation-',num2str(dataset.nodeNo),'nodes-
    ',datestr(now,'mm-dd-yyyy-HH-MM'),' .png')];
    saveas(gcf,figname)
    pause(2)
end

% Rebuild (clone) original graph for ACO workload

Gaco = G; %ACO Graph creation cloned from original G object

%Creation of some stuff needed
garbage.deadnodelist=[];
garbage.deadnodeneighbors=[];

%% finding routes

% Finding shortest path route
G2 = shortestpathtree(G,1,2);

```

```

%Clone dataset before changes to use in ACO script
ACOdaset = dataset;
pause(2)

%Prepare output filename to save the reports
name=[strcat('NO-ACO-simulation-',num2str(dataset.nodeNo),'nodes-',datestr(now,'mm-dd-
yyyy-HH-MM'),'txt')];
fileID = fopen(fullfile(pwd,name),'w');
fprintf(fileID,'%6s %20s %20s %20s %20s\r\n','|NodeNo|','|No ACO Scene|','|Hops|','|Packets
sent|','|Dead node|');

%% Initialize path existance test for loops

while ~isempty(G2.Edges)
    G2 = shortestpathtree(G,1,2);
    iterationcounter=iterationcounter+1;
    % Test if there is connection between node 1 and 2. If not, terminate!
    if isempty(G2.Edges)
        break
    end

    %Find edges found by shorthestpathtree in main G object
    for a = 1 : height(G2.Edges)
        source = G2.Edges.EndNodes(a,1);
        target = G2.Edges.EndNodes(a,2);
        garbage.edgesrow(a,:) = findedge(G,source,target);
    end

    %Find nodes involved in routing event
    garbage.routingnodes = unique(G2.Edges.EndNodes);
    garbage.routepath = shortestpath(G,1,2);
    %shortestpath(G,1,2)

    %Construct localization dataset to nodes involved in routing event for plot effects
    for a = 1 : length(garbage.routingnodes)
        garbage.b=garbage.routingnodes(a,1);
        dataset.routingnodesPosition(a,:)=dataset.nodePosition(garbage.b,:);
    end

    %Find nodes not involved in routing and its energy to energy increase
    [garbage.Outroutenodes, garbage.Outroutenodesidx] =
    setdiff(dataset.nodePosition(:,1),garbage.routingnodes(:,1),'stable');
    for a = 1 : length(garbage.Outroutenodes)
        garbage.b=garbage.Outroutenodes(a,1);

```

```

    garbage.Outroutenodes(a,2)=dataset.nodePosition(garbage.b,4);
end

%Code block for energy usage (decrease) and packet send count

while min(dataset.nodePosition(:,4))>0
    randomenergyfactor=(dataset.maxenergyfactor-dataset.minenergyfactor).*rand(1,1);

    for a = 1 : length(garbage.routingnodes)
        node=garbage.routingnodes(a,1);
        dataset.nodePosition(node,4)=dataset.nodePosition(node,4)-
dataset.energyconsumptionperCicle^randomenergyfactor+dataset.energyrecoveryperCicle^ra
ndomenergyfactor;

        end
        packet=packet+1;
        if min(garbage.Outroutenodes(:,2))<90
            [garbage.Lowenergyidx] = find(garbage.Outroutenodes(:,2)<=90);
            for b = 1 : length(garbage.Lowenergyidx)
                node=garbage.Outroutenodes(garbage.Lowenergyidx(b,1),1);

garbage.Outroutenodes(b,2)=garbage.Outroutenodes(b,2)+dataset.energyrecoveryperCicle^ra
ndomenergyfactor;
                dataset.nodePosition(node,4)=garbage.Outroutenodes(b,2);
            end
        end
    end
end

%Find dead node ID to discover its neighbors and disable relative edges
[garbage.deadnoderow] = find(dataset.nodePosition(:,4)<=0);
for a = 1 : length(garbage.deadnoderow)
    deadnode=garbage.deadnoderow(a,1);
    for b = 1 : height(G.Edges)
        if ismember(G.Edges.EndNodes(b,1),deadnode) == 1 ||
ismember(G.Edges.EndNodes(b,2),deadnode) == 1
            G.Edges.ActiveEdge(b)=0;
            deadnode;
            pause(2)
        end
    end
end
garbage.deadnodelist(length(garbage.deadnodelist)+1,1)=deadnode;
[garbage.deadedgerow]=find(G.Edges.ActiveEdge==0);

%Mark node energy as NaN

```

```

[garbage.deaddatasetnoderow]=find(dataset.nodePosition(:,4)<=0);
for a = 1 : length(garbage.deaddatasetnoderow)
    b=garbage.deaddatasetnoderow(a,1);
    dataset.nodePosition(b,4)=NaN;
end

%Get number of hops between source and destination for this routing
%session
hopsnumber=length(garbage.routingnodes);

%Informative section in command line - If need to disable plot loop to increase
%processing speed

msg=['Without ACO - Scene #: ',num2str(iterationcounter),' | Hops :
',num2str(hopsnumber),' | Packets sent: ', num2str(packet),' | Dead node: ',
num2str(deadnode),' | Routing nodes: ', num2str(garbage.routepath)];
disp(msg)

if plotgraphs == 1
    %plot for every dead edges iteration's result
    figure('units','normalized','innerposition',[0 0 1 1],'MenuBar','none')
    p = plot(G,'XData',(dataset.nodePosition(:,2)), 'YData',(dataset.nodePosition(:,3)));

    line(dataset.nodePosition(1:2,2),dataset.nodePosition(1:2,3),'color','green','marker','o','linestyl
e','none','markersize',50)
    garbage.ax = gca;
    garbage.ax.XAxis.TickValues = 0:100:1000;
    garbage.ax.YAxis.TickValues = 0:100:1000;
    hold on

    %Plot all dead nodes in red
    for a = 1 : length(garbage.deadnodelist)
        garbage.b=garbage.deaddatasetnoderow(a,1);

        scatter(dataset.nodePosition(garbage.b,2),dataset.nodePosition(garbage.b,3),'MarkerFaceColor',
'red');
    end

    %Title for every iteration's result
    title(['WSN shortest path: ',num2str(iterationcounter),' |hops: ',num2str(hopsnumber),'
|Packets sent: ',num2str(packet),' |Dead node: ',num2str(deadnode),' |Router nodes: ',
num2str(garbage.routepath)])
    grid on
    pause(2)

    %Mark nodes involved in route with green color

```



```

scatter(dataset.routingnodesPosition(:,2),dataset.routingnodesPosition(:,3),'MarkerFaceColor','
green');

    filename=[strcat('FIG-NO-ACO-simulation-',num2str(dataset.nodeNo),'nodes-', 'scene-
',num2str(iterationcounter),'-date-',datestr(now,'mm-dd-yyyy-HH-MM'),'png')];
    saveas(gcf,filename)

    pause(0.2)

end

%Remove dead edges from graph
G = rmedge(G,garbage.deadedgerow(:,1));

%Colect data to append to a external file for later usage - file open
%command line in the begining

report1=[dataset.nodeNo;iterationcounter;hopsnumber;packet;deadnode];
fprintf(fileID,'%6.0f %20.0f %20.0f %20.0f %20.0f\r\n',report1);

%Clear router nodes position to avoid erroneous plot
clear dataset.routingnodesPosition
end

fclose(fileID); %Close external data collector file

if plotgraphs == 1
%Plot all dead nodes in red
    for a = 1 : length(garbage.deadnodelist)
        garbage.b=garbage.deadnodelist(a,1);

scatter(dataset.nodePosition(garbage.b,2),dataset.nodePosition(garbage.b,3),'MarkerFaceColo
r','red');
        end
    end

%Title for the last plot
try
    title(['WSN shortest path: ',num2str(iterationcounter),' |hops: ',num2str(hopsnumber),'
|Packets sent: ',num2str(packet),' |Dead node: ',num2str(deadnode),' |Router nodes: ',
num2str(garbage.routepath),'{\color{red} - ALL ROUTES UNAVAILABLE}'])
catch
    disp('NO ROUTES BETWEEN SOURCE (NODE1) AND TARGET (NODE2)')
end
end

```

```

%Message if there is no path between source and target even in the first iteration
disp('NO ROUTES BETWEEN SOURCE (NODE1) AND TARGET (NODE2)')

%Call aco.m file to run same scenario within aco script
toc
run aco.m

Arquivo aco.m

disp('##### STARTING ACO SCRIPT #####')

%% Main configuration values for ACO simulation

ACOsourcenode = 1;
ACOtargenode = 2;
ACOmaxAntiteration = 100;
ACOantsNo = 21;
ACOpheromone = 1; %tau - pheromone concentration - most papers use 0 as its value
ACOevapcoefficient = 0.2; % 5 percent - (rho) pheromone evaporation rate
ACOalpha = -0.1; %Pheromone exponent
ACObeta = 0.1; %Energy exponent
ACOomega = 1; %Pathloss expoent
ACOpacket=0;
ACOiterationcounter=1;
ACObestfitness=inf;
ACOenergycycles=100;
ACObesttourhistory=[];

%Replot original graph that will be used by ACO - just for comparison

if plotgraphs == 1
    figure('units','normalized','innerposition',[0 0 1 1],'MenuBar','none')
    subplot(1,1,1) % 1,3,1 (Line number,collumn number, graph id) - if you want to show more
    than 1 graph in same windows
    garbage.Xmax = 1500;
    garbage.Xmin = 0;
    garbage.Ymax = 1500;
    garbage.Ymin = 0;
    p =
    plot(Gaco,'XData',(ACOdaset.nodePosition(:,2)), 'YData',(ACOdaset.nodePosition(:,3)));

    line(ACOdaset.nodePosition(1:2,2),ACOdaset.nodePosition(1:2,3),'color','green','marker','
    o','linestyle','none','markersize',50)
    garbage.ax = gca;
    garbage.ax.XAxis.TickValues = 0:100:1000;

```

```

garbage.ax.YAxis.TickValues = 0:100:1000;
grid on
hold on
title(['Original WSN clone to ACO usage | ', 'Nodes number:
', num2str(ACOdaset.nodeNo), ' | Nodes range: ', num2str(ACOdaset.range)])
pause(2)
end

%% Main calcs section

%Get all neighbors from source an target
ACOneighborssource = neighbors(Gaco,ACOsourcenode);
ACOneighborstarget = neighbors(Gaco,ACOtargetnode);

%Get all source and target neighbors
counter=1;
for a = 1 : length(ACOneighborssource)
    for b = 1 : length(ACOneighborstarget)
        ACOallpossiblesourceANDtargets(counter,1)=ACOneighborssource(a,1);
        ACOallpossiblesourceANDtargets(counter,2)=ACOneighborstarget(b,1);
        counter=counter+1;
    end
end

%Get all Shortest paths between all source and target neighbors
for a = 1 : length(ACOallpossiblesourceANDtargets)

ACOallpossibleshortestpaths(a).routes=shortestpathtree(Gaco,ACOallpossiblesourceANDtarg
ets(a,1),ACOallpossiblesourceANDtargets(a,2));
end

%Get the total pheromone, energy and pathloss of each path
for a = 1 : length(ACOallpossibleshortestpaths)
    ACOtotalpathloss(a,:)=1/sum(ACOallpossibleshortestpaths(a).routes.Edges.Pathloss);
    ACOtotalenergy(a,:)=sum(ACOallpossibleshortestpaths(a).routes.Edges.Edgeenergy);
    ACOtotalpheromone(a,:)=ACOPheromone;
end

%First ACO call

for t = 1 : ACOmaxAntiteration

    %create ants
    colony=[];

```

```

    colony =
    createColony(colony,ACOantsNo,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone,A
COalpha,ACObeta,ACOomega);

end

% Fitness function - to know how good solutions found by colony are
for a = 1 : ACOantsNo
    tour = colony(a,:);
%    colony.ant(a).fitness =
fitnessFunction(ACOedgesenergymatrix,ACOdesirebilitymatrix,tour);
    fitness(a,1) =
fitnessFunction(tour,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone);
end

%Find the best fitness
ACOAllNeighborfitness = fitness(:,1);
[minVal,minIndex] = min(ACOAllNeighborfitness);
if minVal < ACObestfitness
    ACObestfitness = fitness(minIndex);
    ACObesttour=colony(minIndex);
end

%Pheromone update
ACOtotalpheromone(ACObesttour,1) =
ACOtotalpheromone(ACObesttour,1)+ACOpheromone;

%Evaporation event
ACOtotalpheromone=(1 - ACOevapcoefficient).*ACOtotalpheromone;

%Append first bebsttour index to best tour history
ACObesttourhistory(end+1,1)=ACObesttour;

%First call for nodes involved

%Find all nodes involved in best tour event

counter=1;
for a = 1 : length(ACOallpossibleshortestpaths)
    for b = 1 : length(ACOallpossibleshortestpaths(a).routes.Edges.EndNodes)
ACOallnodesinvolvedinpossiblepaths(counter,1)=ACOallpossibleshortestpaths(a).routes.Edg
es.EndNodes(b,1);

```

```

ACOallnodesinvolvedinpossiblepaths(counter,2)=ACOallpossibleshortestpaths(a).routes.Edges.EndNodes(b,2);
    counter=counter+1;
    end
end
ACOallnodesinvolvedinpossiblepaths=unique(ACOallnodesinvolvedinpossiblepaths);

%Find nodes involved in besttour routing event

ACObesttournodes=unique(ACOallpossibleshortestpaths(ACObesttour).routes.Edges.EndNodes);

%Find nodes NOT involved in besttour routing event

ACOnodesNOTinvolvedinroutingevent=setdiff(ACOdataset.nodePosition(:,1),ACObestournodes);

%Involved nodes energy usage
for a = 1 : length(ACObestournodes)

ACOdataset.nodePosition(ACObestournodes(a,1),4)=ACOdataset.nodePosition(ACObestournodes(a,1),4)-
dataset.energyconsumptionperCicle^randomenergyfactor+dataset.energyrecoveryperCicle^randomenergyfactor;
    end

    for a = 1 : length(ACOnodesNOTinvolvedinroutingevent)
        if ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)<90

ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)=ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)+dataset.energyrecoveryperCicle^randomenergyfactor;

            end
        end
        ACOpacket=ACOpacket+1;

        for a = 1 : length(ACOallpossibleshortestpaths)
            ACOtotalenergy(a,1)=0;
            for b = 1 : length(ACOallpossibleshortestpaths(a).routes.Edges.EndNodes)
                ACOallpossibleshortestpaths(a).routes.Edges.EndNodes(b,2);

ACOtotalenergy(a,1)=ACOtotalenergy(a,1)+ACOdataset.nodePosition(ACOallpossibleshorte

```

```

stpaths(a).routes.Edges.EndNodes(b,1),4)+ACOdataset.nodePosition(ACOallpossibleshortest
paths(a).routes.Edges.EndNodes(b,2),4);
    end
end

%Prepare output filename to save the reports
name=[strcat('WITH-ACO-simulation-',num2str(ACOdataset.nodeNo),'nodes-
',datestr(now,'mm-dd-yyyy-HH-MM'),' .txt')];
fileID = fopen(fullfile(pwd,name),'w');
fprintf(fileID,'%6s %20s %20s %20s\r\n','|NodeNo|','|WITH ACO Scene|','|Packets sent|','|Best
route|');

%% Main ACO loop and energy consumption

disp('Starting ants activity')

while ~isempty(ACOtotalenergy)

    ACOiterationcounter=ACOiterationcounter+1;

    if ACOdataset.nodePosition(:,4)<=0
        disp('Firt dead node detected')
        break
    end

    %Record best tour history
    if ACObesttourhistory(end) ~= ACObesttour

        %Find all nodes involved in best tour event

        counter=1;
        for a = 1 : length(ACOallpossibleshortestpaths)
            for b = 1 : length(ACOallpossibleshortestpaths(a).routes.Edges.EndNodes)

                ACOallnodesinvolvedinpossiblepaths(counter,1)=ACOallpossibleshortestpaths(a).routes.Edges.EndNodes(b,1);

                ACOallnodesinvolvedinpossiblepaths(counter,2)=ACOallpossibleshortestpaths(a).routes.Edges.EndNodes(b,2);
                counter=counter+1;
            end
        end
        ACOallnodesinvolvedinpossiblepaths=unique(ACOallnodesinvolvedinpossiblepaths);

        %Find nodes involved in besttour routing event

```

```

ACObesttournodes=unique(ACOallpossiblestestpaths(ACObesttour).routes.Edges.EndNodes);

    %Find nodes NOT involved in besttour routing event

ACOnodesNOTinvolvedinroutingevent=setdiff(ACOdataset.nodePosition(:,1),ACObesttournodes);

    ACObesttourhistory(end+1,1)=ACObesttour;

end

    %Involved nodes energy usage
    for a = 1 : length(ACObesttournodes)

ACOdataset.nodePosition(ACObesttournodes(a,1),4)=ACOdataset.nodePosition(ACObesttournodes(a,1),4)-
dataset.energyconsumptionperCicle^randomenergyfactor+dataset.energyrecoveryperCicle^randomenergyfactor;

        end
        ACOpacket=ACOpacket+1;
        for a = 1 : length(ACOnodesNOTinvolvedinroutingevent)
            if ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)<90

ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)=ACOdataset.nodePosition(ACOnodesNOTinvolvedinroutingevent(a,1),4)+dataset.energyrecoveryperCicle^randomenergyfactor;

                end
            end

        for a = 1 : length(ACOallpossiblestestpaths)
            ACOtotalenergy(a,1)=0;
            for b = 1 : length(ACOallpossiblestestpaths(a).routes.Edges.EndNodes)
                ACOallpossiblestestpaths(a).routes.Edges.EndNodes(b,2);

ACOtotalenergy(a,1)=ACOtotalenergy(a,1)+ACOdataset.nodePosition(ACOallpossiblestestpaths(a).routes.Edges.EndNodes(b,1),4)+ACOdataset.nodePosition(ACOallpossiblestestpaths(a).routes.Edges.EndNodes(b,2),4);
            end
        end
    end
end

```

```

msg=['With ACO - Scene #: ',num2str(ACOiterationcounter),' | Packets sent: ',
num2str(ACOpacket),' | Best route: ', num2str(ACObesttour),' | Routing nodes: ',
num2str(ACObesttournodes)];
disp(msg)

```

```

report1=[dataset.nodeNo;iterationcounter;ACOpacket;ACObesttour];
fprintf(fileID,'%6.0f %20.0f %20.0f %20.0f\r\n',report1);

```

```

%Second ACO call

```

```

for t = 1 : ACOmaxAntiteration

```

```

    %create ants
    colony=[];
    colony =

```

```

createColony(colony,ACOantsNo,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone,A
COalpha,ACObeta,ACOomega);

```

```

end

```

```

% Fitness function - to know how good solutions found by colony are
for a = 1 : ACOantsNo

```

```

    tour = colony(a,:);

```

```

    fitness(a,1) =
fitnessFunction(tour,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone);
end

```

```

%Find the best fitness

```

```

ACOAllNeighborfitness = fitness(:,1);
[minVal,minIndex] = min(ACOAllNeighborfitness);
if minVal < ACObestfitness
    ACObestfitness = fitness(minIndex);
    ACObesttour=colony(minIndex);
end

```

```

%Pheromone update

```

```

    ACOtotalpheromone(ACObesttour,1) =
ACOtotalpheromone(ACObesttour,1)+ACOPheromone;

```

```

%Evaporation event

```

```

    ACOtotalpheromone=(1 - ACOevapcoefficient).*ACOtotalpheromone;

```


end

fclose(fileID); %Close external data collector file

Arquivo createColony.m

```
function [ colony ] =
createColony(colony,ACOantsNo,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone,A
COalpha,ACObeta,ACOomega)
```

```
    for a = 1 : ACOantsNo
        for b = 1 : length(ACOttotalpheromone)
```

```
            ACOp_allNodes =
ACOttotalpheromone(:,1).^ACOalpha.*ACOttotalenergy(:,1).^ACObeta;
```

```
            P = ACOp_allNodes./ sum(ACOp_allNodes);
```

```
            ACOnextroute = rouletteWheel(P);
```

```
            colony(a,b) = ACOnextroute;
        end
```

```
    end
```

end

Arquivo rouletteWheel.m

```
function [ ACOnextroute ] = rouletteWheel( P )
```

```
cumsumP = cumsum(P);
```

```
r = rand();
```

```
ACOnextroute = find(r <= cumsumP);
```

```
ACOnextroute = ACOnextroute(1);
```

end

Arquivo fitnessFunction.m

```
function [ fitness ] =
fitnessFunction(tour,ACOttotalpathloss,ACOttotalenergy,ACOttotalpheromone)
```

```
fitness = 0;  
for i = 1 : length(tour)  
    ACOcurrentroute = tour(i);  
    fitness = fitness + ACOtotalpheromone(ACOcurrentroute, 1);  
end
```

Apêndice B

ASPECTOS DA SIMULAÇÃO

Saídas de console durante a simulação sem ACO

Without ACO - Scene #: 2 | Hops : 7 | Packets sent: 5189 | Dead node: 58 | Routing nodes: 1
4 26 12 58 53 2

Without ACO - Scene #: 3 | Hops : 7 | Packets sent: 5304 | Dead node: 26 | Routing nodes: 1
4 26 12 65 53 2

Without ACO - Scene #: 4 | Hops : 7 | Packets sent: 6104 | Dead node: 4 | Routing nodes: 1 4
34 54 16 21 2

Without ACO - Scene #: 5 | Hops : 7 | Packets sent: 11481 | Dead node: 21 | Routing nodes: 1
37 34 54 16 21 2

Without ACO - Scene #: 6 | Hops : 7 | Packets sent: 12034 | Dead node: 16 | Routing nodes: 1
37 34 54 16 48 2

Without ACO - Scene #: 7 | Hops : 7 | Packets sent: 17829 | Dead node: 48 | Routing nodes: 1
37 34 54 24 48 2

Without ACO - Scene #: 8 | Hops : 7 | Packets sent: 24157 | Dead node: 65 | Routing nodes: 1
37 34 62 65 53 2

Without ACO - Scene #: 9 | Hops : 8 | Packets sent: 30450 | Dead node: 18 | Routing nodes: 1
30 49 44 23 18 53 2

Without ACO - Scene #: 10 | Hops : 8 | Packets sent: 30607 | Dead node: 23 | Routing nodes:
1 30 49 44 23 29 53 2

Without ACO - Scene #: 11 | Hops : 8 | Packets sent: 30861 | Dead node: 30 | Routing nodes:
1 30 49 44 64 5 53 2

Without ACO - Scene #: 12 | Hops : 8 | Packets sent: 37015 | Dead node: 14 | Routing nodes:
1 37 34 6 14 5 53 2

Without ACO - Scene #: 13 | Hops : 8 | Packets sent: 37169 | Dead node: 5 | Routing nodes: 1
37 34 8 27 5 53 2

Without ACO - Scene #: 14 | Hops : 8 | Packets sent: 43176 | Dead node: 29 | Routing nodes:
1 37 34 8 27 29 53 2

Without ACO - Scene #: 15 | Hops : 8 | Packets sent: 43441 | Dead node: 27 | Routing nodes:
1 37 34 8 27 67 53 2

Without ACO - Scene #: 16 | Hops : 8 | Packets sent: 49149 | Dead node: 67 | Routing nodes:
1 37 34 8 64 67 53 2

NO ROUTES BETWEEN SOURCE (NODE1) AND TARGET (NODE2)

Saídas de console durante a simulação com ACO

With ACO - Scene #: 2 | Packets sent: 101 | Best route: 1 | Routing nodes: 4 16 21 26 52 |
Nodes energy: 94.3271 / 80.3867 / 95.698 / 81.9598 / 93.4806

With ACO - Scene #: 3 | Packets sent: 201 | Best route: 8 | Routing nodes: 16 34 37 48 54 |
Nodes energy: 78.989 / 90.5912 / 90.6604 / 98.4678 / 90.6839

With ACO - Scene #: 4 | Packets sent: 301 | Best route: 5 | Routing nodes: 4 16 26 30 48
52 | Nodes energy: 92.9294 / 77.5912 / 90.4176 / 90.8464 / 97.07 / 92.0828

With ACO - Scene #: 5 | Packets sent: 401 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 91.5316 / 97.7819 / 89.0198 / 89.4486 / 96.792 / 90.8836

With ACO - Scene #: 6 | Packets sent: 501 | Best route: 5 | Routing nodes: 4 16 26 30 48
52 | Nodes energy: 90.1339 / 90.9768 / 87.6221 / 88.0509 / 95.6723 / 90.685

With ACO - Scene #: 7 | Packets sent: 601 | Best route: 2 | Routing nodes: 4 16 26 48 52 |
Nodes energy: 88.7361 / 89.5791 / 86.2243 / 94.2745 / 89.2873

With ACO - Scene #: 8 | Packets sent: 701 | Best route: 2 | Routing nodes: 4 16 26 48 52 |
Nodes energy: 87.3384 / 88.1813 / 84.8266 / 92.8768 / 87.8895

With ACO - Scene #: 9 | Packets sent: 801 | Best route: 9 | Routing nodes: 34 37 53 62 65 |
Nodes energy: 90.179 / 90.2482 / 95.3942 / 90.3955 / 99.9241

With ACO - Scene #: 10 | Packets sent: 901 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 90.8684 / 90.7258 / 94.3002 / 90.3277 / 90.5953 / 90.434

With ACO - Scene #: 11 | Packets sent: 1001 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 89.4706 / 89.328 / 92.9025 / 88.93 / 89.1976 / 89.0362

With ACO - Scene #: 12 | Packets sent: 1101 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 87.9303 / 90.7523 / 90.8216 / 91.479 / 90.2717

With ACO - Scene #: 13 | Packets sent: 1201 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 86.5325 / 89.3546 / 89.4238 / 90.0813 / 88.874

With ACO - Scene #: 14 | Packets sent: 1301 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 85.1348 / 87.9568 / 88.0261 / 88.6835 / 87.4762

With ACO - Scene #: 15 | Packets sent: 1401 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 90.044 / 96.3841 / 90.4889 / 93.9965 / 90.4714

With ACO - Scene #: 16 | Packets sent: 1501 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 88.6462 / 94.9864 / 89.0911 / 92.5987 / 89.0736

With ACO - Scene #: 17 | Packets sent: 1601 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 87.2485 / 93.5886 / 87.6934 / 91.201 / 87.6759

With ACO - Scene #: 18 | Packets sent: 1701 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 90.6359 / 91.5047 / 90.5013 / 90.5705 / 90.0207

With ACO - Scene #: 19 | Packets sent: 1801 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 89.2382 / 90.107 / 89.1035 / 89.1728 / 88.6229

With ACO - Scene #: 20 | Packets sent: 1901 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 87.8404 / 88.7092 / 87.7058 / 87.775 / 87.2252

With ACO - Scene #: 21 | Packets sent: 2001 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 90.7785 / 86.4426 / 87.3114 / 90.2378 / 90.7565 / 90.5952

With ACO - Scene #: 22 | Packets sent: 2101 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 89.3808 / 85.0449 / 85.9137 / 88.8401 / 89.3587 / 89.1974

With ACO - Scene #: 23 | Packets sent: 2201 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 87.983 / 83.6471 / 84.5159 / 87.4423 / 87.961 / 87.7996

With ACO - Scene #: 24 | Packets sent: 2301 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 82.2494 / 90.2503 / 90.3195 / 90.2424 / 90.7552

With ACO - Scene #: 25 | Packets sent: 2401 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 80.8516 / 88.8525 / 88.9217 / 88.8447 / 89.3575

With ACO - Scene #: 26 | Packets sent: 2501 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 79.4539 / 87.4547 / 87.524 / 87.4469 / 87.9597

With ACO - Scene #: 27 | Packets sent: 2601 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 78.0561 / 86.057 / 86.1262 / 86.0491 / 86.5619

With ACO - Scene #: 28 | Packets sent: 2701 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 76.6583 / 84.6592 / 84.7285 / 84.6514 / 85.1642

With ACO - Scene #: 29 | Packets sent: 2801 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 90.5275 / 75.2606 / 90.9724 / 83.2536 / 90.3441

With ACO - Scene #: 30 | Packets sent: 2901 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 89.1297 / 92.1909 / 89.5746 / 90.5055 / 90.7888 / 90.2203

With ACO - Scene #: 31 | Packets sent: 3001 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 87.732 / 90.7931 / 88.1768 / 89.1077 / 89.391 / 88.8226

With ACO - Scene #: 32 | Packets sent: 3101 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 86.3342 / 89.3953 / 86.7791 / 87.7099 / 87.9932 / 87.4248

With ACO - Scene #: 33 | Packets sent: 3201 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 84.9364 / 87.9976 / 85.3813 / 86.3122 / 86.5955 / 86.0271

With ACO - Scene #: 34 | Packets sent: 3301 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 83.5387 / 86.5998 / 83.9836 / 84.9144 / 85.1977 / 84.6293

With ACO - Scene #: 35 | Packets sent: 3401 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 82.1409 / 85.2021 / 82.5858 / 83.5167 / 83.8 / 83.2316

With ACO - Scene #: 36 | Packets sent: 3501 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 90.1604 / 90.2296 / 82.4022 / 90.9689 / 98.5264

With ACO - Scene #: 37 | Packets sent: 3601 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 88.7626 / 88.8319 / 81.0045 / 89.5711 / 97.1286

With ACO - Scene #: 38 | Packets sent: 3701 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 87.3649 / 87.4341 / 79.6067 / 88.1733 / 95.7308

With ACO - Scene #: 39 | Packets sent: 3801 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 85.9671 / 86.0363 / 78.2089 / 86.7756 / 94.3331

With ACO - Scene #: 40 | Packets sent: 3901 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 84.5694 / 84.6386 / 76.8112 / 85.3778 / 92.9353

With ACO - Scene #: 41 | Packets sent: 4001 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 83.1716 / 83.2408 / 75.4134 / 83.9801 / 91.5376

With ACO - Scene #: 42 | Packets sent: 4101 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 81.7738 / 81.8431 / 74.0157 / 82.5823 / 90.1398

With ACO - Scene #: 43 | Packets sent: 4201 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 80.3761 / 80.4453 / 72.6179 / 81.1845 / 88.7421

With ACO - Scene #: 44 | Packets sent: 4301 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 90.5988 / 90.6173 / 90.0171 / 90.0581 / 90.9175

With ACO - Scene #: 45 | Packets sent: 4401 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 89.201 / 89.2196 / 88.6603 / 90.7259 / 89.5197

With ACO - Scene #: 46 | Packets sent: 4501 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 87.8218 / 90.805 / 90.8743 / 89.3281 / 90.6653

With ACO - Scene #: 47 | Packets sent: 4601 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 90.7599 / 86.4241 / 90.2192 / 90.0034 / 87.9304 / 90.0931

With ACO - Scene #: 48 | Packets sent: 4701 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 89.3622 / 85.0263 / 88.8215 / 88.6056 / 86.5326 / 88.6953

With ACO - Scene #: 49 | Packets sent: 4801 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 87.9644 / 83.6285 / 87.4237 / 87.2079 / 85.1349 / 87.2976

With ACO - Scene #: 50 | Packets sent: 4901 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 86.5666 / 82.2308 / 86.026 / 85.8101 / 83.7371 / 85.8998

With ACO - Scene #: 51 | Packets sent: 5001 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 85.1689 / 80.833 / 84.6282 / 84.4124 / 82.3394 / 84.5021

With ACO - Scene #: 52 | Packets sent: 5101 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 83.7711 / 79.4353 / 83.2305 / 83.0146 / 80.9416 / 83.1043

With ACO - Scene #: 53 | Packets sent: 5201 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 82.3734 / 78.0375 / 81.8327 / 81.6168 / 79.5438 / 81.7065

With ACO - Scene #: 54 | Packets sent: 5301 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 80.9756 / 76.6397 / 80.4349 / 80.2191 / 78.1461 / 80.3088

With ACO - Scene #: 55 | Packets sent: 5401 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 79.5779 / 75.242 / 79.0372 / 78.8213 / 76.7483 / 78.911

With ACO - Scene #: 56 | Packets sent: 5501 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 78.1801 / 73.8442 / 77.6394 / 77.4236 / 75.3506 / 77.5133

With ACO - Scene #: 57 | Packets sent: 5601 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 76.7823 / 72.4465 / 76.2417 / 76.0258 / 73.9528 / 76.1155

With ACO - Scene #: 58 | Packets sent: 5701 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 75.3846 / 71.0487 / 74.8439 / 74.6281 / 72.5551 / 74.7178

With ACO - Scene #: 59 | Packets sent: 5801 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 73.9868 / 69.651 / 73.4461 / 73.2303 / 71.1573 / 73.32

With ACO - Scene #: 60 | Packets sent: 5901 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 72.5891 / 68.2532 / 72.0484 / 71.8325 / 69.7595 / 71.9222

With ACO - Scene #: 61 | Packets sent: 6001 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 71.1913 / 66.8554 / 70.6506 / 70.4348 / 68.3618 / 70.5245

With ACO - Scene #: 62 | Packets sent: 6101 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 69.7936 / 65.4577 / 69.2529 / 69.037 / 66.964 / 69.1267

With ACO - Scene #: 63 | Packets sent: 6201 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 68.3958 / 64.0599 / 67.8551 / 67.6393 / 65.5663 / 67.729

With ACO - Scene #: 64 | Packets sent: 6301 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 66.998 / 62.6622 / 66.4574 / 66.2415 / 64.1685 / 66.3312

With ACO - Scene #: 65 | Packets sent: 6401 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 65.6003 / 61.2644 / 65.0596 / 64.8437 / 62.7707 / 64.9334

With ACO - Scene #: 66 | Packets sent: 6501 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 59.8667 / 90.5904 / 90.3928 / 90.4621 / 90.2531

With ACO - Scene #: 67 | Packets sent: 6601 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 58.4689 / 89.1927 / 88.9951 / 89.0643 / 88.8554

With ACO - Scene #: 68 | Packets sent: 6701 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 57.0711 / 87.7949 / 87.5973 / 87.6665 / 87.4576

With ACO - Scene #: 69 | Packets sent: 6801 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 55.6734 / 86.3972 / 86.1996 / 86.2688 / 86.0599

With ACO - Scene #: 70 | Packets sent: 6901 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 90.8126 / 54.2756 / 84.9994 / 90.2719 / 90.0561 / 90.1458

With ACO - Scene #: 71 | Packets sent: 7001 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 89.4149 / 52.8779 / 83.6017 / 88.8742 / 88.6583 / 88.748

With ACO - Scene #: 72 | Packets sent: 7101 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 88.0171 / 51.4801 / 82.2039 / 87.4764 / 87.2606 / 87.3503

With ACO - Scene #: 73 | Packets sent: 7201 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 86.6193 / 50.0823 / 80.8061 / 86.0787 / 85.8628 / 85.9525

With ACO - Scene #: 74 | Packets sent: 7301 | Best route: 4 | Routing nodes: 4 16 21 26 30
52 | Nodes energy: 85.2216 / 48.6846 / 79.4084 / 84.6809 / 84.4651 / 84.5548

With ACO - Scene #: 75 | Packets sent: 7401 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 83.8238 / 90.7032 / 83.2831 / 90.9313 / 90.7038

With ACO - Scene #: 76 | Packets sent: 7501 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 82.4261 / 89.3055 / 81.8854 / 89.5336 / 89.3061

With ACO - Scene #: 77 | Packets sent: 7601 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 81.0283 / 87.9077 / 80.4876 / 88.1358 / 87.9083

With ACO - Scene #: 78 | Packets sent: 7701 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 79.6306 / 86.5099 / 79.0899 / 86.7381 / 86.5106

With ACO - Scene #: 79 | Packets sent: 7801 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 78.2328 / 85.1122 / 77.6921 / 85.3403 / 85.1128

With ACO - Scene #: 80 | Packets sent: 7901 | Best route: 3 | Routing nodes: 4 12 26 53 58
| Nodes energy: 76.835 / 83.7144 / 76.2944 / 83.9425 / 83.715

With ACO - Scene #: 81 | Packets sent: 8001 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 75.4373 / 90.6514 / 74.8966 / 90.9398 / 90.0559

With ACO - Scene #: 82 | Packets sent: 8101 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 74.0395 / 89.2537 / 73.4988 / 89.542 / 88.6581

With ACO - Scene #: 83 | Packets sent: 8201 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 72.6418 / 87.8559 / 72.1011 / 88.1442 / 87.2604

With ACO - Scene #: 84 | Packets sent: 8301 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 71.244 / 86.4582 / 70.7033 / 86.7465 / 85.8626

With ACO - Scene #: 85 | Packets sent: 8401 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 69.8463 / 85.0604 / 69.3056 / 85.3487 / 84.4649

With ACO - Scene #: 86 | Packets sent: 8501 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 68.4485 / 83.6626 / 67.9078 / 83.951 / 83.0671

With ACO - Scene #: 87 | Packets sent: 8601 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 67.0507 / 82.2649 / 66.5101 / 82.5532 / 81.6694

With ACO - Scene #: 88 | Packets sent: 8701 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 65.653 / 80.8671 / 65.1123 / 81.1555 / 80.2716

With ACO - Scene #: 89 | Packets sent: 8801 | Best route: 2 | Routing nodes: 4 16 26 48 52
| Nodes energy: 64.2552 / 79.4694 / 63.7145 / 79.7577 / 78.8738

With ACO - Scene #: 90 | Packets sent: 8901 | Best route: 1 | Routing nodes: 4 16 21 26 52
| Nodes energy: 62.8575 / 78.0716 / 90.8229 / 62.3168 / 77.4761

With ACO - Scene #: 91 | Packets sent: 9001 | Best route: 1 | Routing nodes: 4 16 21 26 52
| Nodes energy: 61.4597 / 76.6738 / 89.4251 / 60.919 / 76.0783

With ACO - Scene #: 92 | Packets sent: 9101 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 60.0619 / 90.2011 / 59.5213 / 90.9518 / 90.4293 / 90.2018

With ACO - Scene #: 93 | Packets sent: 9201 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 58.6642 / 88.8034 / 58.1235 / 89.554 / 89.0315 / 88.804

With ACO - Scene #: 94 | Packets sent: 9301 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 57.2664 / 87.4056 / 56.7257 / 88.1562 / 87.6337 / 87.4062

With ACO - Scene #: 95 | Packets sent: 9401 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 55.8687 / 86.0079 / 55.328 / 86.7585 / 86.236 / 86.0085

With ACO - Scene #: 96 | Packets sent: 9501 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 54.4709 / 84.6101 / 53.9302 / 85.3607 / 84.8382 / 84.6107

With ACO - Scene #: 97 | Packets sent: 9601 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 53.0732 / 83.2124 / 52.5325 / 83.963 / 83.4405 / 83.213

With ACO - Scene #: 98 | Packets sent: 9701 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 51.6754 / 81.8146 / 51.1347 / 82.5652 / 82.0427 / 81.8152

With ACO - Scene #: 99 | Packets sent: 9801 | Best route: 6 | Routing nodes: 4 12 26 30 53
58 | Nodes energy: 50.2776 / 80.4168 / 49.737 / 81.1675 / 80.645 / 80.4175

With ACO - Scene #: 100 | Packets sent: 9901 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 48.8799 / 79.0191 / 48.3392 / 79.7697 / 79.2472 / 79.0197

With ACO - Scene #: 101 | Packets sent: 10001 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 47.4821 / 77.6213 / 46.9414 / 78.3719 / 77.8494 / 77.6219

With ACO - Scene #: 102 | Packets sent: 10101 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 46.0844 / 76.2236 / 45.5437 / 76.9742 / 76.4517 / 76.2242

With ACO - Scene #: 103 | Packets sent: 10201 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 44.6866 / 74.8258 / 44.1459 / 75.5764 / 75.0539 / 74.8264

With ACO - Scene #: 104 | Packets sent: 10301 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 43.2889 / 73.4281 / 42.7482 / 74.1787 / 73.6562 / 73.4287

With ACO - Scene #: 105 | Packets sent: 10401 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 41.8911 / 72.0303 / 41.3504 / 72.7809 / 72.2584 / 72.0309

With ACO - Scene #: 106 | Packets sent: 10501 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 40.4933 / 70.6325 / 39.9527 / 71.3832 / 70.8606 / 70.6331

With ACO - Scene #: 107 | Packets sent: 10601 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 39.0956 / 69.2348 / 38.5549 / 69.9854 / 69.4629 / 69.2354

With ACO - Scene #: 108 | Packets sent: 10701 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 37.6978 / 67.837 / 37.1571 / 68.5876 / 68.0651 / 67.8376

With ACO - Scene #: 109 | Packets sent: 10801 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 36.3001 / 66.4393 / 35.7594 / 67.1899 / 66.6674 / 66.4399

With ACO - Scene #: 110 | Packets sent: 10901 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 34.9023 / 65.0415 / 34.3616 / 65.7921 / 65.2696 / 65.0421

With ACO - Scene #: 111 | Packets sent: 11001 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 33.5045 / 63.6437 / 32.9639 / 64.3944 / 63.8719 / 63.6444

With ACO - Scene #: 112 | Packets sent: 11101 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 32.1068 / 62.246 / 31.5661 / 62.9966 / 62.4741 / 62.2466

With ACO - Scene #: 113 | Packets sent: 11201 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 30.709 / 60.8482 / 30.1684 / 61.5989 / 61.0763 / 60.8488

With ACO - Scene #: 114 | Packets sent: 11301 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 29.3113 / 59.4505 / 28.7706 / 60.2011 / 59.6786 / 59.4511

With ACO - Scene #: 115 | Packets sent: 11401 | Best route: 6 | Routing nodes: 4 12 26 30
53 58 | Nodes energy: 27.9135 / 58.0527 / 27.3728 / 58.8033 / 58.2808 / 58.0533

With ACO - Scene #: 116 | Packets sent: 11501 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 26.5158 / 90.0595 / 25.9751 / 57.4056 / 90.1866 / 90.4495

With ACO - Scene #: 117 | Packets sent: 11601 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 25.118 / 88.6617 / 24.5773 / 56.0078 / 88.7889 / 89.0518

With ACO - Scene #: 118 | Packets sent: 11701 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 23.7202 / 87.264 / 23.1796 / 54.6101 / 87.3911 / 87.654

With ACO - Scene #: 119 | Packets sent: 11801 | Best route: 5 | Routing nodes: 4 16 26 30 48 52 | Nodes energy: 22.3225 / 85.8662 / 21.7818 / 53.2123 / 85.9934 / 86.2562

With ACO - Scene #: 120 | Packets sent: 11901 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 84.4684 / 90.9841 / 90.7152 / 90.7844 / 90.5755

With ACO - Scene #: 121 | Packets sent: 12001 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 83.0707 / 89.5863 / 89.3174 / 89.3866 / 89.1777

With ACO - Scene #: 122 | Packets sent: 12101 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 81.6729 / 88.1885 / 87.9196 / 87.9889 / 87.78

With ACO - Scene #: 123 | Packets sent: 12201 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 80.2752 / 86.7908 / 86.5219 / 86.5911 / 86.3822

With ACO - Scene #: 124 | Packets sent: 12301 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 78.8774 / 85.393 / 85.1241 / 85.1934 / 84.9844

With ACO - Scene #: 125 | Packets sent: 12401 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 77.4797 / 83.9953 / 83.7264 / 83.7956 / 83.5867

With ACO - Scene #: 126 | Packets sent: 12501 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 76.0819 / 82.5975 / 82.3286 / 82.3978 / 82.1889

With ACO - Scene #: 127 | Packets sent: 12601 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 74.6841 / 81.1997 / 80.9309 / 81.0001 / 80.7912

With ACO - Scene #: 128 | Packets sent: 12701 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 73.2864 / 79.802 / 79.5331 / 79.6023 / 79.3934

With ACO - Scene #: 129 | Packets sent: 12801 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 71.8886 / 78.4042 / 78.1353 / 78.2046 / 77.9957

With ACO - Scene #: 130 | Packets sent: 12901 | Best route: 7 | Routing nodes: 16 21 34 37 54 | Nodes energy: 70.4909 / 77.0065 / 76.7376 / 76.8068 / 76.5979

With ACO - Scene #: 131 | Packets sent: 13001 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 90.8994 / 90.164 / 90.3587 / 90.3921 / 90.1646

With ACO - Scene #: 132 | Packets sent: 13101 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 90.7754 / 90.1232 / 90.1924 / 90.509 / 90.9691

With ACO - Scene #: 133 | Packets sent: 13201 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 89.3776 / 88.7254 / 88.7947 / 89.1112 / 89.5713

With ACO - Scene #: 134 | Packets sent: 13301 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 87.9799 / 87.3277 / 87.3969 / 87.7135 / 88.1736

With ACO - Scene #: 135 | Packets sent: 13401 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 86.5821 / 85.9299 / 85.9992 / 86.3157 / 86.7758

With ACO - Scene #: 136 | Packets sent: 13501 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 85.1844 / 84.5322 / 84.6014 / 84.9179 / 85.378

With ACO - Scene #: 137 | Packets sent: 13601 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 83.7866 / 83.1344 / 83.2036 / 83.5202 / 83.9803

With ACO - Scene #: 138 | Packets sent: 13701 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 82.3889 / 81.7367 / 81.8059 / 82.1224 / 82.5825

With ACO - Scene #: 139 | Packets sent: 13801 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 80.9911 / 80.3389 / 80.4081 / 80.7247 / 81.1848

With ACO - Scene #: 140 | Packets sent: 13901 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 79.5933 / 78.9411 / 79.0104 / 79.3269 / 79.787

With ACO - Scene #: 141 | Packets sent: 14001 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 77.5434 / 77.6126 / 90.9654 / 90.6279 / 90.301

With ACO - Scene #: 142 | Packets sent: 14101 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 76.1456 / 76.2149 / 89.5677 / 89.2302 / 88.9032

With ACO - Scene #: 143 | Packets sent: 14201 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 74.7479 / 74.8171 / 88.1699 / 87.8324 / 87.5055

With ACO - Scene #: 144 | Packets sent: 14301 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 73.3501 / 73.4193 / 86.7722 / 86.4347 / 86.1077

With ACO - Scene #: 145 | Packets sent: 14401 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 71.9524 / 72.0216 / 85.3744 / 85.0369 / 84.7099

With ACO - Scene #: 146 | Packets sent: 14501 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 70.5546 / 70.6238 / 83.9766 / 83.6391 / 83.3122

With ACO - Scene #: 147 | Packets sent: 14601 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 69.1568 / 69.2261 / 82.5789 / 82.2414 / 81.9144

With ACO - Scene #: 148 | Packets sent: 14701 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 67.7591 / 67.8283 / 81.1811 / 80.8436 / 80.5167

With ACO - Scene #: 149 | Packets sent: 14801 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 66.3613 / 66.4306 / 79.7834 / 79.4459 / 79.1189

With ACO - Scene #: 150 | Packets sent: 14901 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 64.9636 / 65.0328 / 78.3856 / 78.0481 / 77.7211

With ACO - Scene #: 151 | Packets sent: 15001 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 63.5658 / 63.635 / 76.9879 / 76.6504 / 76.3234

With ACO - Scene #: 152 | Packets sent: 15101 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 62.1681 / 62.2373 / 75.5901 / 75.2526 / 74.9256

With ACO - Scene #: 153 | Packets sent: 15201 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 60.7703 / 60.8395 / 74.1923 / 73.8548 / 73.5279

With ACO - Scene #: 154 | Packets sent: 15301 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 59.3725 / 59.4418 / 72.7946 / 72.4571 / 72.1301

With ACO - Scene #: 155 | Packets sent: 15401 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 57.9748 / 58.044 / 71.3968 / 71.0593 / 70.7324

With ACO - Scene #: 156 | Packets sent: 15501 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 56.577 / 56.6462 / 69.9991 / 69.6616 / 69.3346

With ACO - Scene #: 157 | Packets sent: 15601 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 55.1793 / 55.2485 / 68.6013 / 68.2638 / 67.9368

With ACO - Scene #: 158 | Packets sent: 15701 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 53.7815 / 53.8507 / 67.2036 / 66.8661 / 66.5391

With ACO - Scene #: 159 | Packets sent: 15801 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 52.3837 / 52.453 / 65.8058 / 65.4683 / 65.1413

With ACO - Scene #: 160 | Packets sent: 15901 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 50.986 / 51.0552 / 64.408 / 64.0705 / 63.7436

With ACO - Scene #: 161 | Packets sent: 16001 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 49.5882 / 49.6575 / 63.0103 / 62.6728 / 62.3458

With ACO - Scene #: 162 | Packets sent: 16101 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 48.1905 / 48.2597 / 61.6125 / 61.275 / 60.9481

With ACO - Scene #: 163 | Packets sent: 16201 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 90.4872 / 90.0223 / 90.3921 / 90.9321 / 90.2513 / 90.7718

With ACO - Scene #: 164 | Packets sent: 16301 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 89.0895 / 88.6245 / 88.9943 / 89.5343 / 88.8536 / 89.3741

With ACO - Scene #: 165 | Packets sent: 16401 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 87.6917 / 87.2268 / 87.5966 / 88.1366 / 87.4558 / 87.9763

With ACO - Scene #: 166 | Packets sent: 16501 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 86.2939 / 85.829 / 86.1988 / 86.7388 / 86.0581 / 86.5786

With ACO - Scene #: 167 | Packets sent: 16601 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 84.8962 / 84.4313 / 84.8011 / 85.3411 / 84.6603 / 85.1808

With ACO - Scene #: 168 | Packets sent: 16701 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 83.4984 / 83.0335 / 83.4033 / 83.9433 / 83.2626 / 83.783

With ACO - Scene #: 169 | Packets sent: 16801 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 82.1007 / 81.6357 / 82.0056 / 82.5455 / 81.8648 / 82.3853

With ACO - Scene #: 170 | Packets sent: 16901 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 80.7029 / 80.238 / 80.6078 / 81.1478 / 80.467 / 80.9875

With ACO - Scene #: 171 | Packets sent: 17001 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 79.3052 / 78.8402 / 79.21 / 79.75 / 79.0693 / 79.5898

With ACO - Scene #: 172 | Packets sent: 17101 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 77.9074 / 77.4425 / 77.8123 / 78.3523 / 77.6715 / 78.192

With ACO - Scene #: 173 | Packets sent: 17201 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 76.5096 / 76.0447 / 76.4145 / 76.9545 / 76.2738 / 76.7943

With ACO - Scene #: 174 | Packets sent: 17301 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 75.1119 / 74.647 / 75.0168 / 75.5568 / 74.876 / 75.3965

With ACO - Scene #: 175 | Packets sent: 17401 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 73.7141 / 73.2492 / 73.619 / 74.159 / 73.4783 / 73.9987

With ACO - Scene #: 176 | Packets sent: 17501 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 72.3164 / 71.8514 / 72.2213 / 72.7612 / 72.0805 / 72.601

With ACO - Scene #: 177 | Packets sent: 17601 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 70.9186 / 70.4537 / 70.8235 / 71.3635 / 70.6827 / 71.2032

With ACO - Scene #: 178 | Packets sent: 17701 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 69.5208 / 69.0559 / 69.4257 / 69.9657 / 69.285 / 69.8055

With ACO - Scene #: 179 | Packets sent: 17801 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 68.1231 / 67.6582 / 68.028 / 68.568 / 67.8872 / 68.4077

With ACO - Scene #: 180 | Packets sent: 17901 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 66.7253 / 66.2604 / 66.6302 / 67.1702 / 66.4895 / 67.01

With ACO - Scene #: 181 | Packets sent: 18001 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 65.3276 / 64.8627 / 65.2325 / 65.7725 / 65.0917 / 65.6122

With ACO - Scene #: 182 | Packets sent: 18101 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 63.9298 / 63.4649 / 63.8347 / 64.3747 / 63.6939 / 64.2144

With ACO - Scene #: 183 | Packets sent: 18201 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 62.5321 / 62.0671 / 62.4369 / 62.9769 / 62.2962 / 62.8167

With ACO - Scene #: 184 | Packets sent: 18301 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 61.1343 / 60.6694 / 61.0392 / 61.5792 / 60.8984 / 61.4189

With ACO - Scene #: 185 | Packets sent: 18401 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 59.7365 / 59.2716 / 59.6414 / 60.1814 / 59.5007 / 60.0212

With ACO - Scene #: 186 | Packets sent: 18501 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 58.3388 / 57.8739 / 58.2437 / 58.7837 / 58.1029 / 58.6234

With ACO - Scene #: 187 | Packets sent: 18601 | Best route: 4 | Routing nodes: 4 16 21 26 30 52 | Nodes energy: 56.941 / 56.4761 / 56.8459 / 57.3859 / 56.7052 / 57.2256

With ACO - Scene #: 188 | Packets sent: 18701 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 55.5433 / 55.0783 / 55.9881 / 90.7414 / 55.8279

With ACO - Scene #: 189 | Packets sent: 18801 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 54.1455 / 53.6806 / 54.5904 / 89.3437 / 54.4301

With ACO - Scene #: 190 | Packets sent: 18901 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 52.7478 / 52.2828 / 53.1926 / 87.9459 / 53.0324

With ACO - Scene #: 191 | Packets sent: 19001 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 51.35 / 50.8851 / 51.7949 / 86.5481 / 51.6346

With ACO - Scene #: 192 | Packets sent: 19101 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 49.9522 / 49.4873 / 50.3971 / 85.1504 / 50.2369

With ACO - Scene #: 193 | Packets sent: 19201 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 48.5545 / 48.0896 / 48.9994 / 83.7526 / 48.8391

With ACO - Scene #: 194 | Packets sent: 19301 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 47.1567 / 46.6918 / 47.6016 / 82.3549 / 47.4413

With ACO - Scene #: 195 | Packets sent: 19401 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 45.759 / 45.294 / 46.2038 / 80.9571 / 46.0436

With ACO - Scene #: 196 | Packets sent: 19501 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 44.3612 / 43.8963 / 44.8061 / 79.5594 / 44.6458

With ACO - Scene #: 197 | Packets sent: 19601 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 42.9635 / 42.4985 / 90.9283 / 43.4083 / 43.2481

With ACO - Scene #: 198 | Packets sent: 19701 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 41.5657 / 41.1008 / 89.5305 / 42.0106 / 41.8503

With ACO - Scene #: 199 | Packets sent: 19801 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 40.1679 / 39.703 / 88.1328 / 40.6128 / 40.4526

With ACO - Scene #: 200 | Packets sent: 19901 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 38.7702 / 38.3053 / 86.735 / 39.2151 / 39.0548

With ACO - Scene #: 201 | Packets sent: 20001 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 37.3724 / 36.9075 / 85.3372 / 37.8173 / 37.657

With ACO - Scene #: 202 | Packets sent: 20101 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 35.9747 / 35.5097 / 83.9395 / 36.4195 / 36.2593

With ACO - Scene #: 203 | Packets sent: 20201 | Best route: 1 | Routing nodes: 4 16 21 26 52 | Nodes energy: 34.5769 / 34.112 / 82.5417 / 35.0218 / 34.8615

With ACO - Scene #: 204 | Packets sent: 20301 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 33.1791 / 32.7142 / 81.144 / 33.624 / 33.4638

With ACO - Scene #: 205 | Packets sent: 20401 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 31.7814 / 31.3165 / 79.7462 / 32.2263 / 32.066

With ACO - Scene #: 206 | Packets sent: 20501 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 30.3836 / 29.9187 / 78.3485 / 30.8285 / 30.6683

With ACO - Scene #: 207 | Packets sent: 20601 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 28.9859 / 28.521 / 76.9507 / 29.4307 / 29.2705

With ACO - Scene #: 208 | Packets sent: 20701 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 27.5881 / 27.1232 / 75.5529 / 28.033 / 27.8727

With ACO - Scene #: 209 | Packets sent: 20801 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 26.1904 / 25.7254 / 74.1552 / 26.6352 / 26.475

With ACO - Scene #: 210 | Packets sent: 20901 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 24.3277 / 72.7574 / 90.1573 / 90.2265 / 90.216

With ACO - Scene #: 211 | Packets sent: 21001 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 22.9299 / 71.3597 / 88.7596 / 88.8288 / 88.8182

With ACO - Scene #: 212 | Packets sent: 21101 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 90.825 / 90.7373 / 90.2844 / 90.7671 / 90.7379

With ACO - Scene #: 213 | Packets sent: 21201 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 89.4273 / 89.3396 / 88.8866 / 89.3693 / 89.3402

With ACO - Scene #: 214 | Packets sent: 21301 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 88.0295 / 87.9418 / 87.4889 / 87.9716 / 87.9424

With ACO - Scene #: 215 | Packets sent: 21401 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 86.6318 / 86.5441 / 86.0911 / 86.5738 / 86.5447

With ACO - Scene #: 216 | Packets sent: 21501 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 85.234 / 85.1463 / 84.6933 / 85.1761 / 85.1469

With ACO - Scene #: 217 | Packets sent: 21601 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 83.8363 / 83.7485 / 83.2956 / 83.7783 / 83.7491

With ACO - Scene #: 218 | Packets sent: 21701 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 82.4385 / 82.3508 / 81.8978 / 82.3805 / 82.3514

With ACO - Scene #: 219 | Packets sent: 21801 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 81.0407 / 80.953 / 80.5001 / 80.9828 / 80.9536

With ACO - Scene #: 220 | Packets sent: 21901 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 79.643 / 79.5553 / 79.1023 / 79.585 / 79.5559

With ACO - Scene #: 221 | Packets sent: 22001 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 78.2452 / 78.1575 / 77.7045 / 78.1873 / 78.1581

With ACO - Scene #: 222 | Packets sent: 22101 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 76.8475 / 90.5213 / 76.3068 / 90.7875 / 90.9739 / 90.1241

With ACO - Scene #: 223 | Packets sent: 22201 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 75.4497 / 89.1235 / 74.909 / 89.3898 / 89.5761 / 88.7264

With ACO - Scene #: 224 | Packets sent: 22301 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 74.052 / 87.7258 / 73.5113 / 87.992 / 88.1783 / 87.3286

With ACO - Scene #: 225 | Packets sent: 22401 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 72.6542 / 86.328 / 72.1135 / 86.5942 / 86.7806 / 85.9308

With ACO - Scene #: 226 | Packets sent: 22501 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 84.9303 / 90.3185 / 90.3877 / 85.3828 / 90.3771

With ACO - Scene #: 227 | Packets sent: 22601 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 83.5325 / 88.9207 / 88.9899 / 83.9851 / 88.9794

With ACO - Scene #: 228 | Packets sent: 22701 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 82.1347 / 87.523 / 87.5922 / 82.5873 / 87.5816

With ACO - Scene #: 229 | Packets sent: 22801 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 80.737 / 86.1252 / 86.1944 / 81.1896 / 86.1839

With ACO - Scene #: 230 | Packets sent: 22901 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 79.3392 / 84.7274 / 84.7967 / 79.7918 / 84.7861

With ACO - Scene #: 231 | Packets sent: 23001 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 77.9415 / 83.3297 / 83.3989 / 78.394 / 83.3883

With ACO - Scene #: 232 | Packets sent: 23101 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 76.5437 / 81.9319 / 82.0012 / 76.9963 / 81.9906

With ACO - Scene #: 233 | Packets sent: 23201 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 75.146 / 80.5342 / 80.6034 / 75.5985 / 80.5928

With ACO - Scene #: 234 | Packets sent: 23301 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 73.7482 / 79.1364 / 79.2056 / 74.2008 / 79.1951

With ACO - Scene #: 235 | Packets sent: 23401 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 72.3504 / 77.7386 / 77.8079 / 72.803 / 77.7973

With ACO - Scene #: 236 | Packets sent: 23501 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 70.9527 / 76.3409 / 76.4101 / 71.4053 / 76.3995

With ACO - Scene #: 237 | Packets sent: 23601 | Best route: 8 | Routing nodes: 16 34 37 48
54 | Nodes energy: 69.5549 / 74.9431 / 75.0124 / 70.0075 / 75.0018

With ACO - Scene #: 238 | Packets sent: 23701 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 68.1572 / 73.5454 / 73.6146 / 68.6097 / 73.604

With ACO - Scene #: 239 | Packets sent: 23801 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 66.7594 / 72.1476 / 72.2168 / 67.212 / 72.2063

With ACO - Scene #: 240 | Packets sent: 23901 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 65.3616 / 70.7499 / 70.8191 / 65.8142 / 70.8085

With ACO - Scene #: 241 | Packets sent: 24001 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 63.9639 / 69.3521 / 69.4213 / 64.4165 / 69.4108

With ACO - Scene #: 242 | Packets sent: 24101 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 62.5661 / 67.9543 / 68.0236 / 63.0187 / 68.013

With ACO - Scene #: 243 | Packets sent: 24201 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 61.1684 / 66.5566 / 66.6258 / 61.6209 / 66.6152

With ACO - Scene #: 244 | Packets sent: 24301 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 59.7706 / 65.1588 / 65.2281 / 60.2232 / 65.2175

With ACO - Scene #: 245 | Packets sent: 24401 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 58.3729 / 63.7611 / 63.8303 / 58.8254 / 63.8197

With ACO - Scene #: 246 | Packets sent: 24501 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 56.9751 / 62.3633 / 62.4325 / 57.4277 / 62.422

With ACO - Scene #: 247 | Packets sent: 24601 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 55.5773 / 60.9656 / 61.0348 / 56.0299 / 61.0242

With ACO - Scene #: 248 | Packets sent: 24701 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 54.1796 / 59.5678 / 59.637 / 54.6322 / 59.6265

With ACO - Scene #: 249 | Packets sent: 24801 | Best route: 8 | Routing nodes: 16 34 37 48 54 | Nodes energy: 52.7818 / 58.17 / 58.2393 / 53.2344 / 58.2287

With ACO - Scene #: 250 | Packets sent: 24901 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 90.9676 / 90.5576 / 90.4269 / 90.1243 / 90.5873 / 90.5582

With ACO - Scene #: 251 | Packets sent: 25001 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 89.5699 / 89.1598 / 89.0292 / 88.7265 / 89.1896 / 89.1604

With ACO - Scene #: 252 | Packets sent: 25101 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 88.1721 / 87.7621 / 87.6314 / 87.3288 / 87.7918 / 87.7627

With ACO - Scene #: 253 | Packets sent: 25201 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 86.7743 / 86.3643 / 86.2337 / 85.931 / 86.3941 / 86.3649

With ACO - Scene #: 254 | Packets sent: 25301 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 85.3766 / 84.9665 / 84.8359 / 84.5333 / 84.9963 / 84.9672

With ACO - Scene #: 255 | Packets sent: 25401 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 83.9788 / 83.5688 / 83.4381 / 83.1355 / 83.5985 / 83.5694

With ACO - Scene #: 256 | Packets sent: 25501 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 82.5811 / 82.171 / 82.0404 / 81.7377 / 82.2008 / 82.1716

With ACO - Scene #: 257 | Packets sent: 25601 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 81.1833 / 80.7733 / 80.6426 / 80.34 / 80.803 / 80.7739

With ACO - Scene #: 258 | Packets sent: 25701 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 79.7856 / 79.3755 / 79.2449 / 78.9422 / 79.4053 / 79.3761

With ACO - Scene #: 259 | Packets sent: 25801 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 78.3878 / 77.9778 / 77.8471 / 77.5445 / 78.0075 / 77.9784

With ACO - Scene #: 260 | Packets sent: 25901 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 76.99 / 76.58 / 76.4494 / 76.1467 / 76.6098 / 76.5806

With ACO - Scene #: 261 | Packets sent: 26001 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 75.5923 / 75.1822 / 75.0516 / 74.749 / 75.212 / 75.1828

With ACO - Scene #: 262 | Packets sent: 26101 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 74.1945 / 73.7845 / 73.6538 / 73.3512 / 73.8142 / 73.7851

With ACO - Scene #: 263 | Packets sent: 26201 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 72.7968 / 72.3867 / 72.2561 / 71.9534 / 72.4165 / 72.3873

With ACO - Scene #: 264 | Packets sent: 26301 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 71.399 / 70.989 / 70.8583 / 70.5557 / 71.0187 / 70.9896

With ACO - Scene #: 265 | Packets sent: 26401 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 70.0013 / 69.5912 / 69.4606 / 69.1579 / 69.621 / 69.5918

With ACO - Scene #: 266 | Packets sent: 26501 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 68.6035 / 68.1934 / 68.0628 / 67.7602 / 68.2232 / 68.1941

With ACO - Scene #: 267 | Packets sent: 26601 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 67.2057 / 66.7957 / 66.6651 / 66.3624 / 66.8255 / 66.7963

With ACO - Scene #: 268 | Packets sent: 26701 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 65.808 / 65.3979 / 65.2673 / 64.9646 / 65.4277 / 65.3985

With ACO - Scene #: 269 | Packets sent: 26801 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 64.4102 / 64.0002 / 63.8695 / 63.5669 / 64.0299 / 64.0008

With ACO - Scene #: 270 | Packets sent: 26901 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 63.0125 / 62.6024 / 62.4718 / 62.1691 / 62.6322 / 62.603

With ACO - Scene #: 271 | Packets sent: 27001 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 61.6147 / 61.2047 / 61.074 / 60.7714 / 61.2344 / 61.2053

With ACO - Scene #: 272 | Packets sent: 27101 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 60.2169 / 59.8069 / 59.6763 / 59.3736 / 59.8367 / 59.8075

With ACO - Scene #: 273 | Packets sent: 27201 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 58.8192 / 58.4091 / 58.2785 / 57.9759 / 58.4389 / 58.4098

With ACO - Scene #: 274 | Packets sent: 27301 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 57.4214 / 57.0114 / 56.8807 / 56.5781 / 57.0412 / 57.012

With ACO - Scene #: 275 | Packets sent: 27401 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 56.0237 / 55.6136 / 55.483 / 55.1803 / 55.6434 / 55.6142

With ACO - Scene #: 276 | Packets sent: 27501 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 54.6259 / 54.2159 / 54.0852 / 53.7826 / 54.2456 / 54.2165

With ACO - Scene #: 277 | Packets sent: 27601 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 53.2282 / 52.8181 / 52.6875 / 52.3848 / 52.8479 / 52.8187

With ACO - Scene #: 278 | Packets sent: 27701 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 51.8304 / 51.4204 / 51.2897 / 50.9871 / 51.4501 / 51.421

With ACO - Scene #: 279 | Packets sent: 27801 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 50.4326 / 50.0226 / 49.892 / 49.5893 / 50.0524 / 50.0232

With ACO - Scene #: 280 | Packets sent: 27901 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 49.0349 / 48.6248 / 48.4942 / 48.1916 / 48.6546 / 48.6254

With ACO - Scene #: 281 | Packets sent: 28001 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 47.6371 / 47.2271 / 47.0964 / 46.7938 / 47.2568 / 47.2277

With ACO - Scene #: 282 | Packets sent: 28101 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 46.2394 / 45.8293 / 45.6987 / 45.396 / 45.8591 / 45.8299

With ACO - Scene #: 283 | Packets sent: 28201 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 44.8416 / 44.4316 / 44.3009 / 43.9983 / 44.4613 / 44.4322

With ACO - Scene #: 284 | Packets sent: 28301 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 43.4439 / 43.0338 / 42.9032 / 42.6005 / 43.0636 / 43.0344

With ACO - Scene #: 285 | Packets sent: 28401 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 42.0461 / 41.636 / 41.5054 / 41.2028 / 41.6658 / 41.6367

With ACO - Scene #: 286 | Packets sent: 28501 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 40.6483 / 40.2383 / 40.1077 / 39.805 / 40.2681 / 40.2389

With ACO - Scene #: 287 | Packets sent: 28601 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 39.2506 / 38.8405 / 38.7099 / 38.4072 / 38.8703 / 38.8411

With ACO - Scene #: 288 | Packets sent: 28701 | Best route: 6 | Routing nodes: 4 12 26 30 53 58 | Nodes energy: 37.8528 / 37.4428 / 37.3121 / 37.0095 / 37.4725 / 37.4434

With ACO - Scene #: 289 | Packets sent: 28801 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 90.2813 / 90.3505 / 36.0748 / 90.4296 / 90.1026

With ACO - Scene #: 290 | Packets sent: 28901 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 88.8835 / 88.9528 / 34.677 / 89.0318 / 88.7049

With ACO - Scene #: 291 | Packets sent: 29001 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 87.4858 / 87.555 / 33.2793 / 87.6341 / 87.3071

With ACO - Scene #: 292 | Packets sent: 29101 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 86.088 / 86.1572 / 31.8815 / 86.2363 / 85.9094

With ACO - Scene #: 293 | Packets sent: 29201 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 84.6903 / 84.7595 / 30.4838 / 84.8386 / 84.5116

With ACO - Scene #: 294 | Packets sent: 29301 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 83.2925 / 83.3617 / 29.086 / 83.4408 / 83.1138

With ACO - Scene #: 295 | Packets sent: 29401 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 81.8947 / 81.964 / 27.6882 / 82.043 / 81.7161

With ACO - Scene #: 296 | Packets sent: 29501 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 80.497 / 80.5662 / 26.2905 / 80.6453 / 80.3183

With ACO - Scene #: 297 | Packets sent: 29601 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 79.0992 / 79.1685 / 24.8927 / 79.2475 / 78.9206

With ACO - Scene #: 298 | Packets sent: 29701 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 77.7015 / 77.7707 / 23.495 / 77.8498 / 77.5228

With ACO - Scene #: 299 | Packets sent: 29801 | Best route: 9 | Routing nodes: 34 37 53 62 65 | Nodes energy: 76.3037 / 76.3729 / 22.0972 / 76.452 / 76.125

With ACO - Scene #: 300 | Packets sent: 29901 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 90.6608 / 90.8064 / 90.1201 / 90.2734 / 90.4464

With ACO - Scene #: 301 | Packets sent: 30001 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 89.263 / 89.4087 / 88.7224 / 88.8757 / 89.0487

With ACO - Scene #: 302 | Packets sent: 30101 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 87.8653 / 88.0109 / 87.3246 / 87.4779 / 87.6509

With ACO - Scene #: 303 | Packets sent: 30201 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 86.4675 / 86.6132 / 85.9269 / 86.0802 / 86.2532

With ACO - Scene #: 304 | Packets sent: 30301 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 85.0698 / 85.2154 / 84.5291 / 84.6824 / 84.8554

With ACO - Scene #: 305 | Packets sent: 30401 | Best route: 2 | Routing nodes: 4 16 26 48 52 | Nodes energy: 83.672 / 83.8176 / 83.1313 / 83.2847 / 83.4576

With ACO - Scene #: 306 | Packets sent: 30501 | Best route: 2 | Routing nodes: 4 16 26 48
52 | Nodes energy: 82.2743 / 82.4199 / 81.7336 / 81.8869 / 82.0599

With ACO - Scene #: 307 | Packets sent: 30601 | Best route: 2 | Routing nodes: 4 16 26 48
52 | Nodes energy: 80.8765 / 81.0221 / 80.3358 / 80.4891 / 80.6621

With ACO - Scene #: 308 | Packets sent: 30701 | Best route: 2 | Routing nodes: 4 16 26 48
52 | Nodes energy: 79.4787 / 79.6244 / 78.9381 / 79.0914 / 79.2644

With ACO - Scene #: 309 | Packets sent: 30801 | Best route: 2 | Routing nodes: 4 16 26 48
52 | Nodes energy: 78.081 / 78.2266 / 77.5403 / 77.6936 / 77.8666

With ACO - Scene #: 310 | Packets sent: 30901 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 76.8289 / 90.6587 / 90.6749 / 90.7441 / 90.3399

With ACO - Scene #: 311 | Packets sent: 31001 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 75.4311 / 89.2609 / 89.2771 / 89.3464 / 88.9422

With ACO - Scene #: 312 | Packets sent: 31101 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 74.0333 / 87.8631 / 87.8794 / 87.9486 / 87.5444

With ACO - Scene #: 313 | Packets sent: 31201 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 72.6356 / 86.4654 / 86.4816 / 86.5509 / 86.1467

With ACO - Scene #: 314 | Packets sent: 31301 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 71.2378 / 85.0676 / 85.0839 / 85.1531 / 84.7489

With ACO - Scene #: 315 | Packets sent: 31401 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 69.8401 / 83.6699 / 83.6861 / 83.7553 / 83.3512

With ACO - Scene #: 316 | Packets sent: 31501 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 68.4423 / 82.2721 / 82.2883 / 82.3576 / 81.9534

With ACO - Scene #: 317 | Packets sent: 31601 | Best route: 7 | Routing nodes: 16 21 34 37
54 | Nodes energy: 67.0445 / 80.8743 / 80.8906 / 80.9598 / 80.5556

With ACO - Scene #: 318 | Packets sent: 31701 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 90.4811 / 65.6468 / 79.4766 / 90.9259 / 90.2667

With ACO - Scene #: 319 | Packets sent: 31801 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 89.0833 / 64.249 / 78.0788 / 89.5282 / 88.8689

With ACO - Scene #: 320 | Packets sent: 31901 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 87.6855 / 62.8513 / 76.6811 / 88.1304 / 87.4712

With ACO - Scene #: 321 | Packets sent: 32001 | Best route: 1 | Routing nodes: 4 16 21 26
52 | Nodes energy: 86.2878 / 61.4535 / 75.2833 / 86.7327 / 86.0734

With ACO - Scene #: 322 | Packets sent: 32101 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 84.89 / 60.0558 / 73.8856 / 85.3349 / 90.803 / 84.6757

With ACO - Scene #: 323 | Packets sent: 32201 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 83.4923 / 58.658 / 72.4878 / 83.9371 / 89.4053 / 83.2779

With ACO - Scene #: 324 | Packets sent: 32301 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 82.0945 / 57.2602 / 71.09 / 82.5394 / 88.0075 / 81.8801

With ACO - Scene #: 325 | Packets sent: 32401 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 80.6968 / 55.8625 / 69.6923 / 81.1416 / 86.6098 / 80.4824

With ACO - Scene #: 326 | Packets sent: 32501 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 79.299 / 54.4647 / 68.2945 / 79.7439 / 85.212 / 79.0846

With ACO - Scene #: 327 | Packets sent: 32601 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 77.9012 / 53.067 / 66.8968 / 78.3461 / 83.8142 / 77.6869

With ACO - Scene #: 328 | Packets sent: 32701 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 76.5035 / 51.6692 / 65.499 / 76.9484 / 82.4165 / 76.2891

With ACO - Scene #: 329 | Packets sent: 32801 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 75.1057 / 50.2715 / 64.1013 / 75.5506 / 81.0187 / 74.8913

With ACO - Scene #: 330 | Packets sent: 32901 | Best route: 4 | Routing nodes: 4 16 21 26
30 52 | Nodes energy: 73.708 / 48.8737 / 62.7035 / 74.1528 / 79.621 / 73.4936

With ACO - Scene #: 331 | Packets sent: 33001 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 72.3102 / 47.4759 / 72.7551 / 78.2232 / 90.0937 / 72.0958

With ACO - Scene #: 332 | Packets sent: 33101 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 70.9124 / 46.0782 / 71.3573 / 76.8255 / 88.6959 / 70.6981

With ACO - Scene #: 333 | Packets sent: 33201 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 69.5147 / 44.6804 / 69.9596 / 75.4277 / 87.2982 / 69.3003

With ACO - Scene #: 334 | Packets sent: 33301 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 68.1169 / 43.2827 / 68.5618 / 74.0299 / 85.9004 / 67.9026

With ACO - Scene #: 335 | Packets sent: 33401 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 66.7192 / 41.8849 / 67.164 / 72.6322 / 84.5027 / 66.5048

With ACO - Scene #: 336 | Packets sent: 33501 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 65.3214 / 40.4871 / 65.7663 / 71.2344 / 83.1049 / 65.107

With ACO - Scene #: 337 | Packets sent: 33601 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 63.9237 / 39.0894 / 64.3685 / 69.8367 / 81.7071 / 63.7093

With ACO - Scene #: 338 | Packets sent: 33701 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 62.5259 / 37.6916 / 62.9708 / 68.4389 / 80.3094 / 62.3115

With ACO - Scene #: 339 | Packets sent: 33801 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 61.1281 / 36.2939 / 61.573 / 67.0412 / 78.9116 / 60.9138

With ACO - Scene #: 340 | Packets sent: 33901 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 59.7304 / 34.8961 / 60.1753 / 65.6434 / 77.5139 / 59.516

With ACO - Scene #: 341 | Packets sent: 34001 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 58.3326 / 33.4984 / 58.7775 / 64.2456 / 76.1161 / 58.1183

With ACO - Scene #: 342 | Packets sent: 34101 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 56.9349 / 32.1006 / 57.3797 / 62.8479 / 74.7184 / 56.7205

With ACO - Scene #: 343 | Packets sent: 34201 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 55.5371 / 30.7028 / 55.982 / 61.4501 / 73.3206 / 55.3227

With ACO - Scene #: 344 | Packets sent: 34301 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 54.1394 / 29.3051 / 54.5842 / 60.0524 / 71.9228 / 53.925

With ACO - Scene #: 345 | Packets sent: 34401 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 52.7416 / 27.9073 / 53.1865 / 58.6546 / 70.5251 / 52.5272

With ACO - Scene #: 346 | Packets sent: 34501 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 51.3438 / 26.5096 / 51.7887 / 57.2568 / 69.1273 / 51.1295

With ACO - Scene #: 347 | Packets sent: 34601 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 49.9461 / 25.1118 / 50.391 / 55.8591 / 67.7296 / 49.7317

With ACO - Scene #: 348 | Packets sent: 34701 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 48.5483 / 23.7141 / 48.9932 / 54.4613 / 66.3318 / 48.3339

With ACO - Scene #: 349 | Packets sent: 34801 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 47.1506 / 22.3163 / 47.5954 / 53.0636 / 64.9341 / 46.9362

With ACO - Scene #: 350 | Packets sent: 34901 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 45.7528 / 20.9185 / 46.1977 / 51.6658 / 63.5363 / 45.5384

With ACO - Scene #: 351 | Packets sent: 35001 | Best route: 5 | Routing nodes: 4 16 26 30
48 52 | Nodes energy: 44.355 / 19.5208 / 44.7999 / 50.2681 / 62.1385 / 44.1407

With ACO - Scene #: 352 | Packets sent: 35101 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 42.9573 / 90.2508 / 43.4022 / 90.6741 / 90.2514

With ACO - Scene #: 353 | Packets sent: 35201 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 41.5595 / 88.853 / 42.0044 / 89.2764 / 88.8536

With ACO - Scene #: 354 | Packets sent: 35301 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 40.1618 / 87.4552 / 40.6067 / 87.8786 / 87.4559

With ACO - Scene #: 355 | Packets sent: 35401 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 38.764 / 86.0575 / 39.2089 / 86.4809 / 86.0581

With ACO - Scene #: 356 | Packets sent: 35501 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 37.3663 / 84.6597 / 37.8111 / 85.0831 / 84.6603

With ACO - Scene #: 357 | Packets sent: 35601 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 35.9685 / 83.262 / 36.4134 / 83.6853 / 83.2626

With ACO - Scene #: 358 | Packets sent: 35701 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 34.5707 / 81.8642 / 35.0156 / 82.2876 / 81.8648

With ACO - Scene #: 359 | Packets sent: 35801 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 33.173 / 80.4665 / 33.6179 / 80.8898 / 80.4671

With ACO - Scene #: 360 | Packets sent: 35901 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 31.7752 / 79.0687 / 32.2201 / 79.4921 / 79.0693

With ACO - Scene #: 361 | Packets sent: 36001 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 30.3775 / 77.6709 / 30.8223 / 78.0943 / 77.6716

With ACO - Scene #: 362 | Packets sent: 36101 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 28.9797 / 76.2732 / 29.4246 / 76.6966 / 76.2738

With ACO - Scene #: 363 | Packets sent: 36201 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 27.582 / 74.8754 / 28.0268 / 75.2988 / 74.876

With ACO - Scene #: 364 | Packets sent: 36301 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 26.1842 / 73.4777 / 26.6291 / 73.901 / 73.4783

With ACO - Scene #: 365 | Packets sent: 36401 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 24.7864 / 72.0799 / 25.2313 / 72.5033 / 72.0805

With ACO - Scene #: 366 | Packets sent: 36501 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 23.3887 / 70.6822 / 23.8336 / 71.1055 / 70.6828

With ACO - Scene #: 367 | Packets sent: 36601 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 21.9909 / 69.2844 / 22.4358 / 69.7078 / 69.285

With ACO - Scene #: 368 | Packets sent: 36701 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 20.5932 / 67.8866 / 21.038 / 68.31 / 67.8872

With ACO - Scene #: 369 | Packets sent: 36801 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 19.1954 / 66.4889 / 19.6403 / 66.9123 / 66.4895

With ACO - Scene #: 370 | Packets sent: 36901 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 17.7977 / 65.0911 / 18.2425 / 65.5145 / 65.0917

With ACO - Scene #: 371 | Packets sent: 37001 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 16.3999 / 63.6934 / 16.8448 / 64.1167 / 63.694

With ACO - Scene #: 372 | Packets sent: 37101 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 15.0021 / 62.2956 / 15.447 / 62.719 / 62.2962

With ACO - Scene #: 373 | Packets sent: 37201 | Best route: 3 | Routing nodes: 4 12 26 53
58 | Nodes energy: 13.6044 / 60.8978 / 14.0493 / 61.3212 / 60.8985

With ACO - Scene #: 374 | Packets sent: 37301 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 12.2066 / 59.5001 / 12.6515 / 59.9235 / 59.5007

With ACO - Scene #: 375 | Packets sent: 37401 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 10.8089 / 58.1023 / 11.2537 / 58.5257 / 58.1029

With ACO - Scene #: 376 | Packets sent: 37501 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 9.41111 / 56.7046 / 9.85598 / 57.1279 / 56.7052

With ACO - Scene #: 377 | Packets sent: 37601 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 8.01335 / 55.3068 / 8.45822 / 55.7302 / 55.3074

With ACO - Scene #: 378 | Packets sent: 37701 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 6.61559 / 53.9091 / 7.06047 / 54.3324 / 53.9097

With ACO - Scene #: 379 | Packets sent: 37801 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 5.21783 / 52.5113 / 5.66271 / 52.9347 / 52.5119

With ACO - Scene #: 380 | Packets sent: 37901 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 3.82007 / 51.1135 / 4.26495 / 51.5369 / 51.1142

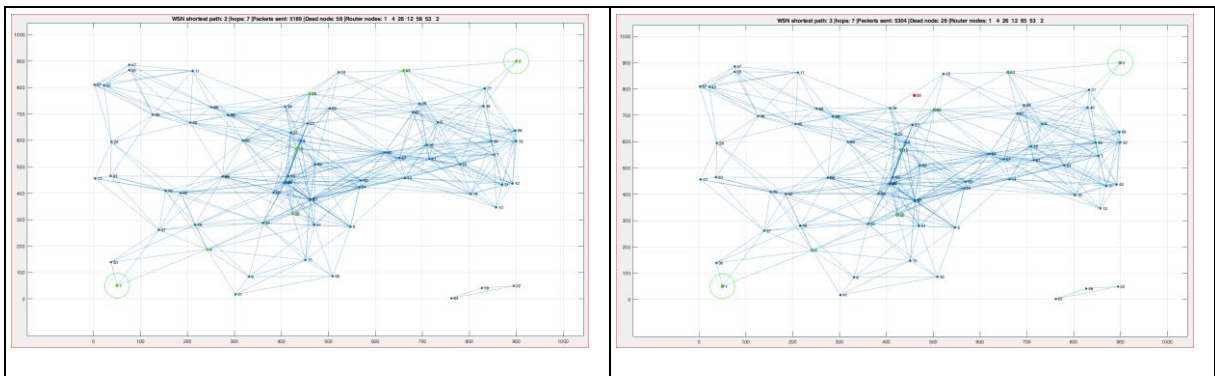
With ACO - Scene #: 381 | Packets sent: 38001 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 2.42232 / 49.7158 / 2.86719 / 50.1392 / 49.7164

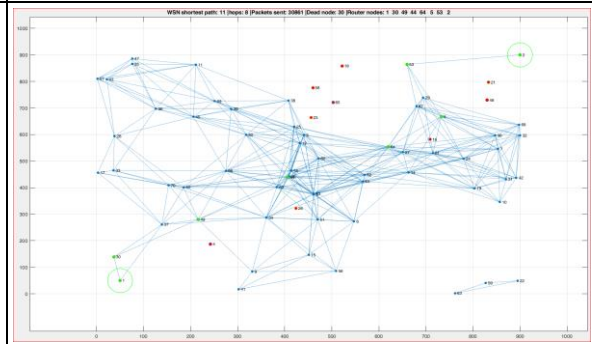
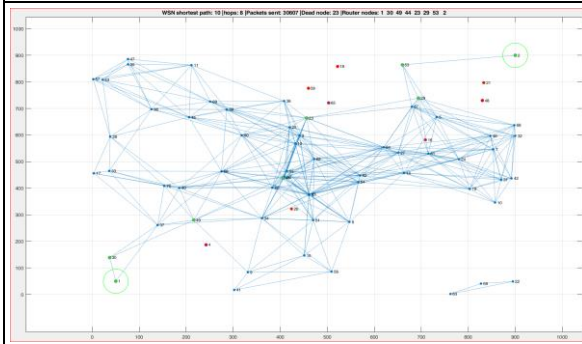
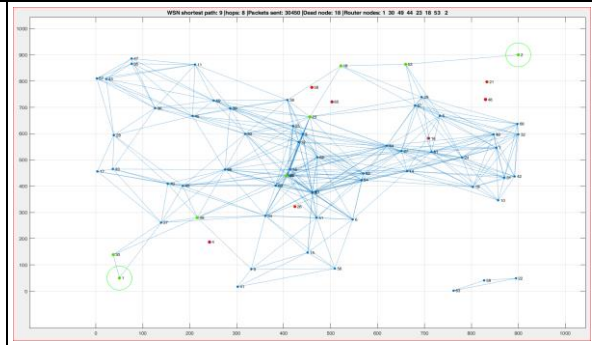
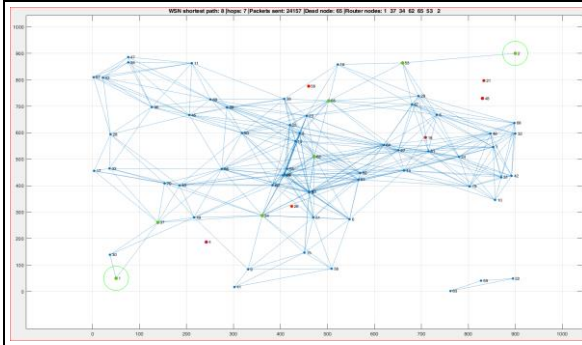
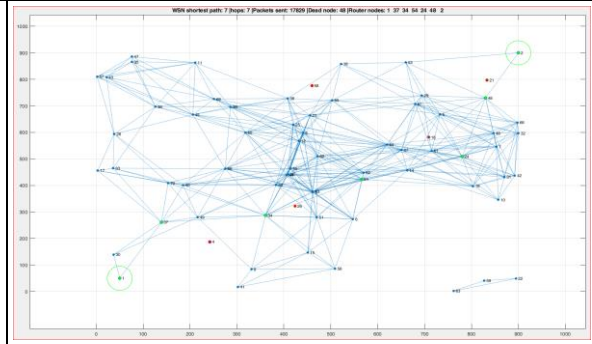
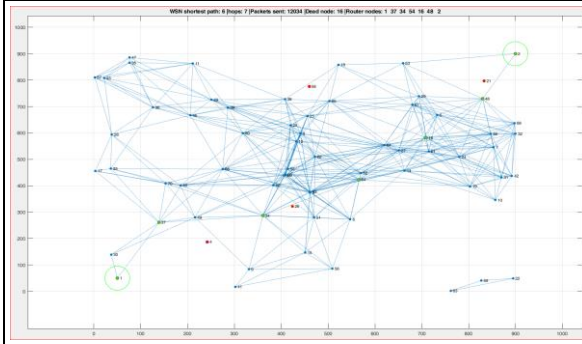
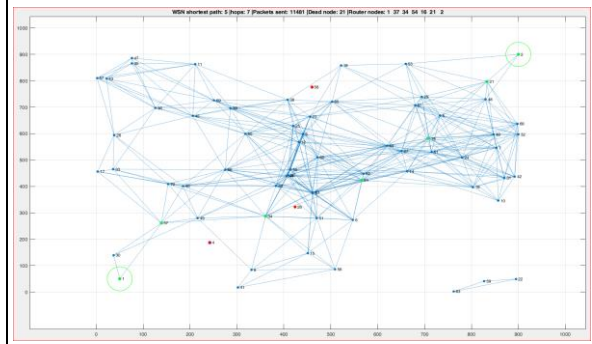
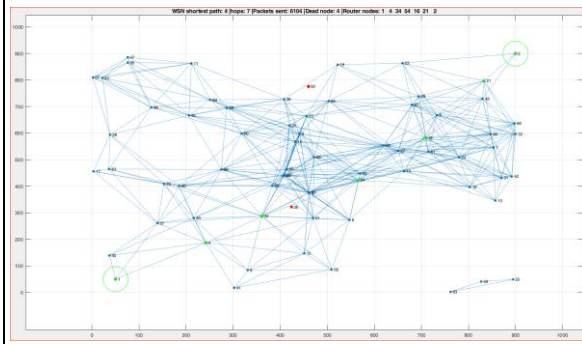
With ACO - Scene #: 382 | Packets sent: 38101 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: 1.02456 / 48.318 / 1.46943 / 48.7414 / 48.3186

With ACO - Scene #: 383 | Packets sent: 38201 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: -0.373199 / 46.9203 0.0716772 / 47.3436 / 46.9209

With ACO - Scene #: 384 | Packets sent: 38301 | Best route: 3 | Routing nodes: 4 12 26 53 58 | Nodes energy: -1.77096 / 45.5225 -1.32608 / 45.9459 / 45.5231

Iniciadas as rodadas de transmissão de pacotes com respectivo consumo energético, a sequência de imagens abaixo representa a série temporal dos acontecimentos com seus respectivos dados.





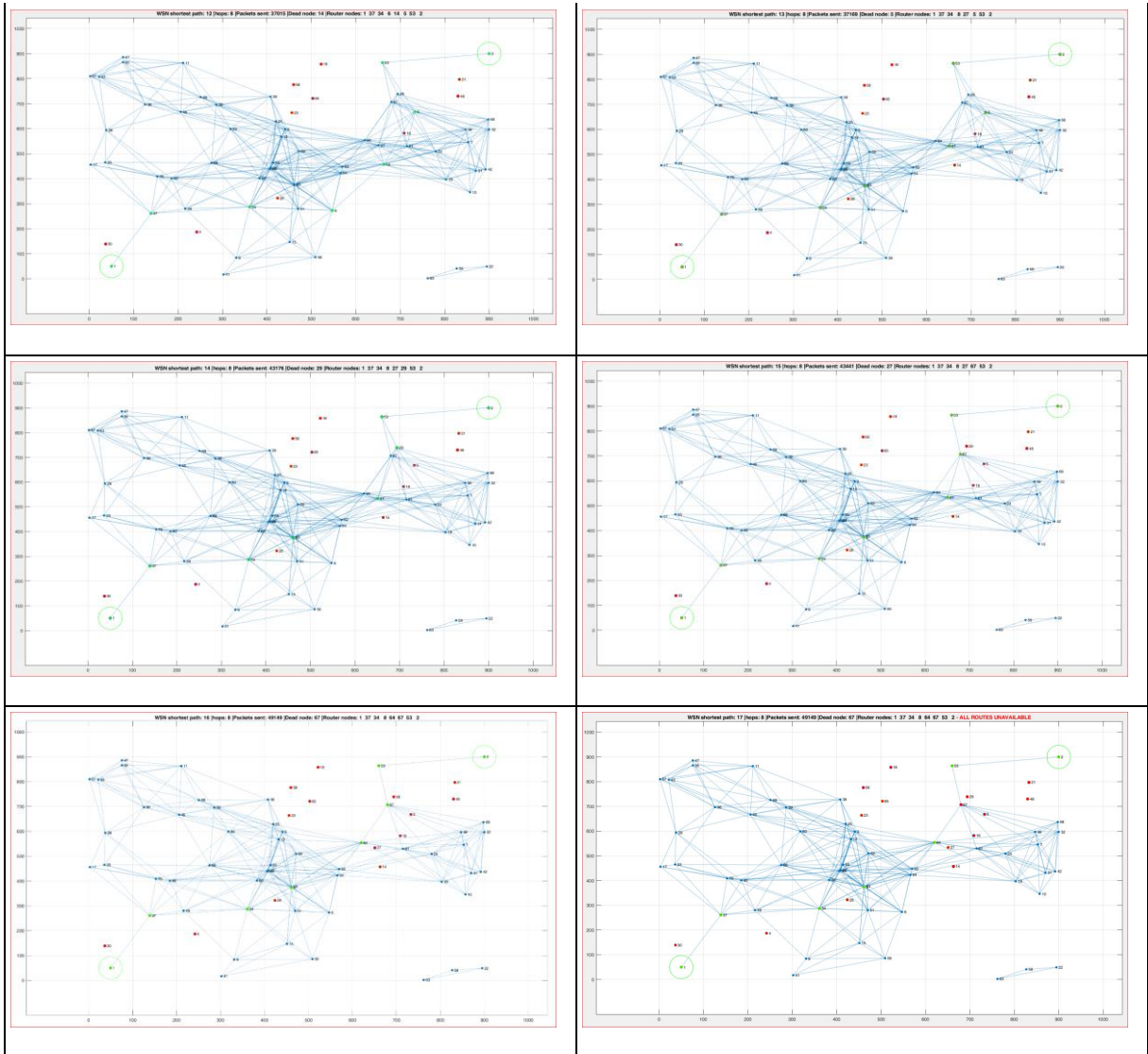


Tabela 1 - Caracterização da Indisponibilidade dos nós da rede IoT simulada

Apêndice C

RESULTADOS ENERGÉTICOS DA SIMULAÇÃO

Tabelas comparativas do total energético após a transmissão de 38.201 pacotes para indisponibilidade de apenas um nó na rede com ACO e 49.149 pacotes para toda rede sem ACO.

Tabela 4 - Totais energéticos

NodeID	Cenário sem ACO	Cenário com ACO
3	943.544.033.073.140	943.544.033.073.140
4	NaN	NaN
5	NaN	909.375.527.032.857
6	246.002.192.494.065	905.991.165.304.299
7	958.863.153.409.666	958.863.153.409.666
8	196.586.460.525.372	903.814.052.940.495
9	934.977.496.144.265	934.977.496.144.265
10	215.228.583.002.610	906.669.699.929.251
11	978.878.265.845.770	978.878.265.845.770
12	881.574.501.666.908	439.430.455.542.070
13	988.339.425.431.947	988.339.425.431.947
14	NaN	959.656.139.334.242
15	970.757.480.597.961	970.757.480.597.961
16	NaN	900.688.279.907.305
17	982.814.071.417.633	982.814.071.417.633
18	NaN	903.223.007.872.274
19	914.101.429.308.342	914.101.429.308.342
20	937.539.067.524.354	937.539.067.524.354
21	NaN	908.725.068.929.847
22	990.174.706.125.573	990.174.706.125.573
23	NaN	987.838.566.640.242
24	632.108.313.826.936	901.115.544.693.468
25	913.364.327.472.703	903.250.409.623.975
26	NaN	-290.554.682.377.153
27	NaN	900.346.334.583.995
28	103.764.730.961.152	900.840.885.983.967
29	NaN	963.661.405.669.572
30	NaN	902.637.766.770.979
31	906.527.368.565.019	906.527.368.565.019
32	962.455.745.966.401	962.455.745.966.401
33	520.064.228.326.786	907.845.737.192.496
34	461.344.189.719.572	903.339.805.391.005
35	995.645.869.328.248	995.645.869.328.248

36	938.133.215.671.652	938.133.215.671.652
37	462.448.618.912.023	904.032.099.508.527
38	520.217.419.530.152	906.623.505.052.114
39	935.103.542.964.977	935.103.542.964.977
40	905.795.993.046.331	905.795.993.046.331
41	926.072.471.234.632	926.072.471.234.632
42	958.624.222.941.759	958.624.222.941.759
43	980.874.650.122.397	980.874.650.122.397
44	508.800.664.793.347	902.739.460.640.452
45	907.731.100.271.795	907.731.100.271.795
46	519.978.562.957.498	900.666.878.636.137
47	518.907.037.848.953	906.073.155.733.120
48	NaN	903.075.451.273.417
49	510.142.642.282.196	903.836.739.315.883
50	520.512.182.250.374	905.382.892.594.171
51	520.225.349.389.397	909.783.027.396.059
52	934.945.325.187.508	900.497.461.631.320
53	404.979.029.662.685	443.664.206.968.771
54	500.917.555.160.631	909.845.889.177.479
55	518.668.857.657.804	909.712.267.056.091
56	520.456.928.710.231	902.011.824.659.018
57	991.754.349.161.547	991.754.349.161.547
58	NaN	439.436.558.683.568
59	922.418.760.786.812	922.418.760.786.812
60	519.365.350.394.880	902.258.519.061.812
61	966.879.827.163.279	966.879.827.163.279
62	509.968.800.774.354	908.232.009.480.104
63	520.004.178.671.299	902.908.890.659.174
64	517.836.504.927.944	964.678.768.143.343
65	NaN	904.962.336.013.045
66	520.199.985.119.181	904.538.574.555.941
67	NaN	914.159.526.341.100
68	959.988.187.580.422	959.988.187.580.422
69	520.530.535.057.311	904.635.959.880.045
70	926.132.805.663.391	926.132.805.663.391
Totais	38.023.270.296.631.200	59.455.778.412.773.200