UNIVERSIDADE FEDERAL DO RIO GRANDE
CENTRO DE CIÊNCIAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL
CURSO DE MESTRADO EM MODELAGEM COMPUTACIONAL

Dissertação de Mestrado

Uma Abordagem Evolucionária para o Planejamento Online de Rotas para Múltiplos Robôs Móveis em Ambiente Multiagente

Tauã Milech Cabreira

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Modelagem Computacional da Universidade Federal do Rio Grande, como requisito parcial para a obtenção do grau de Mestre em Modelagem Computacional

Orientador: Prof. Dr. Marilton Sanchotene de Aguiar
Co-orientadora: Prof[a]. Dr[a]. Graçaliz Pereira Dimuro

Rio Grande, 2013

TAUÃ MILECH CABREIRA

"UMA ABORDAGEM EVOLUCIONÁRIA PARA O PLANEJAMENTO ONLINE DE ROTAS PARA MÚLTIPLOS ROBÔS MÓVEIS EM AMBIENTE MULTIAGENTE"

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional da Universidade Federal do Rio Grande -FURG, como requisito parcial para obtenção do Grau de Mestre. Área concentração: Modelagem Computacional.

Aprovada em

BANCA EXAMINADORA

Prof. Dr. Marilton Sanchotene de Aguiar
Orientador - FURG

Profª. Drª Graçaliz Pereira Dimuro
Co-orientadora-FURG

Prof. Dr Leonardo Ramos Emmendorfer
FURG

Profª. Drª Cátia Maria dos Santos Machado
FURG

Prof. Dr. Paulo Roberto Ferreira Júnior
UFPEL

Rio Grande -RS
2013

C117a Cabreira, Tauã Milech.
Uma abordagem evolucionária para o planejamento online de rotas para múltiplos robôs móveis em ambiente multiagente / Tauã Milech Cabreira. – 2013.
55 f.

Dissertação (mestrado) – Universidade Federal do Rio Grande/FURG, Programa de Pós-Graduação em Modelagem Computacional.

Orientador: Dr. Marilton Sanchotene de Aguiar.
Coorientadora: Drª. Graçaliz Pereira Dimuro.

1. Algoritmos genéticos. 2. Planejamento de rotas. 3. Robôs móveis. 4. Ambientes dinâmicos. I. Aguiar, Marilton Sanchotene de. II. Dimuro, Graçaliz Pereira. III. Título.

CDU: 004.421

Catalogação na fonte: Bibliotecária Alessandra de Lemos CRB10/1530

Dedico este trabalho à minha família amada,
Rubiner, Elaine e Cláudia,
que estão sempre comigo nos momentos bons e ruins da vida.

AGRADECIMENTOS

Agradeço a minha família pela educação, amor, carinho e suporte que me proporcionaram sempre e pelo exemplo e importância que representam em minha vida.

Agradeço ao meu orientador e amigo professor Dr. Marilton Sanchotene de Aguiar, que desde o início desta jornada esteve ao meu lado incentivando e apoiando o desenvolvimento deste trabalho. Agradeço pela solicitude, sinceridade e motivação em todos os momentos, assim como os ensinamentos e lições que levarei comigo para o resto da vida.

Agradeço a minha co-orientadora professora Dr[a] Graçaliz Pereira Dimuro pelo apoio e dedicação no progresso deste trabalho, pelo compartilhamento de seus conhecimentos e pelo crescimento acadêmico e profissional proporcionado.

Agradeço aos meus verdadeiros amigos, que torcem pelo meu sucesso e me motivam a ir cada vez mais longe. Todos nós já fomos colegas um dia, acabamos nos tornando amigos e hoje podemos nos considerar irmãos.

Agradeço aos colegas de mestrado pela ajuda e pela união nos momentos mais difíceis durante o curso.

Agradeço aos professores Norberto, Roger Albernaz de Araujo, Ricardo Andrade Cava, Paulo Roberto Ferreira Junior, Ricardo Matsumura Araujo, Marilton Sanchotene de Aguiar, Graçaliz Pereira Dimuro, Diana Adamatti, Leonardo Emmendorfer, Nelson Duarte Filho, Antônio Carlos da Rocha Costa e Éder Mateus Nunes Gonçalves. Agradeço pelos conhecimentos repassados, pelo exemplo que representam e pela inspiração que me proporcionaram. Graças a cada um de vocês eu também sou um professor hoje.

Agradeço a CAPES pelo auxílio na forma de bolsa de mestrado REUNI. Este trabalho é parcialmente financiado pelo CNPq (Proc. 476234/2011-5, 560118/10-4,305131/2010-9) e FAPERGS (Proc. 11/0872-3).

Agradeço a Universidade Federal de Rio Grande e ao Programa de Pós-Graduação em Modelagem Computacional, que além de proporcionar uma formação de qualidade, propiciaram a participação em diversos congressos e eventos para a apresentação de trabalhos.

Não é preciso ter olhos abertos para ver o sol,
nem é preciso ter ouvidos afiados para ouvir o trovão.
Para ser vitorioso você precisa ver o que não está visível. — SUN TZU

RESUMO

CABREIRA, Tauã Milech. **Uma Abordagem Evolucionária para o Planejamento Online de Rotas para Múltiplos Robôs Móveis em Ambiente Multiagente**. 2013. 55 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Modelagem Computacional. Universidade Federal do Rio Grande, Rio Grande.

O problema de planejamento de rotas de robôs móveis consiste em determinar a melhor rota para um robô, em um ambiente estático e/ou dinâmico, que seja capaz de deslocá-lo de um ponto inicial até um ponto final, também conhecido como estado objetivo. O presente trabalho emprega o uso de uma abordagem baseada em Algoritmos Genéticos para o planejamento de rotas de múltiplos robôs em um ambiente complexo composto por obstáculos fixos e obstáculos móveis. Através da implementação do modelo no software do NetLogo, uma ferramenta utilizada em simulações de aplicações multiagentes, possibilitou-se a modelagem de robôs e obstáculos presentes no ambiente como agentes interativos, viabilizando assim o desenvolvimento de processos de detecção e desvio de obstáculos. A abordagem empregada busca pela melhor rota para robôs e apresenta um modelo composto pelos operadores básicos de reprodução e mutação, acrescido de um novo operador duplo de refinamento capaz de aperfeiçoar as melhores soluções encontradas através da eliminação de movimentos inúteis. Além disso, o cálculo da rota de cada robô adota um método de geração de subtrechos, ou seja, não calcula apenas uma única rota que conecta os pontos inicial e final do cenário, mas sim várias pequenas subrotas que conectadas formam um caminho único capaz de levar o robô ao estado objetivo. Neste trabalho foram desenvolvidos dois cenários, para avaliação da sua escalabilidade: o primeiro consiste em um cenário simples composto apenas por um robô, um obstáculo móvel e alguns obstáculos fixos; já o segundo, apresenta um cenário mais robusto, mais amplo, composto por múltiplos robôs e diversos obstáculos fixos e móveis. Ao final, testes de desempenho comparativos foram efetuados entre a abordagem baseada em Algoritmos Genéticos e o Algoritmo A*. Como critério de comparação foi utilizado o tamanho das rotas obtidas nas vinte simulações executadas em cada abordagem. A análise dos resultados foi especificada através do Teste t de Student.

Palavras-chave: Algoritmos Genéticos, Planejamento de Rotas, Robôs Móveis, Ambientes Dinâmicos.

ABSTRACT

CABREIRA, Tauã Milech. **An Evolutionary Learning Approach for Multiple Mobile Robot Path Planning in a Multi-Agent Environment**. 2013. 55 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Modelagem Computacional. Universidade Federal do Rio Grande, Rio Grande.

The problem of path planning of mobile robots is to determine the best path for a robot in a static and/or dynamic environment , that is capable of moving a robot from a start point to a final point, also know as goal state. This study employs the use of an approach based on Genetic Algorithms for path planning of multiple robots in an complex environment composed by fixed and mobile obstacles. Through the implementation of the model in NetLogo software, a tool used in simulations of multi-agent applications, robots and obstacles in the environment could be modeled as interactive agents, thus enabling the development of process of detection and obstacle avoidance. The approach was employed in search of the best path to robots and presents a model composed by basic operators of reproduction and mutation, plus a new dual refinement operator that can improve the best solutions found by eliminating unnecessary movements. In addition, the path calculation of each robot adopts a method of generating mini-paths, i.e., does not calculate only one route that connects the starting and ending points of the scenario, but several small mini-paths that connected form a only path able to take the robot to the goal state. In this work were developed two scenarios for evaluation of its scalability: the first consists of a simple scenario composed by only one robot, one mobile obstacle and some fixed obstacles, whereas the second presents a scenario more robust, broader, composed by multiple robots and various fixed and mobile obstacles. Finally, comparative performance tests were carried out between the Genetic Algorithm approach and the A* Algorithm. As comparison criteria was used the size of the obtained paths in the twenty simulations performed on each approach. The analysis of the results was specified through the Student's t test.

Keywords: Genetic Algorithm, Path Planning, Mobile Robots, Dynamic Environments.

LISTA DE FIGURAS

Figura 1 Modelos iRobot com diversas funcionalidades: Limpeza doméstica (chão e piscina) e Segurança 15
Figura 2 Abordagem Clássica de Roadmap: Grafo de Visibilidade 16
Figura 3 Abordagem Clássica de Roadmap: Diagrama de Voronoi 17
Figura 4 Abordagem Clássica de Decomposição de Células 18
Figura 5 Abordagem de Histograma Vetorial 19

Figura 6 Fluxograma do Modelo Evolucionário de Planejamento de Rotas de Robôs 24
Figura 7 Representação dos oito movimentos e um exemplo de rota 25
Figura 8 Processo de Seleção por Torneio 27
Figura 9 Operador de crossover de um ponto 28
Figura 10 Operador de mutação atuando sobre um gene de um cromossomo . . 30
Figura 11 Otimização da rota através do operador de refinamento 32

Figura 12 Ambiente do Modelo Inicial x Ambiente do Modelo Estendido 34
Figura 13 Interface do Software NetLogo 35
Figura 14 Botões de SETUP e GO 35
Figura 15 Sliders 35
Figura 16 Monitores e plot 36
Figura 17 Chooser 36
Figura 18 Botões switch 36
Figura 19 Cálculo da rota através de subtrechos 37
Figura 20 Solução final gerada pelo Algoritmo Genético após 3 recálculos de rota 41
Figura 21 Gráfico apresentando a melhor e a pior solução após 20 gerações . . . 42

Figura 22 Exemplos de simulações do Algorimo A* (a) e do Algoritmo Genético (b) 44

LISTA DE TABELAS

Tabela 1 Relação entre abordagens e requisitos do problema de planejamento *online* de rotas 21

Tabela 2 Medida da quantidade de movimentos executados nas rotas 46

LISTA DE ABREVIATURAS E SIGLAS

AG	Algoritmo Genético
A*	Algoritmo A *Star*
DP	Desvio Padrão
GV	Grafos de Visibilidade
DV	Diagrama de Voronoi
DC	Decomposição de Células
CPA	Campo de Potencial Artificial
JD	Janelas Dinâmicas
HV	Histograma Vetorial
MAPF	Multi-agent Pathfinding
CBS	Conflict-Based Search

SUMÁRIO

1 INTRODUÇÃO 12
1.1 Objetivos 13
1.2 Organização do Texto 14

2 ESTADO DA ARTE 15

3 MODELO EVOLUCIONÁRIO DE PLANEJAMENTO DE ROTAS DE ROBÔS MÓVEIS 23
3.1 População de Indivíduos 23
3.2 Função de Avaliação 25
3.3 Seleção de Pais 26
3.4 Reprodução 28
3.5 Elitismo 29
3.6 Mutação 30
3.7 Geração 30
3.8 Refinamento 31

4 AMBIENTE DE SIMULAÇÃO MULTIAGENTE 33
4.1 Modelo Inicial x Modelo Estendido 33
4.2 A Caracterização da Ferramenta 34
4.3 O Fluxo da Simulação 37
4.4 Movimentação dos obstáculos 39
4.5 Recálculo da Rota 40

5 SIMULAÇÕES E ANÁLISES DOS RESULTADOS 43
5.1 Abordagem do Algoritmo A* 43
5.2 Abordagem de Algoritmo Genético 44
5.3 Simulações 45
5.4 Resultados 47

6 CONCLUSÃO E TRABALHOS FUTUROS 48

REFERÊNCIAS 51

ANEXO A RELATÓRIO DE PUBLICAÇÕES 55

1 INTRODUÇÃO

O problema de planejamento de rotas de robôs móveis consiste em determinar uma rota para um robô, em um ambiente estático e/ou dinâmico, capaz de levá-lo de um estado inicial a um estado objetivo através do melhor caminho, ou seja, o caminho mais curto, evitando possíveis colisões com obstáculos.

Este tipo de problema pode ser empregado nos mais diversos tipos de situação, tais como: movimentação de robôs que executam tarefas em um chão de fábrica, deslocamento de ambulâncias e de patrulhas policiais em uma situação emergencial nas cidades (ex.: incêndios, terremotos, etc), planejamento da trajetória de veículos não-tripulados, entre outros.

Diversas abordagens clássicas já foram empregadas para a solução do problema de planejamento de rotas. Apesar de serem capazes de solucionar o problema em questão, tais abordagens são computacionalmente inviáveis em virtude do tempo gasto para encontrar uma solução ótima. Além disso, estas abordagens podem cair em soluções ótimas locais, onde a melhor solução global pode não ser encontrada. Em virtude das dificuldades encontradas nas abordagens clássicas, alternativas baseadas em abordagens evolucionárias, tais como Algoritmos Genéticos, têm sido empregadas para a solução deste tipo de problema.

Esta pesquisa busca o desenvolvimento de um modelo baseado na abordagem de Algoritmos Genéticos para tratar do problema de planejamento *online* de rotas de robôs móveis aplicado em ambientes dinâmicos. Estes ambientes são compostos por obstáculos fixos e móveis que são capazes de deslocar-se através do cenário. O problema consiste em encontrar uma rota para um robô que o leve de um ponto inicial até um ponto final sem colidir com os obstáculos. Em ambientes dinâmicos, o cenário modifica-se a cada instante devido à movimentação dos obstáculos. Desta forma, além da rota inicial planejada, é necessário realizar uma verificação constante para certificar-se que o robô não irá colidir com os obstáculos durante a execução de sua trajetória. Ao detectar a presença de obstáculos e a iminente colisão, o recálculo da rota deve ser executado, modificando a trajetória inicial do robô.

Para alcançar este fim, propõe-se um modelo evolucionário para o problema em

questão em uma plataforma de simulação multiagente, onde é possível modelar o Algoritmo Genético, testar diferentes variações e configurações do modelo e simular a interação entre o robô e os obstáculos presentes no ambiente.

1.1 Objetivos

Este trabalho tem como objetivo a proposta de um modelo multiagente baseado em Algoritmos Genéticos para o problema de planejamento de rotas de robôs móveis em ambientes dinâmicos. Para alcançar esse fim, tem-se os seguintes objetivos específicos:

1. Definir e implementar um modelo multiagente baseado em algoritmos ótimos para o planejamento *online* de rotas de robôs;

2. Propor e implementar um modelo multiagente evolucionário para o planejamento *online* de rotas de robôs;

3. Elaborar cenários para a realização de testes de desempenho comparativos entre os modelos baseados em algoritmos ótimos e algoritmos evolucionários;

4. Implementar mecanismos de detecção e desvio de obstáculos para evitar colisões, possibilitando o recálculo da rota por parte dos robôs;

5. Desenvolver um novo tipo de operador de refinamento capaz de otimizar as melhores soluções;

6. Desenvolver um novo mecanismo de geração de rotas baseado em subtrechos interconectados, reduzindo assim o tempo de processamento na busca pelas melhores soluções;

7. Construir cenários complexos com um grande número de obstáculos fixos e móveis para a realização de testes de desempenho comparativos entre as duas abordagens previamente mencionadas;

8. Testar o modelo baseado em uma abordagem evolucionária com múltiplos robôs no cenário;

9. Avaliar o modelo proposto em relação às abordagens empregadas no problema de planejameto *online* de rotas de robôs.

10. Comparar os resultados obtidos na abordagem de Algoritmos Genéticos e na abordagem do Algoritmo A*.

1.2 Organização do Texto

A presente dissertação está organizada da seguinte forma: no capítulo 2 apresenta-se o Estado da Arte da área de pesquisa, com a caracterização dos robôs móveis e do problema de planejamento de rotas, assim como a explanação sobre as abordagens clássicas e as ineficiências detectadas em sua utilização no problema em questão. Também são apresentados trabalhos relacionados à abordagem de Algoritmos Genéticos com as possibilidades de variações, configurações e modificações, além da agregação de diferentes técnicas ao modelo evolucionário.

No capítulo 3 será abordado o desenvolvimento do projeto de planejamento *online* de rotas de múltiplos robôs móveis com a abordagem de Algoritmos Genéticos.

No capítulo 4 apresentar-se-á o ambiente de simulação multiagente com todos os recursos utilizadas para a criação do cenário, movimentação dos obstáculos, prevenção de colisões e recálculo das rotas.

No capítulo 5 serão expostos os testes de desempenho comparativos realizados entre a abordagem de Algoritmos Genéticos e o Algoritmo A* com as configurações estipuladas para cada modelo.

E por fim, no capítulo 6 será realizada a conclusão do trabalho com uma análise do modelo proposto, dos resultados e objetivos alcançados. Complementa-se ainda com apontamentos para os trabalhos futuros.

2 ESTADO DA ARTE

Os robôs móveis são um importante aspecto a ser considerado em ambientes automatizados (RAJA; PUGAZHENTHI, 2012; NEDJAH; COELHO; MOURELLE, 2007), oferecendo uma ampla variedade de aplicações, tais como: robôs de serviço para pessoas idosas, veículos guiados automatizados para o transporte de mercadorias e/ou utensílios em uma fábrica ou em casa, robôs não-tripulados para o desarmamento de bombas, robôs de limpeza doméstica e de piscinas, robôs de exploração de planetas, robôs subaquáticos, sistemas de robôs remotos para a detecção de presença física, robôs de combate, robôs pessoais e robôs programáveis, que podem ser personalizados pelo usuário. Por exemplo, cita-se Carebot[1], iRobot[2] e os diversos robôs dos projetos de Willow Garage[3]. Nestas aplicações, os robôs móveis usualmente executam suas tarefas de navegação usando blocos de construção (SIEGWART; NOURBAKHSH, 2004; RAJA; PUGAZHENTHI, 2012).

Figura 1: Modelos iRobot com diversas funcionalidades: Limpeza doméstica (chão e piscina) e Segurança

[1] http://www.geckosystems.com/
[2] http://www.irobot.com/us/
[3] http://www.willowgarage.com/

Em geral, a navegação de robôs móveis envolve: (i) percepção do ambiente, (ii) localização e construção do mapa, (iii) cognição e planejamento de rota e (iv) controle de movimento. O planejamento de rota deve ser realizado de acordo com a tarefa do robô, através de decisões cognitivas. Desta forma, o problema de planejamento de rotas de um robô móvel pode ser expresso como: dados a localização inicial, a posição do ponto final e o mapa do ambiente incluindo obstáculos estáticos, como planejar uma rota sem colisões entre os dois pontos especificados, capaz de satisfazer os critérios de otimização, tais como distância, tempo ou energia, sendo a distância o critério mais comumente adotado (RAJA; PUGAZHENTHI, 2012; NEDJAH; COELHO; MOURELLE, 2007; LATOMBE, 1991). E o que acontece se o ambiente for dinâmico?

Na literatura, há um número significativo de abordagens para solucionar o problema de planejamento de rotas de robôs, conforme discutido por Raja e Pugazhenthi (RAJA; PUGAZHENTHI, 2012) e Latombe (LATOMBE, 1991). As duas categorias de algoritmos para o planejamento de rotas de robôs são os algoritmos *offline* e os algoritmos *online*. Estes algoritmos estão classificados de acordo com a informação disponível do ambiente. O planejamento de rotas de robôs *offline* (ou global) pode ser usado se todas as informações sobre os obstáculos estáticos e a trajetória de movimento dos obstáculos móveis no ambiente são conhecidas. Caso contrário, o robô móvel precisa obter as informações através de seus sensores, enquanto movimenta-se pelo ambiente. Em geral, o planejamento de rotas *online* constrói o caminho inicial *offline*, tornando-se *online* quando alguma mudança no ambiente é detectada.

A classe de algoritmos *offline* relacionados à abordagem de Espaço de Configuração (*C-space*) (LOZANO-PEREZ; WESLEY, 1979; LOZANO-PEREZ, 1983), foi discutida por (LATOMBE, 1991). Usando o *C-space* como conceito fundamental, diversas abordagens clássicas de planejamento de rotas como a abordagem *road map* (ex.: grafo de visibilidade e diagrama de Voronoi) e a abordagem de decomposição de células (método de *grid*) foram desenvolvidas.

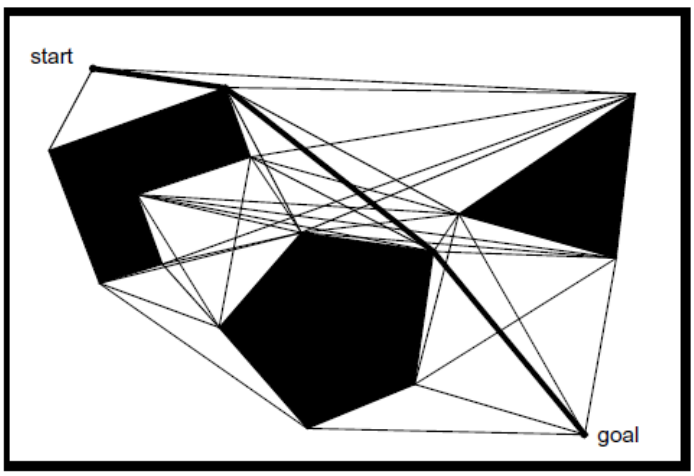

Figura 2: Abordagem Clássica de Roadmap: Grafo de Visibilidade

A abordagem *road map* captura a conectividade do espaço livre do robô em uma rede de curvas ou linhas chamada de *road maps*. Uma vez que uma *road map* é construída, ela é utilizada como uma rede de segmentos de rotas para a movimentação do robô. Sendo assim, o planejamento de rotas é reduzido à busca das séries de caminhos que conectam os pontos inicial e final dentro desta rede. Dentro desta abordagem *road map* podem-se destacar duas abordagens distintas em relação aos caminhos resultantes. No caso da abordagem dos grafos de visibilidade (GV) apresentada na Figura 2, as rotas aproximam-se drasticamente dos obstáculos, resultando em soluções com as menores rotas. Empregados em espaços de configuração poligonais, os grafos de visibilidade conectam todos os pares de vértices que cujas faces apontam um para o outro. As linhas desobstruídas que conectam estes vértices são as menores distâncias entre eles. Desta forma, para encontrar a menor rota basta procurar dentre os caminhos definidos pelo grafo de visibilidade (SIEGWART; NOURBAKHSH, 2004).

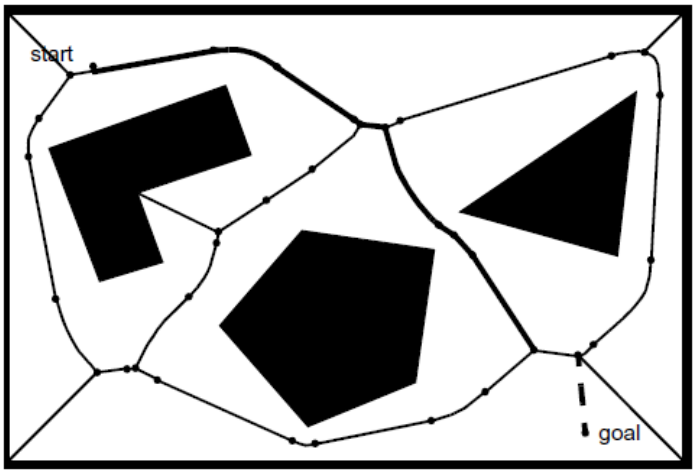

Figura 3: Abordagem Clássica de Roadmap: Diagrama de Voronoi

Já no diagrama de Voronoi (DV) apresentado pela Figura 3, as rotas ficam o mais longe possível dos obstáculos. Nesta abordagem a distância entre o robô e os obstáculos são maximizadas, pois as linhas construídas em todos os pontos são equidistantes de dois ou mais obstáculos. A direção do movimento no diagrama de Voronoi também é selecionada de modo que a distância até os limites aumenta mais rápido. Os pontos do diagrama de Voronoi representam transições a partir de segmentos de linha (distância mínima entre duas linhas) para segmentos parabólicos (distância mínima entre uma linha e um ponto) (SIEGWART; NOURBAKHSH, 2004).

Sobre a abordagem de decomposição de células (DC), ilustrada na Figura 4, a ideia principal é discernir áreas ou células geométricas que são livres e áreas que são ocupadas por objetos. Nesta abordagem, o ambiente é dividido em diversas faixas e um grafo de conectividade é gerado através das células adjacentes. A rota que liga os pontos inicial e final é procurada dentro do grafo de conectividade. Dependendo da decomposição resul-

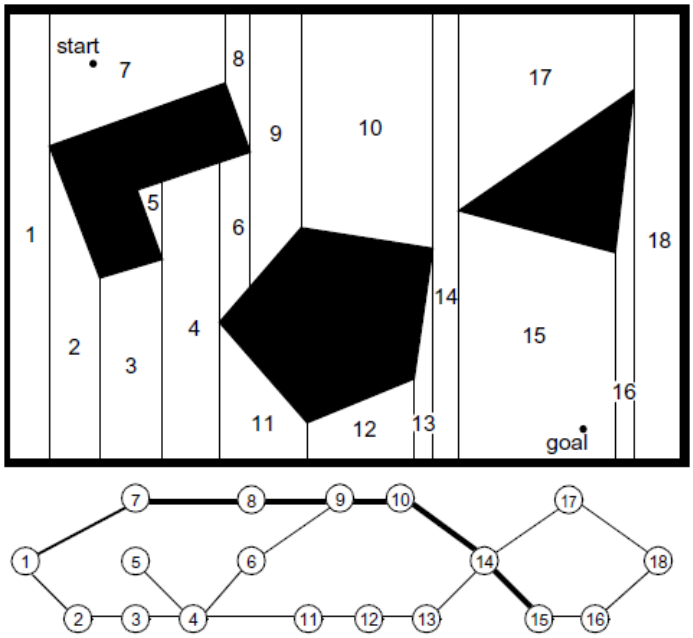

Figura 4: Abordagem Clássica de Decomposição de Células

tante do cenário em questão, pode-se ainda classificar a abordagem como decomposição de células exatas ou decomposição de células aproximada.

Também há a abordagem *Dead-reckoning*, que pode ser adotada em missões não-críticas ou fortemente condicionadas pelo tempo e/ou peso dos robôs. No entanto, o *Dead-reckoning* não é totalmente exato, o que pode levar a sérios erros de estimativa de distâncias (TU; YANG, 2003).

As abordagens clássicas, apesar de efetivas, são conhecidas por levarem mais tempo na determinação de caminhos praticáveis e livre de obstáculos. Outra deficiência deste tipo de abordagem é a tendência de ficar presa em soluções ótimas locais. Além disso, quando múltiplos obstáculos são considerados, o problema de planejamento de rotas torna-se NP-difícil (CANNY; REIF, 1987). O problema é mais difícil se o ambiente é dinâmico, mostrando que o uso das abordagens clássicas não é adequado para ambientes complexos (SUGIHARA; SMITH, 1997).

Abordagens *online* para o planejamento de rotas, tais como campo de potencial artificial, colisão-cone, histograma de campo vetorial, janelas dinâmicas e suas extensões e variações (algoritmos influentes, algoritmos de força de escape, modelo de recuperação de armadilha, algoritmo adaptativo de alvo virtual) têm sido amplamente discutidas desde o crescente interesse em robôs autônomos móveis que são capazes de operar em ambientes dinâmicos (LATOMBE, 1991; LUH; LIU, 2007, 2008; MASEHIAN; KATEBI, 2007; HACHOUR, 2008a,b).

Na abordagem de campo de potencial artificial (CPA) cria-se um campo ou gradiente em torno do mapa do robô que direciona-o até o estado objetivo a partir de múltiplas posições prévias. Este método trata o robô como um ponto sob a influência do campo de

potencial artificial, onde o estado objetivo atua como uma força de atração e os obstáculos atuam como uma força de repulsão. A sobreposição de todas as forças é aplicada ao robô, guiando-o suavemente até o estado objetivo e esquivando-o dos obstáculos conhecidos do ambiente.

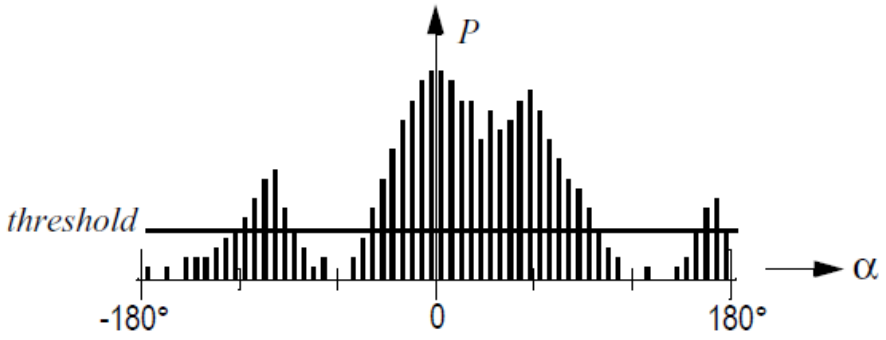

Figura 5: Abordagem de Histograma Vetorial

Na Figura 5 apresenta-se a abordagem de histograma vetorial (HV), que por sua vez, é capaz de criar um mapa local do cenário ao redor do robô. Este mapa local é um pequeno grid de ocupação determinado por uma escala de leitura recente dos sensores do robô. Para evitar colisões com obstáculos, a abordagem gera um histograma polar com o eixo-x representando o ângulo α em que o obstáculo foi detectado e o eixo-y representando a probabilidade *P* de que realmente existe um obstáculo nesta direção baseada no grid de ocupação.

E ainda tem-se a abordagem de janelas dinâmicas (JD) que pode ser dividida em duas abordagens na literatura: local e global. Na abordagem local, a cinemática do robô é levada em consideração através da busca de um espaço de velocidade bem escolhido. Na abordagem global, adiciona-se um conhecimento mais amplo ao algoritmo através do marcação da distância de cada célula até o estado objetivo em uma zona retangular limitada. Se o robô não conseguir alcançar o estado objetivo devido às restrições, aumenta-se o tamanho da região.

Embora as abordagens clássicas para o problema de planejamento de rotas *online* sejam efetivas, o tempo computacional aqui é crucial para o sucesso de qualquer algoritmo, visto que resultados ótimos dificilmente podem ser alcançados em um tempo computacional muito breve, devido à falta de informação sobre o ambiente. Por isso, abordagens evolucionárias como os Algoritmos Genéticos, a Otimização por Nuvem de Partículas, a Otimização por Colônias de Formigas e a Têmpera Simulada têm sido empregadas para solucionar o problema de planejamento de rotas, combinadas com abordagens clássicas (RAJA; PUGAZHENTHI, 2009a, 2012; TAMILSELVI et al., 2012; NEDJAH; COELHO; MOURELLE, 2007).

A abordagem de Algoritmos Genéticos surgiu como uma importante alternativa

em diversos trabalhos recentes relacionados ao problema do planejamento de rotas de robôs (HU; YANG, 2004; LI et al., 2006; LIN; WANG; WU, 2006; SADATI; TAHERI, 2002; TU; YANG, 2003; TAMILSELVI et al., 2012; RAJA; PUGAZHENTHI, 2009b, 2011; NEDJAH; COELHO; MOURELLE, 2007; BHADURI, 2009). Um Algoritmo Genético (AG) é uma variante de busca em feixe estocástica, na qual os estados sucessores são gerados pela combinação de dois estados pais, em vez de serem gerados pela modificação de um único estado (GOLDBERG, 2000).

Além da extensa possibilidade de variações, configurações e modificações, a abordagem de AG's possibilita ainda a agregação de outras técnicas que complementem e/ou aprimorem os resultados obtidos. Em (SADATI; TAHERI, 2002), a abordagem de AG's foi combinada com o uso das Redes Neurais de Hopfield para evitar a colisão com obstáculos no processo de crossover entre dois caminhos distintos. Através do uso deste tipo de rede neural, existe a possibilidade de adicionar novos nodos na rota do robô, gerando desvios ao longo do trajeto e, evitando assim, a colisão com obstáculos. Já em (LIN; WANG; WU, 2006), o campo de potencial numérico foi agregado aos AG's, sendo levado em consideração o *feedback* da posição e dos movimentos dos obstáculos para evitar colisões através de replanejamento da rota.

Além da agregação de novas técnicas ao uso de AG's na resolução do problema de planejamento de rotas, também são realizadas mudanças nas características tradicionais do modelo. Este é o caso do trabalho de (LI et al., 2006), onde um AG híbrido é proposto com a finalidade de evitar uma convergência prematura no espaço de soluções. Para tal, uma estratégia de controle *fuzzy* é empregada para o ajuste das probabilidades de crossover e mutação, tornando o modelo autoadaptativo.

Conhecimentos adicionais sobre o problema em questão também podem ser empregados no modelo de AG's. Em (HU; YANG, 2004), o conhecimento do domínio foi incorporado em operadores especializados, combinados com técnicas de busca local. Neste modelo, foram desenvolvidos alguns operadores dedicados especialmente ao refinamento das soluções, onde se podem ajustar linhas entre dois pontos que atravessam um obstáculo, um ponto que esteja localizado justamente sobre um obstáculo ou até mesmo o posicionamento dos nodos.

Mais recentemente, o algoritmo de Estratégia de Navegação foi proposto em (TAMILSELVI et al., 2012) para encontrar uma rota ótima em um ambiente simulado de grid. O AG descobre a rota que conecta as posições inicial e final do robô através de pontos predefinidos. O tamanho dos cromossomos é dinâmico e o genoma representa um nodo em um ambiente de grid 2D. Porém, detectou-se um problema de perda do melhor cromossomo durante a execução dos operadores de reprodução. Através do conceito de elitismo, a riqueza da população foi mantida e o problema foi solucionado.

Através dos trabalhos pesquisados e analisados, percebe-se o uso das mais variadas técnicas em conjunto com os AG's. A própria abordagem dos AG's muda drasticamente

Tabela 1: Relação entre abordagens e requisitos do problema de planejamento *online* de rotas

	Mapa Local	**Percepção do Ambiente**	**Complexidade do Cenário**	**Ambiente Dinâmico**	**Tipo de Rota**	**Problemas**
GV	Não	Local	Baixa	Não	Única	Aplicável apenas em obstáculos poligonais.
DV	Não	Local	Baixa	Não	Única	Rotas equidistantes dos obstáculos não são as menores.
DC	Não	Local	Baixa	Não	Única	-
CPA	Não	Local	Baixa	Não	Única	-
JD	Sim	Global	Baixa	Não	Única	Mínimos Locais.
HV	Sim	Local	Baixa	Não	Única	Mínimos Locais e Trajetórias Oscilantes.
A*	Não	Global	Alta	Sim	Única	-
AG	Não	Global	Alta	Sim	Partes	-

conforme a proposta de trabalho desenvolvida, variando na modelagem do ambiente, operadores utilizados, composição dos cromossomos, etc.

Na Tabela 1, apresenta-se a relação entre as diversas abordagens clássicas estudadas e os requisitos necessários à implementação do modelo de planejamento *online* de rotas de robôs móveis em ambientes dinâmicos. Agregam-se ainda à tabela, as abordagens de Algoritmos Genéticos, objeto de estudo desta dissertação, e do Algoritmo A*, que será adotado como método de comparação nos testes de qualidade das soluções. Percebe-se que as abordagens clássicas restringem-se a ambientes cuja configuração é de baixa complexidade ou limitam-se à forma dos obstáculos, como é o caso da abordagem de grafos de visibilidade, que somente pode ser empregada em ambientes com obstáculos poligonais. Além disso, os ambientes empregados nestas abordagens não são dinâmicos, ou seja, não possuem obstáculos móveis. O conhecimento limitado do ambiente, devido à falta de informação, também é um fator determinante no problema em questão.

No âmbito do problema de planejamento de rotas de múltiplos robôs, também conhe-

cido na literatura como *multi-agent pathfinding* (MAPF), trabalhos recentes apresentam diferentes abordagens em busca de soluções específicas para o problema, dependendo do ambiente. Tais abordagens podem ser ainda divididas em métodos acoplados (*couple methods*) e técnicas desacopladas (*decoupled techniques*). A abordagem de métodos acoplados, implementada em (SURYNEK, 2009; RYAN, 2008), considera os agentes como um sistema complexo com alta dimensionalidade. Já na abordagem de técnicas desacopladas, empregada em (WANG; BOTEA, 2008; JANSEN; STURTEVANT, 2008a,b), as rotas são computadas individualmente e as colisões são tratadas conforme são detectadas.

Em (KRONTIRIS; SAJID; BEKRIS, 2012), investiga-se o uso de soluções discretas e combinatoriais para resolver desafios em espaços contínuos com agentes cuja geometria são idênticas. O espaço de configuração é discretizado, atingindo uma representação baseada em grafos para um simples agente. Na sequência, uma solução combinatorial para a versão discreta do problema é estudada e caso seja encontrada, a rota discreta é transcrita para a rota contínua no espaço de configuração de todos os agentes.

Já no trabalho de (STANDLEY, 2012), o método chamado de Detecção Independente (DI) é abordado. Através deste método, problemas de maior complexidade são divididos em problemas de menor complexidade que podem ser solucionados separadamente. O método DI precisa ser utilizado em conjunto com um algoritmo de busca completa, como o A*. Além disso, a técnica faz uso de medidas de refinamento através de uma tabela de movimentos ilegais e de uma tabela de colisões para aprimorar os resultados obtidos.

Em (SHARON et al., 2012), é proposto um algoritmo alternativo de dois níveis intitulado de *Conflict-Based Search* (CBS). No nível mais alto, é efetuada uma busca em uma árvore baseada nos conflitos entre os agentes. No nível mais baixo, uma busca é executada para apenas um agente de cada vez. Ainda há uma generalização do algorimo chamada de *Meta-Agent* CBS, aonde um grupo de agentes é reunido em meta-agentes caso o número de conflitos entre eles exceda um limite determinado pelo modelo.

Com base nos estudos realizados e nas ponderações efetuadas acerca do problema de planejamento de rotas de robôs móveis em ambientes dinâmicos, além do referencial teórico explorado em trabalhos recentes na área de Algoritmos Genéticos, propõe-se um modelo evolucionário para o cálculo de rotas de múltiplos robôs em um ambiente dinâmico complexo composto por inúmeros obstáculos fixos e móveis. Este modelo foi especificado através do software NetLogo, cujas ferramentas possibilitaram a implementação de robôs e obstáculos como agentes interativos no ambiente, imitando assim o comportamento de sensores de robôs, possibilitando a detecção de obstáculos e a prevenção de colisões entre as entidades envolvidas.

3 MODELO EVOLUCIONÁRIO DE PLANEJAMENTO DE ROTAS DE ROBÔS MÓVEIS

Os Algoritmos Genéticos foram inventados por John Holland nos anos 60, cujo principal objetivo não foi desenvolver algoritmos para solucionar problemas específicos, mas dedicar-se ao estudo formal do fenômeno de evolução, como ocorre na natureza, e desenvolver maneiras de importá-lo aos sistemas de computação (HOLLAND, 1975). Neste modelo, tem-se inicialmente um conjunto de estados (indivíduos) gerados aleatoriamente intitulado de população. Cada indivíduo é representado por um único cromossomo, contendo a codificação (genótipo) de um candidato à solução de um determinado problema.

Baseando-se na Teoria da Evolução, os Algoritmos Genéticos são capazes de reproduzir os fenômenos evolucionários e adaptivos dos seres vivos na natureza como uma forma de buscar soluções para os mais diversos tipos de problemas. Uma das principais metáforas adotadas na concepção da abordagem de Algoritmos Genéticos indica que indivíduos mais adaptados ao meio ambiente possuem maiores chances de sobrevivência, caracterizando um processo de seleção natural. Com maiores probabilidades de sobrevivência em relação aos demais indivíduos, as chances de reprodução também elevam-se e, por conseguinte, a capacidade de transmitir seus genes para as próximas gerações. As próximas gerações, por sua vez, apesar de carregarem fortes traços genéticos de seus ancestrais, não são exatamente idênticas aos seus antecessores. Isto deve-se ao fato da recombinação genética entre diferentes seres e da mutação que ocorre entre diferentes gerações. Ainda em tempo, indivíduos com capacidade de sobrevivência inferiores também estão aptos à reprodução, embora em menor escala.

Na Figura 6, apresenta-se o fluxograma do modelo evolucionário proposto para o planejamento de rotas de robôs móveis. Na sequência, serão descritos detalhadamente cada um dos elementos do modelo ilustrados graficamente na Figura 6.

3.1 População de Indivíduos

Os Algoritmos Genéticos trabalham com o conceito de população de indivíduos. Ao invés de trabalhar com apenas uma única solução para um determinado problema, os

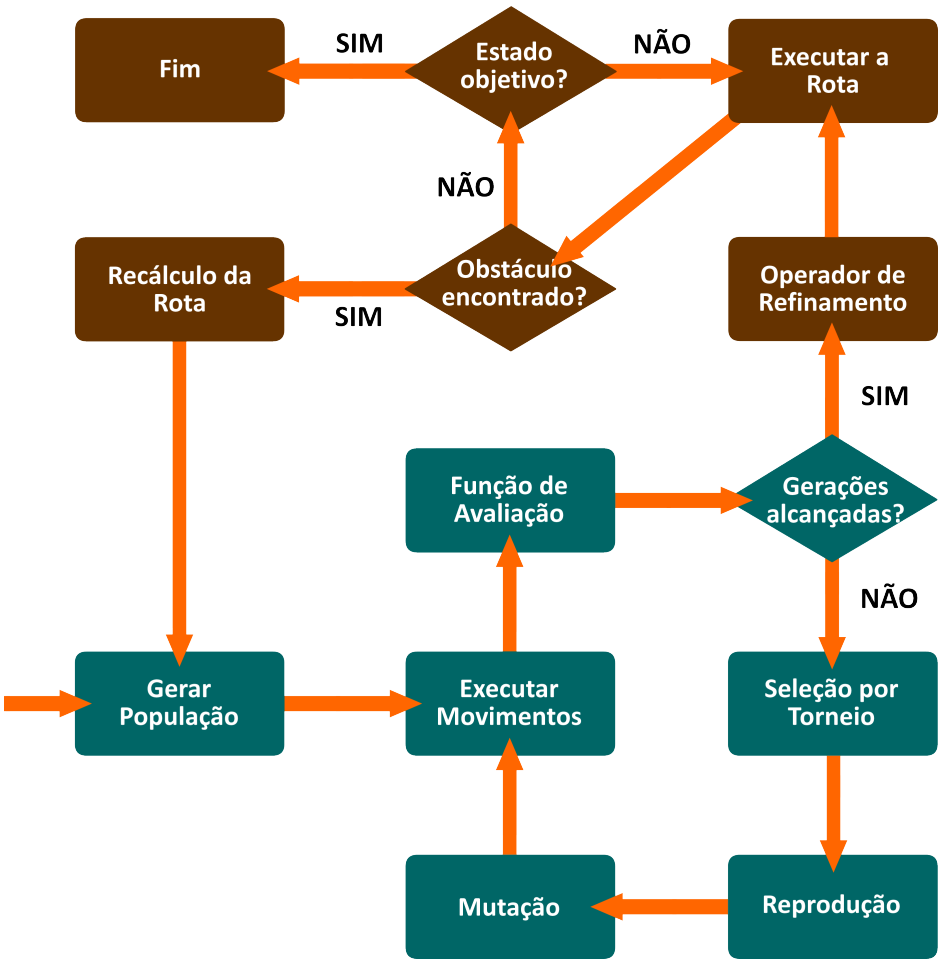

Figura 6: Fluxograma do Modelo Evolucionário de Planejamento de Rotas de Robôs

Algoritmos Genéticos lidam com diversas soluções ao mesmo tempo. Cada uma destas soluções é especificada por um indivíduo da população, que por sua vez, é definido através de um cromossomo. Este cromossomo é a representação vetorial binária (bits de 0's e 1's) de uma possível solução para o problema em questão. Normalmente, a população de indivíduos é gerada aleatoriamente no início do processo.

No modelo proposto, uma população inicial de indivíduos é gerada randomicamente. Cada um destes indivíduos é composto por um cromossomo que constitui uma determinada rota. O cromossomo é formado por um vetor binário que representa uma sequência de movimentos, onde cada movimento é codificado em três bits, necessários para codificar as oito possibilidades de movimentação, conforme observa-se na Figura 7. Os movimentos são codificados no cromossomo da seguinte forma: direita (000), acima-direita (001), acima (010), acima-esquerda (011), esquerda (100), abaixo-esquerda (101), abaixo (110) e abaixo-direita (111). Por exemplo, um cromossomo, que representa um percurso com 15 movimentos, composto por 15 genes, tem como fenótipo 001001001001000001001001010010011001001001000.

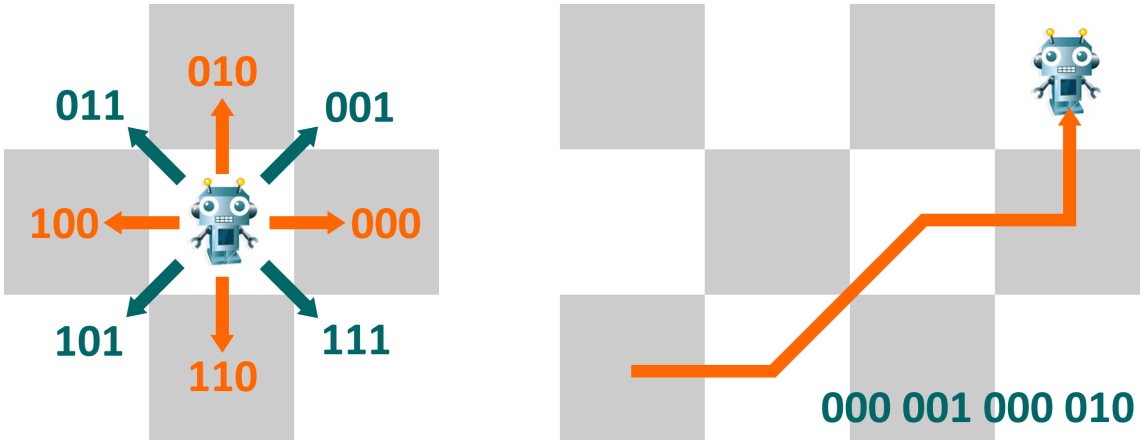

Figura 7: Representação dos oito movimentos e um exemplo de rota

A configuração adotada pelos cromossomos é conhecida como genótipo de um indivíduo (usualmente diz-se que cada indivíduo possui apenas um cromossomo, tornando os termos cromossomos e indivíduos sinônimos). Já a característica que o cromossomo atribui ao indivíduo é conhecida como fenótipo. Genótipos diferentes podem resultar em fenótipos satisfatórios para sobrevivência do indivíduo.

O problema de planejamento de rotas explorado neste trabalho considera apenas a distância como um critério de otimização, ou seja, trata-se de um problema de minimização pela busca da menor rota que conecte os pontos inicial e final. Desta forma, os cromossomos do modelo podem possuir tamanhos distintos.

Todos os indivíduos que compõem a população são possíveis candidatos à solução do problema proposto. Neste instante, no entanto, encontram-se no modelo apenas soluções geradas aleatoriamente sem nenhum critério de escolha. O próximo passo consiste em realizar os movimentos presentes em cada cromossomo para calcular valor da função de avaliação, também conhecida como *fitness*, de cada indivíduo.

3.2 Função de Avaliação

A função de avaliação é o método adotado para inferir a qualidade de cada indivíduo da população como uma das possíveis soluções para o problema em questão. Esta função retorna um valor ou uma lista de valores que externará o desempenho do indivíduo diante do problema. A codificação dos cromossomos e a função de avaliação são exclusivas para cada problema proposto, portanto devem englobar todo o conhecimento adquirido sobre o problema a ser resolvido, agregando tanto as suas restrições quanto seus objetivos de qualidade. Dada a generalidade dos Algoritmos Genéticos, a função de avaliação, em muitos casos, é a única ligação verdadeira do programa com o problema real. Sendo assim, para que seja viável a avaliação criteriosa de cada uma das possíveis soluções e que possibilite-se o emprego da função de avaliação como parâmetro para a escolha dos

indivíduos no módulo de seleção de pais, faz-se necessário que a função penalize soluções impraticáveis e bonifique aquelas que destacam-se diante das demais soluções.

Em um problema de maximização, quanto maior for o valor calculado pela função de avaliação, melhor será a solução encontrada. O problema do planejamento de rotas de robôs móveis consiste em um problema de minimização, pois se almeja determinar a melhor rota de deslocamento de robôs a partir de um ponto inicial até um estado objetivo. Portanto, a melhor rota é determinada pela rota mais curta, ou seja, o menor cromossomo, desde que evite as colisões com os obstáculos.

Para calcular o valor da função de avaliação, são avaliados todos os movimentos presentes no cromossomo de cada indivíduo da população. A cada movimento, verifica-se a presença de obstáculos pelo caminho, penalizando aqueles indivíduos cuja rota colide com os obstáculos. Caso não haja obstáculos no local selecionado para o movimento, uma nova verificação é realizada, mas desta vez relacionada ao estado objetivo. Se o local escolhido for o estado objetivo a ser alcançado pela rota, a solução é bonificada em sua função de avaliação. E finalmente, se o local escolhido for apenas um lugar comum, sem obstáculos ou estado objetivo presente, o valor da função de avaliação é apenas acrescido do valor correspondente ao deslocamento do movimento.

A seguir encontra-se a função de avaliação adotada no modelo proposto:

$$Fitness = \sum_{i=0}^{n} dist \times (1 + peso) + (maxx - px + maxy - py), \tag{1}$$

onde: *dist* é o valor relacionado ao trecho percorrido no movimento (movimentos para frente, para trás ou para os lados consomem uma unidade de distância; já os movimentos diagonais consomem $\sqrt{2}$ unidades de distância); *peso* é o valor atribuído pela penalização ou bonificação (este valor será zero caso nenhuma das opções anteriores seja atribuída à variável); *maxx* e *maxy* representam os valores da dimensão máxima do ambiente; *px* e *py* indicam a posição atual da solução em relação aos eixos x e y.

Após a realização dos movimentos presentes nos cromossomos e o cálculo da função de avaliação de todos os indivíduos da população inicial, dá-se início ao processo de seleção dos indivíduos para a aplicação dos operadores de reprodução e mutação.

3.3 Seleção de Pais

O método de Seleção de Pais é empregado na escolha dos indivíduos que irão submeter-se ao processo de reprodução. Este método assemelha-se ao processo de seleção natural encontrado na natureza e deve favorecer a escolha dos indivíduos mais aptos da população, ou seja, com maior valor obtido através da função de avaliação. Ao mesmo tempo, o método não deve ignorar indivíduos cujos índices avaliativos sejam considerados baixos, pois eles podem conter propriedades favoráveis à geração de soluções superiores

nas gerações futuras. Além disso, descartar os indivíduos inaptos pode ocasionar uma rápida convergência genética de todas as soluções para um mesmo conjunto de características, impedindo uma busca mais ampla no espaço de soluções.

Existem diferentes métodos de Seleção de Pais. Entre estes métodos pode-se destacar a Roleta Viciada e a Seleção por Torneio. No método da Roleta Viciada, cria-se uma roleta virtual dividida proporcionalmente de acordo com os valores obtidos por cada indivíduo através da função de avaliação. Números aleatórios são emitidos para a escolha dos pais. Naturalmente, indivíduos com maior valor de avaliação possuem uma grande probabilidade de serem selecionados, visto que ocupam uma fatia maior na roleta. Indivíduos inferiores, ou seja, com baixo índice de avaliação também dispõe de chances de seleção, mesmo que pequenas, agregando maior diversidade genética à população.

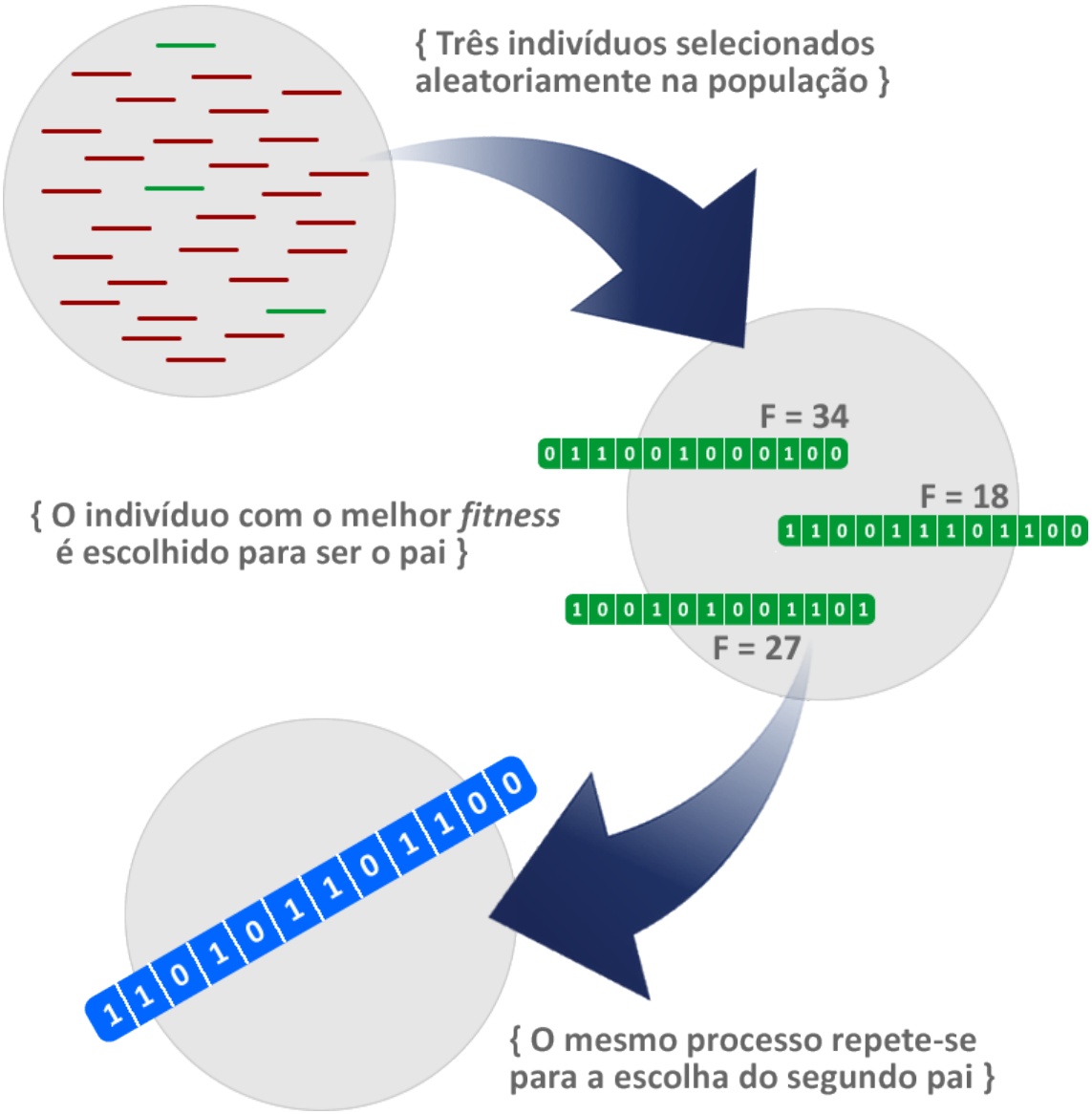

Figura 8: Processo de Seleção por Torneio

Outro processo utilizado é o de Seleção por Torneio, ilustrado pela Figura 8. Neste

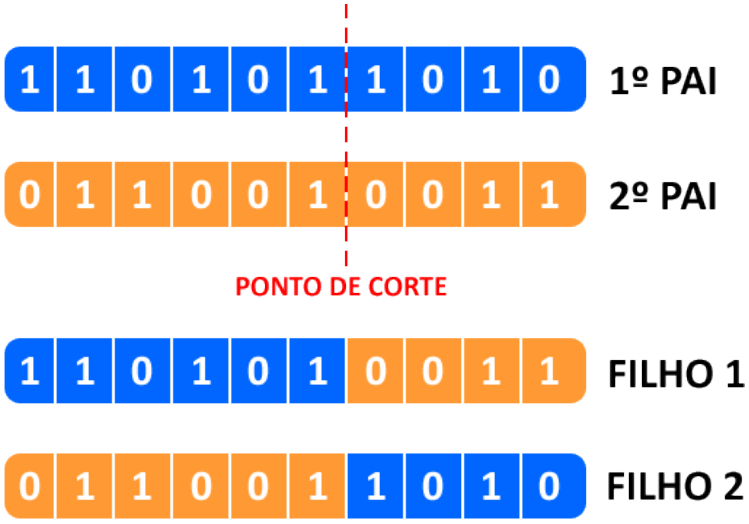

Figura 9: Operador de crossover de um ponto

processo, são selecionados aleatoriamente três indivíduos da população atual. O indivíduo com o melhor valor obtido através da função de avaliação entre os três é escolhido como um dos pais. O mesmo procedimento é executado para a escolha do segundo progenitor do novo indivíduo. Este foi o método de seleção empregado no modelo de planejamento de rotas proposto.

3.4 Reprodução

O processo de reprodução consiste na troca de material genético entre os indivíduos selecionados a partir da Seleção de Pais. Este procedimento resulta na geração de dois novos indivíduos compostos pela mescla dos cromossomos dos pais. Também é conhecido como recombinação genética ou *crossover*. O primeiro passo no processo de reprodução é a definição do método de corte do cromossomo. Existem dois métodos responsáveis pela divisão do material genético dos pais: o *crossover* de um ponto e o *crossover* de dois pontos.

No *crossover* de um ponto, o código genético (cromossomo) de cada um dos pais selecionados é dividido aleatoriamente em duas partes. Cada uma destas partes não possui necessariamente o mesmo tamanho. Na sequência, dois novos indivíduos são gerados e parte do cromossomo de cada progenitor é adicionada ao cromossomo do novo indivíduo. Neste caso, o primeiro indivíduo gerado a partir da reprodução armazenará em seu cromossomo, a primeira parcela dos genes do primeiro pai e a segunda parcela dos genes do segundo pai. Por sua vez, o segundo indivíduo gerado herdará a segunda parte dos genes do primeiro pai e a primeira parte dos genes do segundo pai. Na Figura 9, tem-se um exemplo de operador de crossover de um ponto empregado na geração de dois novos indivíduos.

Já no *crossover* de dois pontos, a principal mudança em relação ao primeiro operador é

no número de pontos que divide os cromossomos. Neste caso, os dois pontos de corte não são aleatórios, mas sim escolhidos de acordo com os padrões de qualidade encontrados. O objetivo é selecionar as melhores partes de material genético dos pais e repassá-las aos filhos para criar uma geração com desempenho ainda mais elevado.

Apesar da aparente superioridade do operador de *crossover* de dois pontos, optou-se pela utilização do *crossover* de um ponto no modelo proposto, pois os cromossomos desenvolvidos para o problema de planejamento de rotas são compostos por movimentos do robô. Desta forma, uma mesma sequência de movimentos (ex.: direita (000), acima (010), direita (000)) pode ser considerada excelente em uma parte do cenário e muito ruim em outra parte do ambiente, de acordo a disposição dos obstáculos e da posição do estado objetivo. Sendo assim, torna-se inviável a definição dos padrões de qualidade dos elementos que compõem o cromossomo. Além disso, a implementação do operador de *crossover* de um ponto é mais simples do que o operador mais sofisticado.

Outra característica importante do processo de reprodução é que nem todos os indivíduos da população procriam. No modelo define-se uma taxa de reprodução que orienta a porcentagem de indivíduos que serão submetidos à reprodução. Por exemplo, se a taxa de *crossover* for definida como 70, isto significa que 70% da nova população será gerada através da reprodução e os outros 30% serão meramente uma cópia dos indivíduos da atual população.

Ao término do processo de reprodução, os demais indivíduos da nova população são gerados por clonagem, ou seja, selecionados através do mesmo processo de Seleção por Torneio empregado na reprodução, porém sem o cruzamento genético entre dois indivíduos. Sendo assim, o indivíduo escolhido é apenas copiado para a geração seguinte.

3.5 Elitismo

Na natureza, os indivíduos mais fortes e melhores adaptados às condições adversas de seu ambiente podem reproduzir com maior frequência, assim como possuir maior longevidade, sobrevivendo por um período de tempo além de sua geração atual. Nos Algoritmos Genéticos, este processo natural é representado através do Elitismo. Através do Elitismo, preservam-se aqueles indivíduos cujo desempenho é superior em relação aos demais membros da população com base nos valores obtidos por meio da função de avaliação. Estes indivíduos não são submetidos aos processos de reprodução e mutação, e são simplesmente repassados de uma geração para outra. Desta forma, resguarda-se o seu material genético para que ele possa ser aproveitado na próxima geração.

No modelo proposto, o processo de Elitismo foi implementado para garantir que o melhor indivíduo da população atual seja copiado para a geração subsequente.

3.6 Mutação

No processo de reprodução entre os seres vivos, a troca de material genético e a concepção de um novo indivíduo não corresponde apenas a mescla de genes dos progenitores. Dependendo das circunstâncias e até mesmo de fatores externos, pequenos erros podem ocorrer ocasionando mutações no código genético. Estas mutações podem ser positivas, negativas ou não sortir efeito nenhum, mas são fundamentais no processo de evolução natural. Com base nestes conceitos é que concebeu-se o operador de mutação nos Algoritmos Genéticos.

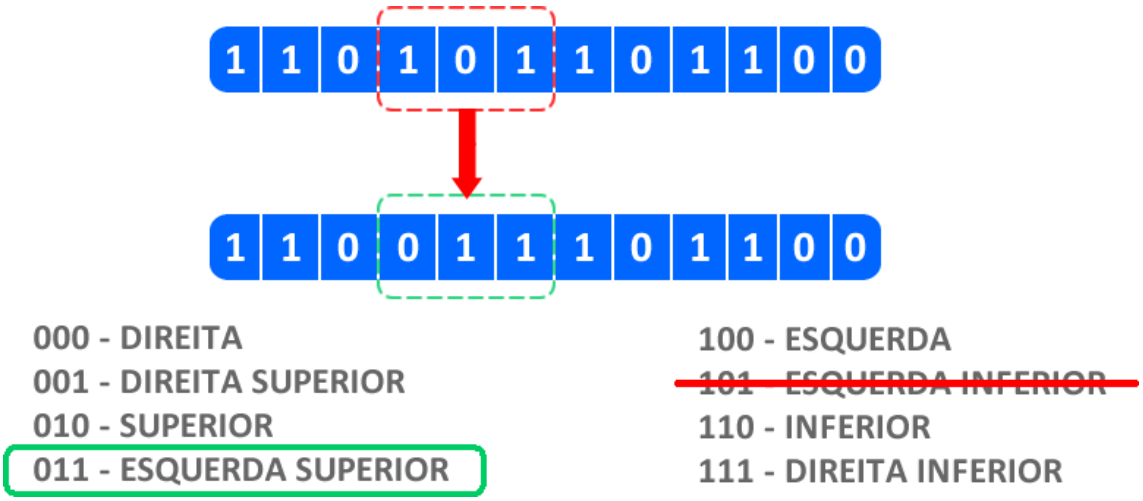

Figura 10: Operador de mutação atuando sobre um gene de um cromossomo

Após o processo de geração de novos indivíduos através da troca de material genético, estes são submetidos ao processo de mutação. Assim como o operador de reprodução, o operador de mutação também possui uma taxa correspondente à porcentagem de sua aplicação sobre os cromossomos. Desta forma, define-se esta taxa de aplicação e sorteia-se um número randômico, entre 0 e 1. Este procedimento é empregado em todos os genes dos dois novos filhos gerados a partir da reprodução. Caso o valor sorteado seja inferior ou igual à taxa de mutação definida, o gene do cromossomo em questão muda de valor, conforme demonstra-se na Figura 10.

A substituição de um dos genes do cromossomo através do processo de mutação consiste na alteração de um dos movimentos da rota do indivíduo. Em caso afirmativo de mutação em um dos genes, escolhe-se randomicamente um dos oito movimentos possíveis para substituí-lo. Através deste operador, adiciona-se diversidade à população de indivíduos.

3.7 Geração

A nova população gerada a partir dos processos de seleção, reprodução e mutação substitui a antiga população. Apesar de destoar da realidade dos seres vivos, aonde di-

versas gerações coexistem simultaneamente, este é o método aplicado na abordagem de Algoritmos Genéticos. Todos os indivíduos da antiga população morrem, dando lugar aos novos indivíduos. Quando isto acontece, considera-se o fim de uma era ou de uma geração. Este período de tempo engloba todos os procedimentos do Algoritmo Genético, desde a seleção de progenitores à concepção de novos indivíduos, incluindo ainda a eliminação da população antiga. É importante ressaltar também que as populações não variam de tamanho ao longo das gerações.

Com o fim de uma geração, todos os indivíduos da população são submetidos à movimentação através das rotas presentes em seus cromossomos para a realização do cálculo da função de avaliação. Através da movimentação dos indivíduos, é possivel verificar quais rotas colidem com obstáculos e quais rotas são favoráveis ao deslocamento dos robôs em direção ao estado objetivo. De posse destas informações, constroi-se a avaliação de cada cromossomo, penalizando-os e/ou bonificando-os de acordo com o cenário atual.

O processo é repetido até que o número de gerações ou um critério de parada específico determinado pelo modelo sejam atingidos. No modelo proposto, adotou-se um número fixo de gerações. Após o alcance deste número de gerações, cada robô define o seu trajeto com base na melhor rota obtida ao final do Algoritmo Genético. Mas a melhor rota de cada robô não é a rota definitiva. Resta ainda o emprego de um novo operador especialmente desenvolvido para este modelo: o operador de refinamento.

3.8 Refinamento

Além dos operadores tradicionais de seleção, reprodução e mutação, podem-se desenvolver operadores especifícos para determinados tipos de problema. Estes operadores são concebidos com o intuito de otimizar as melhores soluções e/ou corrigir eventuais falhas nas respostas obtidas através dos processos do Algoritmo Genético. A partir desta ideia, idealizou-se a implementação do operador de refinamento para o problema de planejamento de rotas.

Este operador duplo otimiza a melhor rota de cada robô eliminando os movimentos inúteis. O primeiro operador elimina movimentos cíclicos que não levam o robô a lugar nenhum. Por exemplo, movimentos para frente e para trás, para direita e para esquerda, para esquerda e para direita, incluindo todas as 8 (oito) possíveis direções. Nestes casos, o robô executa um movimento deslocando-se do lugar A para o B. Logo em seguida, o robô executa outro movimento retornando do lugar B para o A, retornando para um local visitado anteriormente. Sendo assim, o operador de refinamento pode aperfeiçoar a melhor rota e reduzir o número de genes do cromossomo.

O segundo operador de refinamento trabalha eliminando movimentos triplos. Por exemplo, a sequência tripla que leva o robô de A para B, de B para C e de C para A. Estes movimentos formam um triângulo na rota que também não levam o robô a lugar

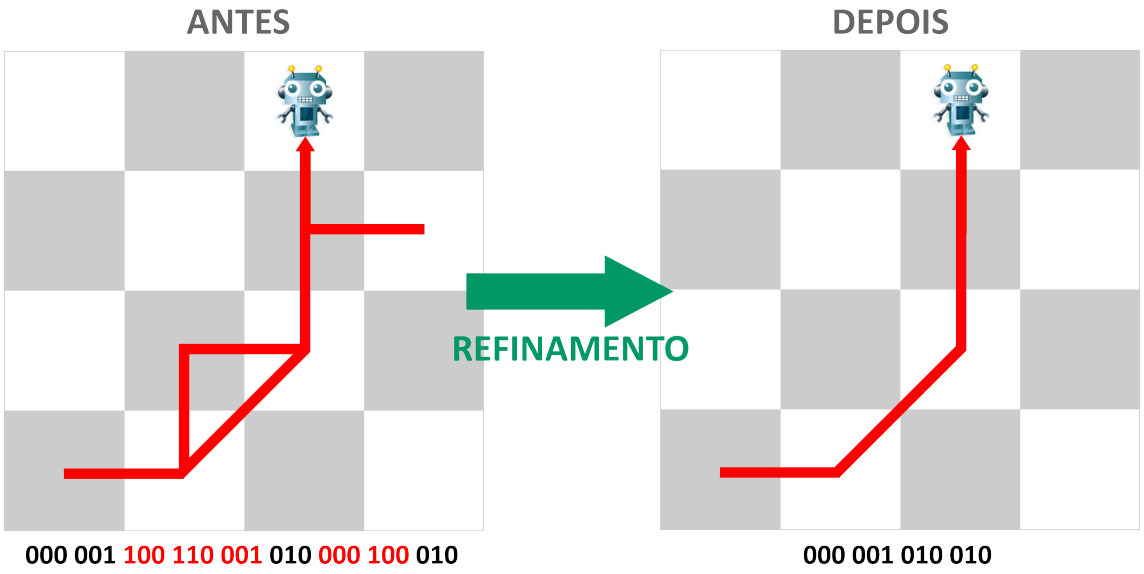

Figura 11: Otimização da rota através do operador de refinamento

nenhum. Assim como o primeiro operador, este também trabalha eliminando os movimentos cíclicos em todas as direções pré-definidas.

Na Figura 11, pode-se observar um exemplo da aplicação dos operadores de refinamento. A rota apresentada do lado esquerdo da imagem possui 9 (nove) movimentos, incluindo um movimento triangular (esquerda (100), abaixo (110) e acima-direita (001)) e de vai-e-vem (direita (000) e esquerda (100)). Estes movimentos são considerados inúteis pelo operador, visto que deslocam o robô pelo cenário, mas não modificam a sua posição em relação ao estado objetivo. Após o refinamento da rota por meio do operador, constata-se a eliminação dos movimentos desnecessários e a redução no número de genes do cromossomo, que passa de 9 (nove) para 4 (quatro), na imagem do lado direito. Com a otimização da rota, além do decréscimo no número de movimentos, reduz-se a probabilidade de colisão do robô com os obstáculos, uma vez que a rota passa a ser mais objetiva.

4 AMBIENTE DE SIMULAÇÃO MULTIAGENTE

A abordagem de Algoritmos Genéticos é uma técnica robusta e flexível, com eficiência comprovada na resolução de diversos tipos de problemas. O presente trabalho propõe uma abordagem evolucionária baseada em Algoritmos Genéticos para o problema de planejamento *online* de rotas para robôs móveis utilizando uma plataforma de desenvolvimento de sistemas multiagentes, o software NetLogo. Esta ferramenta é baseada na linguagem Logo e é amplamente utilizada por estudantes, professores e pesquisadores (WILENSKY, 2012).

Por meio desta plataforma, robôs e obstáculos foram implementados como agentes interativos, visto que o software possibilita, entre outras opções, a movimentação e a interação entre os agentes, bem como a modelagem de todos os processos de um Algoritmo Genético. Além disso, o software viabilizou a elaboração dos cenários de teste correspondentes à problemática e a realização de simulações em ambientes estáticos e dinâmicos. Através das simulações em diferentes cenários, analisou-se a capacidade do modelo proposto de determinar a escolha da melhor rota e de detectar a presença de obstáculos durante a execução das trajetórias dos robôs, evitando assim possíveis colisões.

4.1 Modelo Inicial x Modelo Estendido

O trabalho constitui-se em duas etapas distintas e progediu através da publicação de artigos científicos em congressos e eventos de áreas como Simulação Social, Sistemas Multiagentes, Sistemas *Fuzzy* e Computação Evolucionária. A primeira etapa foi apresentada em (CABREIRA; DIMURO; AGUIAR, 2012) com um modelo inicial composto por apenas um robô, um obstáculo móvel com cinco elementos e quatro obstáculos fixos, dois deles contendo cinco elementos e dois contendo quatro elementos. A dimensão do cenário inicial era de 15 x 15.

Já a segunda etapa do projeto foi apresentada em (CABREIRA; AGUIAR; DIMURO, 2013) e constituiu-se de uma versão estendida do modelo inicialmente proposto. O modelo estendido apresenta múltiplos robôs e doze obstáculos móveis, cada um com cinco elementos. Além disso, o tamanho do cenário e o número de obstáculos fixos foram am-

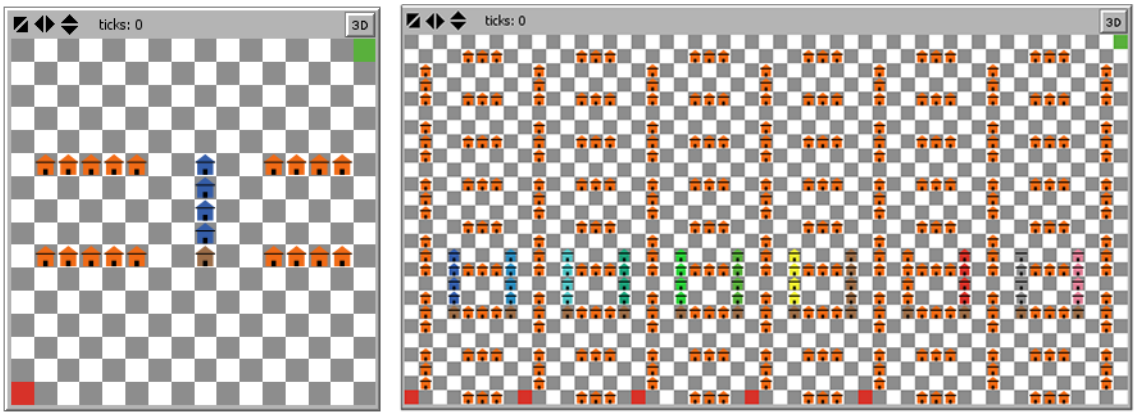

Figura 12: Ambiente do Modelo Inicial x Ambiente do Modelo Estendido

pliados. O novo cenário conta com uma dimensão de 50 x 25, enquanto que o número de obstáculos fixos aumentou em 16 vezes - agora são 96 obstáculos compostos por três elementos cada. A diferença entre os dois modelos pode ser visualmente observada na Figura 12.

Os *patches*[1] vermelho e verde representam os pontos iniciais e o estado objetivo, respectivamente. Os obstáculos móveis (*snakes*) são identificados pelos agentes sob a forma de casas marrom-coloridas, enquanto que os obstáculos fixos são caracterizados pelas casas laranjas. O cenário é marcado como um tabuleiro de xadrez para facilitar a visualização da movimentação dos robôs e dos obstáculos, bem como a observação das rotas traçadas durante a execução das simulações.

4.2 A Caracterização da Ferramenta

O ambiente de desenvolvimento do NetLogo é composto por três abas principais: *Interface*, *Info* e *Code*. Na aba *Interface*, pode-se controlar, executar e visualizar todas as ações do modelo. Por sua vez, na aba *Info*, são acrescentadas todas as informações e características do modelo, bem como a explicação de seu funcionamento. E por fim, na aba *Code*, têm-se todas as funções do modelo programadas na linguagem Logo. Na sequência serão descritos os elementos que compõem a interface e as funções relacionadas a estes elementos implementadas na aba *Code*. A explicação da aba *Info* dar-se-á ao longo do texto.

Na aba *Interface* do NetLogo, foram adicionados diversos elementos e botões para a configuração, a execução e o controle das simulações relacionadas ao problema de planejamento de rota para robôs móveis, conforme apresenta a Figura 13.

Os botões SETUP e GO são os responsáveis pela configuração e execução do modelo por meio de suas respectivas funções. Através do botão SETUP, o cenário é marcado nas

[1]Patch é um tipo de agente presente no NetLogo. São ladrilhos que dividem o ambiente em um grid, aonde os demais agentes (turtles) movimentam-se.

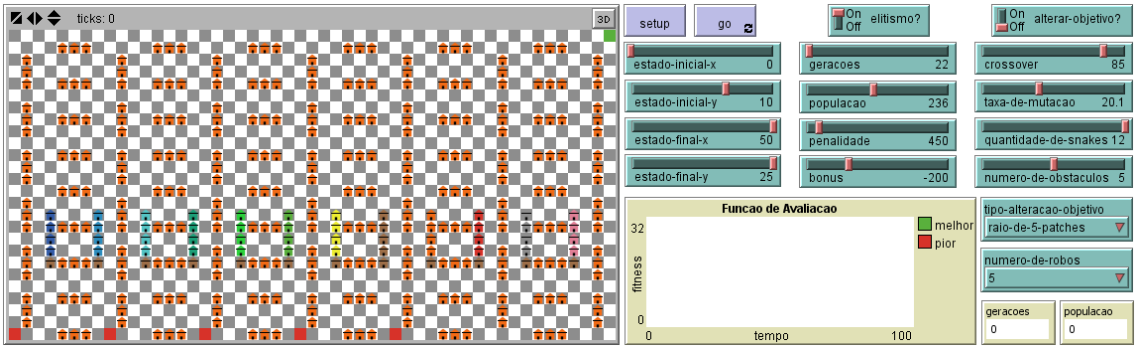

Figura 13: Interface do Software NetLogo

cores cinza e branco como um tabuleiro de xadrez, os obstáculos fixos e móveis são criados e posicionados no ambiente e as variáveis do modelo têm seus valores inicializados.

Figura 14: Botões de SETUP e GO

O botão GO inicia a simulação com a aplicação de um Algoritmo Genético para cada robô do cenário. Como principal função do modelo, o GO também controla o critério de parada da simulação e executa uma série de funções com diferentes atribuições, tais como a movimentação dos obstáculos e a impressão das rotas no cenário para visualização dos observadores da simulação. A simulação encerra-se quando todos os robôs atingem o estado objetivo.

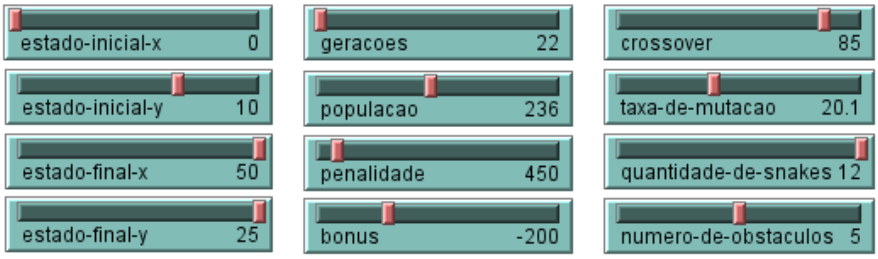

Figura 15: Sliders

Além dos botões SETUP e GO, existem os *sliders* cuja função é configurar os parâmetros adotados no Algoritmo Genético e determinar as principais características do ambiente. Entre os parâmetros configuráveis encontram-se o número de gerações, a quantidade de indivíduos da população, a taxa de reprodução (crossover) e de mutação e os índices de bônus e de penalidade da função de avaliação. Através dos *sliders*, também é possível determinar as posições inicial e final dos robôs, bem como o número e o tamanho de obstáculos móveis - a quantidade de obstáculos varia de 1 a 12, enquanto que o tamanho de cada obstáculo pode ser definido no intervalo de 1 a 10.

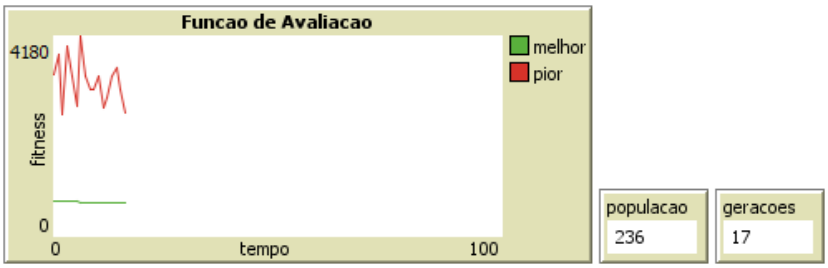

Figura 16: Monitores e plot

Os monitores e *plots* (gráficos) apresentam ao usuário informações pertinentes às simulações do modelo em execução. Neste caso, através dos monitores POPULACAO e GERACOES, é possível visualizar a quantidade de indivíduos do modelo e o progresso das gerações do Algoritmo Genético durante a execução da simulação. No gráfico, pode-se acompanhar o valor da função de avaliação da melhor e da pior solução através do tempo.

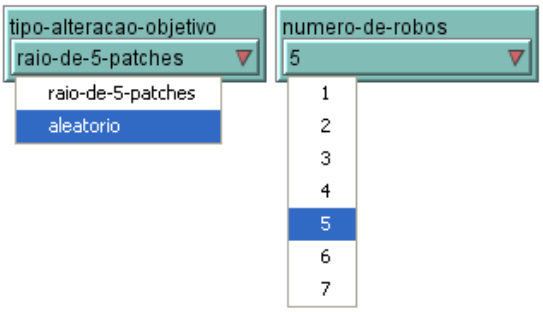

Figura 17: Chooser

O elemento *chooser* permite escolher entre uma série de opções, semelhante a um menu *dropdown* de uma página web. No elemento NUMERO-DE-ROBOS, pode-se definir a quantidade de robôs que serão empregados nas simulações do modelo - de 1 a 7 robôs de acordo com o tamanho pré-definido do cenário e a distância determinada entre eles. Já o elemento TIPO-ALTERACAO-OBJETIVO determina como será realizada a alteração da posição final ou estado objetivo no cenário, se está opção estiver habilitada. Em caso afirmativo, é possível modificar a localização do estado objetivo aleatoriamente em um raio de até cinco *patches* ou por todo o cenário. Esta modificação pode ocorrer a cada ciclo da simulação.

Figura 18: Botões switch

E por fim, têm-se os botões *switch* (chaves de liga-desliga), responsáveis por ativar ou desativar determinada característica do modelo. No modelo constam dois botões *switch*,

ELITISMO? e ALTERAR-OBJETIVO?. O primeiro controla a aplicação ou não de elitismo no Algoritmo Genético. Em caso afirmativo, o melhor indivíduo de cada geração é sempre repassado à geração seguinte. Se o elitismo estiver desativado, existe a possibilidade do melhor indivíduo em uma determinada época não ser repassado à próxima geração. Já o botão *switch* ALTERAR-OBJETIVO? é responsável por ativar e desativar a funcionalidade de modificação da localização do estado objetivo.

Existem outros elementos na *Interface* do NetLogo, porém não são utilizados no modelo proposto.

4.3 O Fluxo da Simulação

No modelo proposto, cada robô possui seu próprio Algoritmo Genético e calcula a sua rota independentemente dos demais robôs presentes no cenário. Não há comunicação nem troca de informações entre os robôs. Após concluir todas as etapas do seu respectivo Algoritmo Genético, cada robô começa a deslocar-se pelo ambiente utilizando a sua rota, ou seja, a melhor solução obtida. No Algoritmo 1, apresenta-se a representação de todas as funcionalidades presentes no modelo proposto.

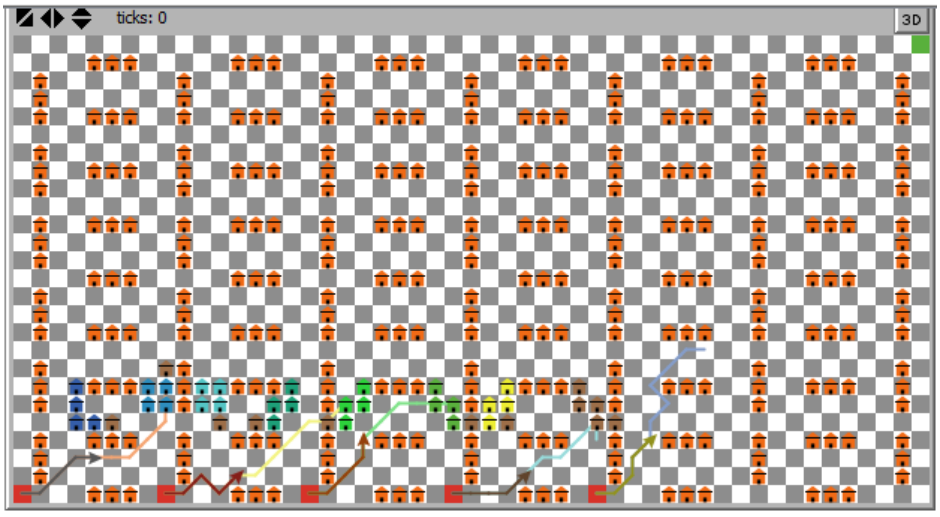

Figura 19: Cálculo da rota através de subtrechos

No modelo inicial, mais simples, uma única rota conectava o ponto de partida e de chegada do cenário. No entanto, no modelo estendido, devido a dimensão ampliada do cenário, um método diferenciado de subtrechos para a geração de rotas foi implementado. Neste caso, cada robô presente no modelo estendido calcula apenas parte de sua respectiva rota, ou seja, um subtrecho. Após movimentar-se pelo cenário usando esta parte da rota, um novo subtrecho é calculado a partir do ponto em que o robô se encontra. Desta forma, cada robô irá calcular pequenos subtrechos interconectados formando a sua rota completa para alcançar o estado objetivo.

Na Figura 19, pode-se observar o método de cálculo de rotas através de subtrechos

Algoritmo 1 Algoritmo do Modelo Evolucionário de Planejamento de Rotas de Robôs

```
repeat
  repeat
    Gere randomicamente a população de indivíduos
    Execute os movimentos presentes em cada cromossomo
    Avalie o fitness de cada indivíduo ao final da rota (subtrecho)
    repeat
      repeat
        Selecione os pais através da Seleção por Torneio
        Divida em dois, em uma posição randômica, cada um dos cromossomos dos pais
        Gere dois novos indivíduos através da mescla do cromossomo dos pais
      until Taxa de Crossover seja alcançada
      if Elitismo está habilitado then
        Selecione o melhor indivíduo
        Transfira-o para a geração subsequente
      end if
      Copie os indivíduos remanescentes para a próxima geração
      Descarte a população antiga
      Execute a mutação de acordo com a taxa definida
      Execute os movimentos definidos em cada cromossomo
      Avalie o fitness de cada indivíduo ao final da rota
    until Número de gerações seja atingido
  until Número de robôs que ainda não alcançaram o estado objetivo seja atingido
  repeat
    Aplique o operador de refinamento na melhor rota
  until A melhor rota de todos os robôs seja otimizada
  while Execute a melhor rota de cada robô do
    Verifique a presença de obstáculos a cada movimento de cada robô
    if Um obstáculo for encontrado then
      Recalcule a rota utilizando os processos do Algoritmo Genético para o robô específico
    end if
  end while
until Todos os robôs atinjam o estado objetivo
```

em ação. Neste exato instante, os cinco robôs presentes no ambiente já calcularam seus respectivos subtrechos que estão visíveis no cenário por meio de linhas com cores claras. Estas linhas são utilizadas para facilitar a visualização das rotas e são empregadas para que o observador da simulação possa prever os próximos movimentos robôs no cenário. As cores mais escuras representam o trecho da rota que já foi percorrida pelos robôs, enquanto que a posição atual de cada robô é identificada pela seta no ambiente.

Este método de cálculo da rota através de subtrechos é aplicado em todos os robôs presentes nas simulações e reduz o tempo de cálculo da rota de cada robô, uma vez que o tamanho do subtrecho é relativamente pequeno, aproximadamente 10 movimentos (este valor pode ser modificado através da interface do modelo). Além disso, o cálculo de uma rota única e completa que fosse capaz de conectar o ponto inicial e o ponto final poderia ser inútil, visto que o ambiente é dinâmico e composto por uma vasta quantidade de obstáculos móveis, resultando em uma alta probabilidade de colisão dos robôs com estes obstáculos. Ao adotar o cálculo de uma rota única, além de consumir um longo tempo em sua execução, os robôs poderiam necessitar prematuramente do recálculo da rota, já que a presença dos obstáculos em meio às rotas planejadas é quase iminente.

Em um ambiente estático apenas o cálculo de uma única rota seria suficiente. Porém, o problema consiste em um ambiente estático e dinâmico, onde os obstáculos podem se movimentar livremente pelo ambiente, afetando diretamente a rota dos robôs. Desta forma, foram implementados procedimentos de detecção de obstáculos para evitar colisões dos robôs com os mesmos. Portanto, a determinação da melhor rota para os robôs, com ponto de partida e de chegada definidos, em um ambiente estático e dinâmico, consiste na busca pela rota mais breve e sem colisões, contendo ainda, a capacidade de identificação de obstáculos presentes no caminho e de recálculo da rota para alcançar o estado objetivo.

4.4 Movimentação dos obstáculos

Inicialmente, os robôs projetam a melhor rota para cada um baseando-se em um ambiente estático. Ao concluir o processo, os robôs começam a executar suas respectivas rotas, ao mesmo tempo em que os obstáculos começam a se movimentar, caracterizando um ambiente dinâmico. Os movimentos são síncronos, ou seja, os movimentos executados pelo robô são alternados com os movimentos realizados pelos obstáculos.

Os obstáculos movimentam-se como uma espécie de serpente, onde sua cabeça é identificada pela cor marrom e determina a direção do movimento dos obstáculos, sendo seguida pelas demais partes do corpo. A cabeça seleciona um dos *patches* vizinhos (vizinhança de von Neumann) para deslocar-se. Se o *patch* selecionado já estiver sendo ocupado por um agente, a função é chamada novamente para a escolha de um novo vizinho. A função é executada recursivamente até que um vizinho disponível seja encontrado. Satisfeita esta condição, a cabeça move-se para este *patch* e o restante dos obstáculos

acompanham o movimento.

Com múltiplos obstáculos móveis e diversos obstáculos fixos no cenário, existe a possibilidade destes obstáculos ficarem presos em certas circunstâncias, tentando encontrar um lugar vazio para deslocar-se sem sucesso. Para evitar esta situação, um procedimento foi implementado para desativar o obstáculo quando não houver *patches* vazios para efetuar um movimento.

4.5 Recálculo da Rota

Em um ambiente complexo, composto por uma grande quantidade de obstáculos fixos e móveis cujo deslocamento pelo cenário é completamente aleatório, a probabilidade de colisão entre os agentes é relativamente alta. Muitas vezes, as rotas encontradas inicialmente pelos robôs não conseguem alcançar o estado objetivo sem a intervenção dos obstáculos móveis ao longo do trajeto, visto que o cenário muda a cada movimento executado pelo robô.

Desta forma, é imprescindível a realização de uma verificação constante da proximidade dos obstáculos com as rotas em questão. Conforme mencionado anteriormente, os movimentos dos obstáculos móveis e dos robôs são efetuados de forma intercalada no cenário. Os movimentos dos obstáculos móveis são guiados por sua "cabeça" e não são concretizados em locais que estejam previamente ocupados. Sendo assim, os obstáculos móveis não são capazes de colidir com os robôs. Estes obstáculos naturalmente desviam tanto dos robôs, quanto dos demais obstáculos fixos e móveis.

No entanto, os robôs podem colidir com os obstáculos, tanto fixos quando móveis. Adotar a melhor rota obtida através da execução do Algoritmo Genético não significa encontrar a rota perfeita ou ótima, completamente livre de obstáculos. A abordagem baseada em Algoritmos Genéticos é um modelo estocástico, baseado em processos evolutivos e com um conjunto de soluções inicializadas aleatoriamente.

Apesar das penalizações implementadas na função de avaliação, em algumas circunstâncias, a melhor rota obtida pelo modelo pode ser aquela que leva o robô diretamente ao estado objetivo, mesmo que ela atravesse um obstáculo fixo no cenário. Esta rota inapropriada pode ter sido escolhida em detrimento de outra rota mais segura, uma vez que esta rota válida não atinge o estado objetivo em um primeiro momento. Estas situações raramente foram presenciadas durante as simulações e testes do modelo proposto.

Já a colisão com os obstáculos móveis é muito mais recorrente. Para impedir que isto aconteça, cada robô realiza uma verificação no local de destino de sua movimentação. Se algum obstáculo for detectado nesta região, o robô interrompe a execução dos movimentos presentes em seu cromossomo e procede com o recálculo da rota, afim de evitar a colisão com o obstáculo.

No recálculo da rota, são realizados os mesmos procedimentos descritos anterior-

mente no capítulo referente ao modelo evolucionário. Gera-se a população inicial, executam-se os movimentos presentes em cada cromossomo, avaliam-se a qualidade das soluções geradas aleatoriamente e executam-se os processos de seleção, reprodução, elitismo e mutação por um determinado número de gerações. Ao final do processo, a melhor rota obtida para o robô em questão é submetida ao operador de refimanento e empregada novamente no deslocamento do robô, substituindo a rota antiga.

O recálculo da rota pode ser executado diversas vezes ao longo do trajeto de cada robô - a cada novo obstáculo encontrado pelo caminho, um novo recálculo de rota deverá ser efetuado. Desta forma, evitam-se as colisões e obtém-se a melhor rota para todos os robôs. Ao atingir o estado objetivo, o cromossomo é editado, removendo-se todos os genes excedentes que se encontram após o gene que atingiu o estado objetivo. Se um cromossomo composto por 50 genes atingiu o estado objetivo no gene 48, os genes 49 e 50 são descartados. Uma vez que todos os robôs atingiram o estado objetivo, a simulação encerra a sua execução.

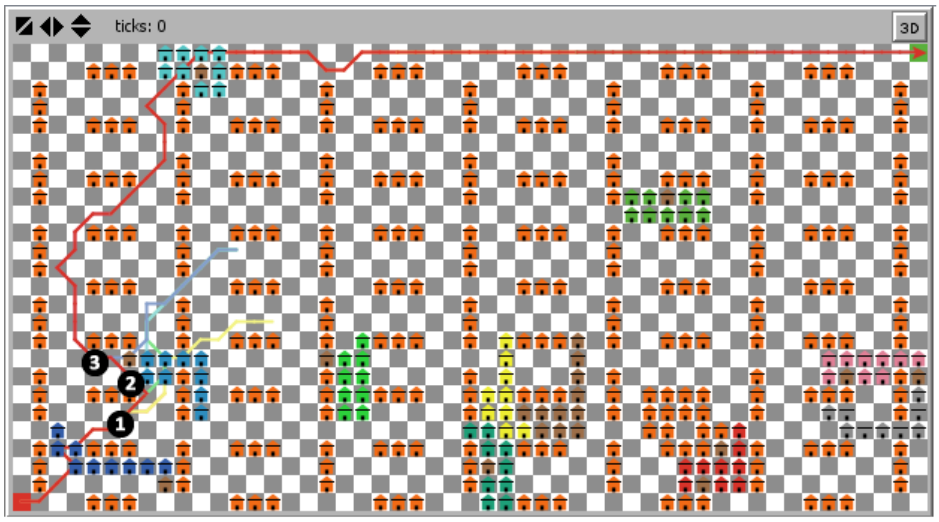

Figura 20: Solução final gerada pelo Algoritmo Genético após 3 recálculos de rota

No exemplo apresentado pela Figura 20, pode-se verificar o encontro entre o robô e os obstáculos móveis em três ocasiões distintas - pontos pretos numerados em 1, 2 e 3. Nestes pontos específicos, a rota foi recalculada para evitar a colisão do robô com os obstáculos. Em cada recálculo da rota, o robô encontra um novo caminho livre de obstáculos. No entanto, nos dois primeiros recálculos, os obstáculos eventualmente acabaram cruzando novamente a trajetória do robô, forçando-o a procurar por uma nova rota. Após o terceiro recálculo, o robô encontrou uma rota e finalmente atingiu o estado objetivo sem colidir com nenhum obstáculo pelo caminho.

Durante a simulação, são impressos no gráfico os valores da função de avaliação do melhor e do pior indivíduo da população de cada robô em função do tempo. Enquanto o gráfico gerado pelos piores indivíduos ao longo das gerações apresenta um aspecto altamente oscilatório, com grandes variações no valor da função de avaliação, o gráfico

Figura 21: Gráfico apresentando a melhor e a pior solução após 20 gerações

gerado pelos melhores indivíduos ao longo do tempo exibe um leve decrescimento no valor da função de avaliação, por vezes atingindo uma estabilidade em determinado ponto da simulação.

5 SIMULAÇÕES E ANÁLISES DOS RESULTADOS

Após a construção do modelo proposto baseado na abordagem de Algoritmos Genéticos e da elaboração dos cenários, foram realizados testes de ajuste para configurar os operadores e os demais elementos presentes no modelo, tais como taxa de reprodução, taxa de mutação, quantidade de indivíduos na população, número de gerações, elitismo e valores de penalização e bonificação da função de avaliação.

5.1 Abordagem do Algoritmo A*

Para avaliar a viabilidade da abordagem de Algoritmos Genéticos para o planejamento de rotas de robôs móveis utilizou-se o Algoritmo A* (A Star) para a realização dos testes comparativos de qualidade das soluções. O Algoritmo A* foi desenvolvido por Peter Hart, Nils Nilsson e Bertram Raphael em 1968 e é amplamente empregado em problemas de planejamento de rotas e grafos transversais devido a sua alta performance e pelo algoritmo ser completo e ótimo (RUSSEL; NORVIG, 2003).

O Algoritmo A* avalia os nodos combinando *g(n)*, o custo para alcançar um nodo, e *h(n)*, o custo para deslocar-se deste nodo até o estado objetivo.

$$f(n) = g(n) + h(n), \tag{2}$$

Na Equação 2, *h(n)* deve ser uma heurística admissível, ou seja, nunca superestimar o custo para atingir o estado objetivo. Então, *f(n)* é o custo estimado da solução com menor custo que atravessa *n* (RUSSEL; NORVIG, 2003).

Na busca pela solução ótima, o Algoritmo A* explora os nodos promissores. Após escolher um nodo baseando-se na Equação 2, o A* explora este nodo para encontrar o vizinho com o menor *f(n)*. Na sequência, o algoritmo escolhe este nodo (o melhor vizinho do nodo anterior) e explora-o novamente, procurando por seus vizinhos. Este processo continua até que o Algoritmo A* encontre a melhor solução para o problema.

5.2 Abordagem de Algoritmo Genético

O Algoritmo Genético utilizado nos testes de desempenho foi configurado da seguinte forma: população de 236 indivíduos, 85% de taxa de crossover, 20% de taxa de mutação e elitismo ativado. Durante a simulação, o Algoritmo Genético é executado por 22 gerações para calcular cada subtrecho da rota de cada robô. Estes parâmetros foram adotados após uma série de testes empíricos em um único robô, aonde diferentes valores foram empregados e testados para cada propriedade. A taxa de crossover foi testada entre o intervalo de 60% e 90% e a taxa de mutação entre 5% e 30%. O tamanho da população variou entre 100 e 500 indivíduos, enquanto que o número de gerações variou entre 20 e 50.

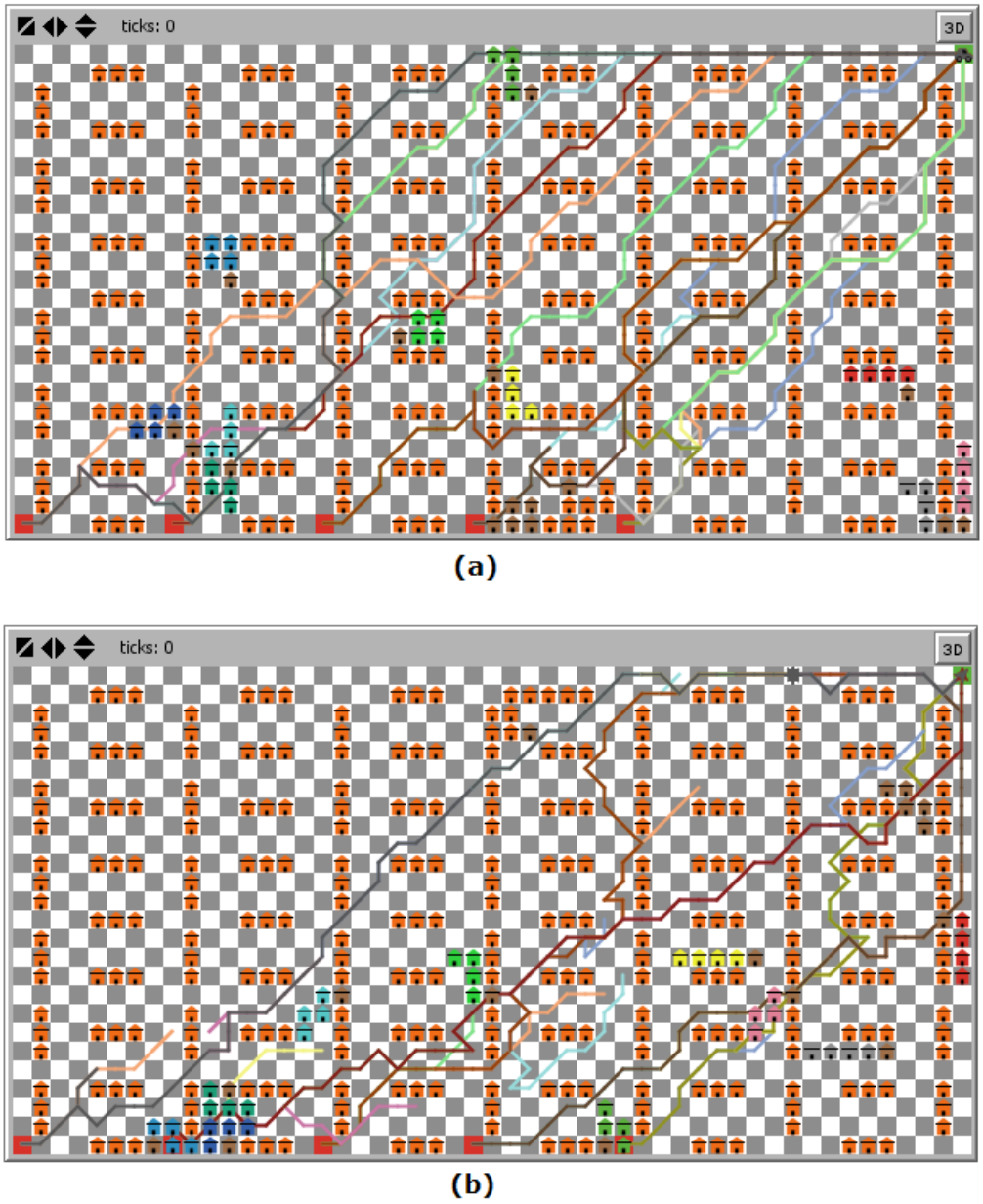

Figura 22: Exemplos de simulações do Algorimo A* (a) e do Algoritmo Genético (b)

Na Figura 22, pode-se observar o resultado final de uma simulação das abordagens adotadas nos testes comparativos de qualidade. Diferentemente da abordagem de Algoritmos Genéticos, o A* efetua o cálculo completo da rota, a partir do ponto em que o robô se encontra até o estado objetivo.

O método de detecção de obstáculos para evitar colisões também foi adotado nesta abordagem. Ainda na imagem referente ao A*, as rotas marcadas pelas cores mais claras representam as rotas inicialmente calculadas para os robôs, mas que acabaram não sendo utilizadas pela necessidade do recálculo devido ao encontro do robô com algum obstáculo no cenário durante a execução de seus movimentos.

Na Figura 22(b), percebe-se que as rotas com as cores mais claras não executadas pelos robôs são menores, devido ao método de cálculo das rotas restrito a subtrechos.

5.3 Simulações

Ao todo, foram realizadas 20 simulações em um cenário com a seguinte configuração: dimensão do cenário definida em 50 x 25; 96 obstáculos fixos (cada obstáculo fixo possui tamanho 3, ou seja, é composto por 3 casas laranjas); 12 obstáculos móveis (cada um com tamanho 5); e 5 robôs móveis. Como parâmetro de comparação qualitativa dos dois modelos, fez-se uso do tamanho da rota - o número de movimentos empregados pelos robôs para atingir o estado objetivo.

A Tabela 2 apresenta os resultados obtidos durante os testes comparativos entre o Algoritmo A* e o Algoritmo Genético. São apresentados o número de movimentos executados por cada robô ao longo das simulações, assim como a média e o desvio padrão. O número sobrescrito (ex.: o número 1 em 50^{1}) representa o número de recálculos efetuados durante a respectiva simulação. Cada recálculo significa um desvio de obstáculos efetuado pelo robô.

De acordo com a sistemática adotada para testes e sua quantidade, a técnica estatística mais apropriada selecionada para a análise dos resultados foi o Teste t de Student[1]. Nesta técnica utiliza-se apenas uma variável, que neste caso é o número de movimentos executados pelos robôs, aonde são avaliadas as diferenças entre as médias de dois grupos.

Para facilitar a realização das simulações, utilizou-se uma funcionalidade do software NetLogo capaz de executar uma sequência de simulações automaticamente. Esta funcionalidade, intitulada de *Behavior Space*, pode executar diversas simulações por um número determinado de vezes, utilizando ainda o recurso de simulações simultâneas dependendo do número de processadores do computador em questão. A coleta de informações sobre as simulações, tais como número de movimentos executados por cada robô e número de recálculos, também fez uso de funcionalidades específicas da ferramenta de desenvolvimento. Foram empregadas funções para a geração de *screenshots* e arquivos .csv com

[1] http://www.graphpad.com/quickcalcs/ttest1/?Format=C

Tabela 2: Medida da quantidade de movimentos executados nas rotas

N	Robô 1		Robô 2		Robô 3		Robô 4		Robô 5	
	A*	**AG**	**A***	**AG**	**A***	**AG**	**A***	**AG**	**A***	**AG**
01	54^{1}	59^{1}	49^{0}	50^{3}	55^{2}	50^{5}	51^{6}	40^{0}	31^{0}	31^{1}
02	57^{0}	61^{1}	48^{1}	49^{0}	39^{0}	43^{1}	32^{1}	39^{1}	37^{4}	35^{0}
03	56^{1}	70^{3}	57^{6}	51^{1}	40^{3}	39^{0}	35^{4}	35^{2}	32^{1}	33^{2}
04	65^{10}	59^{0}	51^{1}	55^{6}	52^{2}	42^{0}	35^{1}	43^{0}	30^{1}	35^{1}
05	57^{1}	66^{0}	48^{1}	55^{1}	39^{2}	48^{2}	39^{6}	41^{8}	31^{0}	43^{3}
06	60^{2}	66^{4}	50^{2}	50^{0}	44^{1}	46^{0}	59^{3}	36^{0}	40^{4}	33^{0}
07	55^{1}	61^{0}	60^{5}	59^{3}	43^{1}	58^{6}	38^{1}	36^{0}	31^{2}	42^{11}
08	60^{2}	58^{0}	46^{1}	59^{1}	47^{2}	50^{1}	33^{1}	55^{6}	31^{0}	39^{2}
09	66^{2}	75^{7}	49^{2}	49^{0}	40^{0}	40^{0}	32^{1}	42^{1}	31^{1}	40^{4}
10	57^{1}	61^{8}	55^{1}	52^{1}	46^{4}	52^{3}	33^{0}	46^{1}	31^{1}	33^{0}
11	56^{0}	68^{8}	68^{5}	60^{3}	56^{5}	60^{8}	37^{6}	49^{7}	34^{1}	33^{0}
12	57^{1}	67^{10}	52^{5}	54^{1}	40^{2}	40^{0}	33^{0}	35^{0}	32^{1}	32^{1}
13	64^{5}	73^{7}	48^{0}	48^{0}	43^{2}	50^{2}	35^{2}	36^{1}	32^{4}	42^{1}
14	57^{3}	69^{4}	48^{0}	59^{2}	58^{3}	48^{3}	41^{2}	38^{0}	31^{1}	32^{3}
15	52^{1}	65^{2}	48^{1}	51^{0}	51^{3}	48^{3}	38^{4}	32^{2}	42^{6}	36^{0}
16	61^{2}	66^{2}	49^{2}	50^{1}	42^{1}	42^{0}	32^{1}	36^{1}	31^{3}	32^{1}
17	57^{2}	56^{0}	51^{3}	60^{1}	41^{2}	54^{5}	45^{3}	45^{2}	44^{4}	37^{1}
18	58^{1}	66^{1}	62^{7}	55^{6}	44^{3}	43^{1}	37^{5}	41^{0}	34^{3}	52^{13}
19	59^{1}	59^{1}	51^{1}	61^{1}	45^{1}	40^{0}	33^{1}	41^{0}	31^{1}	35^{0}
20	56^{3}	64^{2}	59^{2}	50^{1}	40^{0}	44^{1}	42^{5}	36^{0}	31^{0}	42^{2}
Média	58,2	64,45	52,45	53,85	45,25	46,85	38	40,1	33,35	36,85
DP	3.59	5.15	5.84	4.42	6.01	6.05	6.97	5.53	4.09	5.31

todos os dados pertinentes ao problema.

5.4 Resultados

Nos testes de desempenho comparativos realizados com os robôs 1 e 5, o Algoritmo A* apresentou uma diferença estatística significante comparado ao Algoritmo Genético. Através do Teste t de Student, confirmou-se com 95% de confiança de que o A* foi superior ao Algoritmo Genético nos testes realizados.

No entanto, nos testes de desempenho executados com os robôs 2, 3 e 4, localizados na parte central do cenário, a diferença entre o Algoritmo A* e o Algoritmo Genético não foi considerada estatisticamente significativa com 95% de confiança.

Através da abordagem de Algoritmos Genéticos, foi possível modelar soluções para o problema através dos cromossomos compostos pelos movimentos dos robôs no ambiente. A partir de soluções geradas arbitrariamente, os processos de seleção, reprodução, mutação, elitismo e refinamento, baseados na função de avaliação de cada indivíduo, foram capazes de evoluir as candidatas à solução do problema, atingindo uma solução satisfatória para cada uma das simulações. Após o desenvolvimento dos modelos propostos e da análise dos resultados obtidos, conclui-se que a abordagem de Algoritmos Genéticos apresentou-se com uma alternativa viável para a solução do problema em questão.

Apesar da constatação da superioridade do A* em relação ao Algoritmo Genético nos testes desempenho realizados com os robôs 1 e 5, pode-se notar a similaridade de performance nos resultados obtidos com os robôs 2, 3 e 4, aonde um algoritmo não-ótimo como o Algoritmo Genético equiparou-se a um algoritmo ótimo que é amplamente utilizado no problema de planejamento de rotas.

Este é um grande progresso em relação aos resultados obtidos na primeira versão do modelo abordada em (CABREIRA; DIMURO; AGUIAR, 2012), aonde apesar dos resultados desfavoráveis apresentados pela abordagem do Algoritmo Genético em relação ao Algoritmo A*, já tinha-se indícios da melhora dos resultados do Algoritmo Genético conforme a complexidade do cenário crescia. Além disso, a abordagem de Algoritmos Genéticos poderá ser testada e avaliada em ambientes com informação incompleta, aonde o robô pode apresentar uma visão parcial ou limitada do cenário em questão. No primeiro estudo, três cenários diferentes foram considerados: um cenário sem obstáculos, um cenário contendo apenas obstáculos fixos e um cenário com obstáculos fixos e móveis. Em todos estes cenários, o robô possuía informação completa, ou seja, detinha conhecimento da posição dos obstáculos e do estado objetivo em qualquer parte do ambiente.

O desenvolvimento do presente trabalho possibilitou a publicação de artigos científicos em alguns eventos e congressos. Em anexo, encontra-se o relatório de publicações resultantes deste trabalho. Todas as publicações e os modelos do NetLogo estão disponíveis para download em http://minerva.ufpel.edu.br/~marilton/CabreiraTM.

6 CONCLUSÃO E TRABALHOS FUTUROS

O presente trabalho propõe uma abordagem evolucionária baseada em Algoritmos Genéticos para o problema de planejamento de rotas de múltiplos robôs móveis em ambientes complexos e dinâmicos. O modelo proposto utiliza os operadores tradicionais de seleção, reprodução, mutação e elitismo, além de um operador de refinamento especialmente desenvolvido para o problema em questão. A função de avaliação, classificada como uma função de minimização, é responsável por avaliar qualitativamente o conjunto de soluções a cada geração, levando em consideração a distância percorrida pelos robôs, penalizações inferidas por eventuais cruzamentos das rotas geradas por obstáculos presentes no cenário e bonificações conquistadas a partir do alcance do estado objetivo.

O modelo proposto foi implementado na plataforma de desenvolvimento de aplicações multiagentes, NetLogo. Através do uso deste software, possibilitou-se a construção de diferentes tipos de cenários compostos por obstáculos fixos e móveis. Com relação aos obstáculos móveis, a concepção de seus movimentos aleatórios através do ambiente foi viabilizada graças às funcionalidades presentes no NetLogo, que permitem a modelagem de agentes com diversas propriedades, incluindo o deslocamento. Os cenários incluem ainda os pontos de partida e de chegada, identificados pelas cores vermelho e verde. Os robôs, também modelados como agentes no ambiente, são inicialmente posicionados em seus respectivos pontos de partida. A partir da obtenção da sua rota através dos processos do Algoritmo Genético, os robôs começam a deslocar-se pelo ambiente em busca do estado objetivo, comum a todos.

A utilização do software NetLogo foi de suma importância neste quesito, uma vez que possibilitou a modelagem de um ambiente totalmente dinâmico composto por obstáculos móveis, onde os múltiplos robôs são capazes de detectar a presença de obstáculos. Sem esta propriedade, seria inviável a prevenção de possíveis colisões. Através da interação entre os agentes do ambiente, os robôs estão aptos a perceber os obstáculos localizados em seu caminho e recalcular a sua trajetória para alcançar o estado objetivo.

O modelo de Algoritmo Genético empregado neste trabalho utiliza, além dos operadores tradicionais, operadores de refinamento que aprimoram os resultados da abordagem. A partir destes operadores de refinamento, é possível eliminar movimentos realizados

desnecessariamente pelos robôs, que não modificam a sua posição atual em relação ao estado objetivo. Desta forma, reduz-se o número genes presentes em cada cromossomo, ou seja, o número de movimentos efetuados por cada robô.

Além disso, este modelo baseado na abordagem de Algoritmo Genético emprega um método de cálculo de rotas por partes (subtrechos), reduzindo drasticamente o tempo de processamento de rotas para cada robô. Com este método, calcula-se apenas um pequeno trecho da rota dos robôs, visto que o ambiente composto por diversos obstáculos contém uma alta probabilidade de cruzamento de obstáculos pela rota gerada. Calcular uma rota longa e completa, que leve o robô de sua posição atual até o estado objetivo, poderia ocasionar em uma perda de tempo considerável. Além de consumir maior tempo na obtenção da melhor solução, a rota encontrada poderia rapidamente tornar-se inútil com a detecção da presença de um obstáculo em sua trajetória.

Portanto, considerando a abordagem adotada com o cálculo individual de rotas por partes, a interação entre robôs e obstáculos através de sua modelagem baseada em agentes, a capacidade de detecção de obstáculos e o aperfeiçoamento das soluções geradas através dos operadores de refinamento, o modelo introduzido nesta dissertação aponta para uma solução alternativa e promissora para o problema de planejamento de rotas de múltiplos robôs em um contexto multiagente.

Sendo assim, em trabalhos futuros, o cálculo de rotas por partes pode ser muito útil em ambientes com informação incompleta. Em cenários com esta configuração, o Algoritmo A* dificilmente encontraria uma solução válida e o Algoritmo Genético poderia ser testado para a avaliação de sua performance. Como uma técnica robusta e flexível, os Algoritmos Genéticos também poderiam ser combinados com diferentes técnicas afim de aprimorar os resultados obtidos.

Conceitos de Lógica *Fuzzy* empregar-se-ão na detecção da presença de obstáculos que possam colidir com os robôs durante a execução de suas rotas. Através da Lógica *Fuzzy* podem-se utilizar dois atributos linguísticos, a distância e a evidência, para modelar a detecção da presença de obstáculos no ambiente e a análise de uma possível colisão com os mesmos. Com um grau de pertinência variando de 0 a 1, o atributo "distância" pode ser medido através de termos linguísticos, tais como: muito perto, perto, mediano, longe e muito longe. Cada uma destas medidas estaria enquadrada em um intervalo de valores númericos, ainda a serem definidos, que representam a distância real entre o robô e os obstáculos.

O segundo atributo seria o da "evidência", empregado pelo robô para tentar distinguir se uma determinada imagem captada pelo sensor é ou não um obstáculo. Este atributo também seria medido através de termos linguísticos, tais como: muito fraco, fraco, médio, forte e muito forte. Estas medidas também seriam enquadradas em intervalos numéricos, onde cada intervalo seria representado por uma tonalidade variando da cor branca à cor preta. Na sequência, a base de regras de inferência poderia ser construída com base

em todas as combinações entre essas variáveis linguísticas, que determinariam, após a defuzzificação, o grau de certeza para o recálculo ou não da rota do robô.

REFERÊNCIAS

BHADURI, A. A mobile robot path planning using Genetic Artificial Immune Network algorithm. In: WORLD CONGRESS ON NATURE AND BIOLOGICALLY INSPIRED COMPUTING, 2009, Los Alamitos. **Proceedings...** IEEE, 2009. p.1536–1539.

CABREIRA, T.; AGUIAR, M. de; DIMURO, G. An Extended Evolutionary Learning Approach for Multiple Robot Path Planning in a Multi-agent Environment. In: IEEE CONFERENCE ON EVOLUTIONARY COMPUTATION, 2013., 2013, Los Alamitos. **Anais...** IEEE, 2013. v.1, p.3363–3370.

CABREIRA, T. M.; DIMURO, G. P.; AGUIAR, M. S. An Evolutionary Learning Approach for Robot Path Planning with Fuzzy Obstacle Detection and Avoidance in a Multi-agent Environment. In: SOCIAL SIMULATION (BWSS), 2012 THIRD BRAZILIAN WORKSHOP ON, 2012, Los Alamitos. **Anais...** IEEE, 2012. p.60–67.

CANNY, J.; REIF, J. New lower bound techniques for robot motion planning problems. In: IEEE SYMPOSIUM ON THE FOUNDATIONS OF COMPUTER SCIENCE, 1987, Los Alamitos. **Proceedings...** IEEE, 1987. p.49–60.

GOLDBERG, D. **Genetic algorithms in search, optimization and machine learning**. Delhi, India: Addison-Wesley, 2000.

HACHOUR, O. The proposed genetic FPGA implementation for path planning of autonomous mobile robot. **International Journal of Circuits, Systems and Signal Processing**, Sofia, Bulgária, v.2, n.2, p.151–167, 2008.

HACHOUR, O. Path planning of autonomous mobile robot. **International Journal of Systems Applications, Engineering and Development**, Sofia, Bulgária, v.4, n.2, p.178–190, 2008.

HOLLAND, J. **Adaptation in natural and artificial systems**. Ann Arbor, USA: University of Michigan Press, 1975.

HU, Y.; YANG, S. A knowledge Based Genetic Algorithm for Path Planning of a Mobile Robot. In: INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, 2004., 2004, Nova Orleans, Los Angeles. **Proceedings...** IEEE, 2004.

JANSEN, M. R.; STURTEVANT, N. R. Direction maps for cooperative pathfinding. In: AAAI CONFERENCE ON ARTIFICIAL INTELLIGENCE AND INTERACTIVE DIGITAL ENTERTAINMENT, 2008, Palo Alto, Califórnia. **Anais...** AAAI, 2008.

JANSEN, M. R.; STURTEVANT, N. R. A new approach to cooperative pathfinding. In: INTERNATIONAL CONFERENCE ON AUTONOMOUS AGENTS AND MULTIAGENT SYSTEMS, 2008, Estoril, Portugal. **Anais...** AAMAS, 2008. v.3, p.1401–1404.

KRONTIRIS, A.; SAJID, Q.; BEKRIS, K. E. Towards Using Discrete Multiagent Pathfinding to Address Continuous Problems. In: ASSOCIATION FOR THE ADVANCEMENT OF ARTIFICIAL INTELLIGENCE, 2012, Toronto, Canadá. **Anais...** AAAI, 2012.

LATOMBE, J. **Robot Motion Planning**. Norwell: Kluwer, 1991.

LI, Q.; TONG, X.; XIE, S.; ZHANG, Y. Optimum Path Planning for Mobile Robots Based on a Hybrid Genetic Algorithm. **Sixth International Conference on Hybrid Intelligent Systems**, Los Alamitos, 2006.

LIN, L.; WANG, H.; WU, Q. Improved Genetic Algorithms Based Path Planning of Mobile Robot Under Dynamic Unknown Environment. Louyang. **International Conference on Mechatronics and Automation**, Los Alamitos, 2006.

LOZANO-PEREZ, T. Spatial planning: A Configuration Space Approach. **IEEE Transactions on Computer Sciences**, Los Alamitos, v.32, n.3, p.108–120, 1983.

LOZANO-PEREZ, T.; WESLEY, M. An algorithm for planning collision-free paths among polyhedral obstacles. **Communications of ACM.**, Nova Iorque, v.22, p.560–570, 1979.

LUH, G.; LIU, W. Motion planning for mobile robots in dynamic environments using a potential field immune network. **Journal of Systems and Control Engineering**, Reino Unido, v.221, n.7, p.1033–1045, 2007.

LUH, G.; LIU, W. An immunological approach to mobile robot reactive navigation. **Applied Soft Computing**, Amsterdã, Holanda, v.8, n.1, p.30–45, 2008.

MASEHIAN, E.; KATEBI, Y. Robot motion planning in dynamic environments with moving obstacles and target. **International Journal of Mechanical, Industrial and Aerospace Engineering**, [S.l.], v.1, n.1, p.20–25, 2007.

NEDJAH, N.; COELHO, L.; MOURELLE, L. **Mobile Robots**: The Evolutionary Approach. Berlim: Springer, 2007. (Studies in Computational Intelligence).

RAJA, P.; PUGAZHENTHI, S. Path planning for mobile robots in dynamic environments using particle swarm optimization. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCES IN RECENT TECHNOLOGIES IN COMMUNICATION AND COMPUTING (ARTCOM 2009), 2009, Kottayam, India. **Proceedings...** IEEE, 2009. p.401–405.

RAJA, P.; PUGAZHENTHI, S. Path planning for a mobile robot using real coded genetic algorithm. **International Journal of Advanced Robotic Systems**, Croácia, v.10, n.1, p.27–39, 2009.

RAJA, P.; PUGAZHENTHI, S. Path Planning for a mobile robot in dynamic environments. **International Journal of the Physical Sciences**, EUA, v.6, n.20, p.4721–4731, 2011.

RAJA, P.; PUGAZHENTHI, S. Optimal path planning of mobile robots: A review. **International Journal of Physical Sciences**, EUA, v.7, n.9, p.1314 –1320, feb 2012.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence**: A Modern Approach. [S.l.]: Prentice Hall Series, 2003.

RYAN, M. R. K. Exploiting subgraph structure in multi-robot path planning. **Journal of Artificial Intelligence Research**, El Segundo, Califórnia, p.497–542, 2008.

SADATI, N.; TAHERI, J. Genetic Algorithm in Robot Path Planning in Crisp and Fuzzified Environments. **International Conference on Industrial Technology**, Los Alamitos, 2002.

SHARON, G.; STERN, R.; FELNER, A.; STURTEVANT, N. Meta-Agent Conflict-Based Search for Optimal Multi-Agent Path Finding. In: FIFTH ANNUAL SYMPOSIUM ON COMBINATORIAL SEARCH, 2012, Toronto, Canadá. **Anais...** AAAI, 2012.

SIEGWART, R.; NOURBAKHSH, I. **Introduction to autonomous mobile robot**. Cambridge, U.S.A.: Massachusetts Institute of Technology, 2004.

STANDLEY, T. Independence Detection for Multi-Agent Pathfinding Problems. In: AAAI WORKSHOP ON MULTIAGENT PATHFINIDNG, 2012., 2012, Toronto, Canadá. **Anais...** AAAI, 2012.

SUGIHARA, K.; SMITH, J. Genetic algorithms for adaptive motion planning of an autonomous mobile robot. In: IEEE INTERNATIONAL SYMPOSIUM ON COMPUTATIONAL INTELLIGENCE IN ROBOTICS AND AUTOMATION, 1997, Monterey, California. **Proceedings...** IEEE, 1997. p.138–143.

SURYNEK, P. An application of pebble motion on graphs to abstract multi-robot path planning. In: INTERNATIONAL CONFERENCE ON TOOLS WITH ARTIFICIAL INTELLIGENCE, 2009, Los Alamitos. **Anais...** IEEE, 2009. p.151–158.

TAMILSELVI, D.; SHALINIE, S.; THASNEEM, A.; SUNDARI, S. Optimal Path Selection for Mobile Robot Navigation Using Genetic Algorithm in an Indoor Environment. In: THILAGAM, P.; PAIS, A.; CHANDRASEKARAN, K.; BALAKRISHNAN, N. (Ed.). **Advanced Computing, Networking and Security**. Alemanha: Springer, 2012. p.263–269. (Lecture Notes in Computer Science, v.7135).

TU, J.; YANG, S. Genetic Algorithm Based Path Planning for a Mobile Robot. In: INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, 2003., 2003, Los Alamitos. **Proceedings...** IEEE, 2003.

WANG, K. C.; BOTEA, A. Fast and memory efficient multi-agent pathfinding. In: INTERNATIONAL CONFERENCE ON AUTOMATED PLANNING AND SCHEDULING, 2008, Sidney, Austrália. **Anais...** ICAPS, 2008. p.380–387.

WILENSKY, U. **Netlogo User Manual**. EUA: Universidade de Northwestern, 2012. http://ccl.northwestern.edu/netlogo/docs.

ANEXO A RELATÓRIO DE PUBLICAÇÕES

A seguir encontra-se o relatório contendo todas as publicações geradas pelo projeto com apresentações em congressos e eventos:

- A primeira etapa da pesquisa foi apresentada sob a forma de um resumo no II Seminário de Pesquisa em Computação (SPC), realizado durante a XVI Semana Acadêmica da Computação (SACOMP) da Universidade Federal de Pelotas (UFPel) que ocorreu ao final de 2011.

- Um artigo completo foi submetido e aceito no formato de pôster no VI Workshop-Escola de Sistemas de Agentes, seus Ambientes e Aplicações (WESAAC), realizado na Universidade Federal de Santa Catarina (UFSC), no primeiro semestre de 2012.

- A versão em inglês do trabalho proposto foi submetida e aceita para apresentação oral no III Brazilian Workshop on Social Simulation (BWSS), realizado em Outubro de 2012 na cidade de Curitiba.

- Em Novembro de 2012, um resumo foi apresentado sob a forma de pôster no Congresso Brasileiro de Sistemas Fuzzy (CBSF) em Natal, no Rio Grande do Norte.

- Em Dezembro de 2012, a segunda etapa da pesquisa foi apresentada sob a forma de um resumo no III Seminário de Pesquisa em Computação (SPC), organizado novamente na Universidade Federal de Pelotas (UFPel).

- Em 2013, a versão final do trabalho proposto foi submetida, aceita e apresentada oralmente no Congress on Evolutionary Computation (CEC 2013) em Cancún, México. Trata-se um evento Qualis A2, um dos mais importantes na área da Computação Evolucionária.

- Na mesma época, a versão final também foi aceita no Brazilian Conference on Intelligent Systems (BRACIS 2013). No entanto, foi solicitada a retirada do artigo no evento, devido à publicação no CEC 2013.