

MODELAGEM E IMPLEMENTAÇÃO DE UM MÓDULO PARA EVOLUÇÃO DA MORFOLOGIA E CONTROLE DE ROBÔS MÓVEIS

Ariel Felipe Ferreira Marques, Jesimar da Silva Arantes, Raphael Winckler de Bettio, Tales Heimfarth
arielfmarques@gmail.com, jesimar.arantes@gmail.com, raphaelwb@dcc.ufla.br, tales.heimfarth@gmail.com

Universidade Federal de Lavras
Lavras, Minas Gerais

Categoria: ARTIGO SUPERIOR / MULTIMÍDIA



Resumo: O presente artigo apresenta uma ferramenta computacional capaz de projetar o modelo físico e também os algoritmos de controles de robôs móveis com o objetivo de automatizar o processo de projeto e programação dos mesmos. Para atingir este objetivo a plataforma de simulação de robôs móveis *GrubiBots* (ferramenta desenvolvida e mantida pelo Laboratório de Pesquisa *GRUBi*) foi munida de um módulo capaz de utilizar programação genética para gerar os algoritmos de controle e um módulo baseado em algoritmos genéticos capaz de definir o modelo físico (morfologia) do robô. Para validar a proposta foram propostos dois cenários de teste. Em um dos cenários foi avaliada a capacidade do modelo de projetar um robô capaz de se deslocar linearmente em um ambiente sem obstáculos, já no segundo cenário foi avaliada a capacidade de projetar um robô que percorra um ambiente contendo obstáculos maximizando o espaço percorrido e minimizando o número de colisões. Os resultados apresentados demonstram a viabilidade do modelo.

Palavras Chaves: Robótica, Algoritmos Genéticos, Desenvolvimento Automatizado.

Abstract: *This paper presents a computational tool capable of designing the morphology model and also the controls algorithms of mobile robots with the goal to automate the entire creation process. To achieve this goal the simulation platform of mobile robots GrubiBots (tool developed and maintained by Grubi Research Laboratory) was provided with a module capable of generate algorithms for control using genetic programming and another module based on genetic algorithms able to set the physical model (morphology) of the robot. To validate the model was proposed two scenarios. In the first scenario was evaluated the ability to design a robot capable of moving linearly in an environment without obstacles, in the second was evaluated the ability to design a robot capable of drives in an environment containing obstacles maximizing the distance traveled and minimizing the number of collisions. The presented results demonstrate the feasibility of the model.*

Keywords: Robotics, Genetic Algorithms, Automatic Development.

1 INTRODUÇÃO

Este trabalho apresenta o desenvolvimento de um sistema de

robótica evolutiva cujo propósito é a evolução do projeto morfológico e comportamental de robôs móveis. A evolução do projeto morfológico é baseado em uma série de peças pré-definidas e um chassi onde essas peças podem ser encaixadas, já os comportamentos são desenvolvidos baseando-se em uma estrutura de programação inspirada no modelo de subsunção.

Este trabalho encontra-se organizado da seguinte forma: a seção 2 apresenta o trabalho relacionado. A seção 3 descreve os conceitos básicos acerca do trabalho. A seção 4 apresenta a definição do problema. Nas seções 5 e 6 é apresentada a visão geral da morfologia e do sistema de controle do robô, respectivamente. Na seção 7 são apresentados os resultados, e as conclusões são apresentadas na seção 8.

2 TRABALHOS RELACIONADOS

É possível encontrar na literatura uma série de estudos relacionados com o uso de algoritmos genéticos cujo propósito é a modelarem automática de robôs móveis. Estes estudos podem ser agregados em três principais categorias: evolução comportamental (também denominada controle), evolução do corpo (também denominada evolução da morfologia) e evolução das duas características em conjunto.

Este trabalho encaixa-se na categoria que tem por objetivo evoluir a morfologia e o controle de robôs ao mesmo tempo. Pode-se citar a pesquisa de Karl Sims [Sims, 1994] como uma das primeiras tentativas de exploração da área, nesta pesquisa o autor descreve um processo capaz de otimizar o modelo físico e o sistema de controle de robôs (denominados pelo autor de criaturas) através de algoritmos genéticos, nesta pesquisa as criaturas tinham por objetivo movimentar-se em um ambiente simulando movimentos de criaturas reais como caminhar, nadar, pular entre outras.

Em outra pesquisa do mesmo autor [Sims (b), 1994] as criaturas criadas tinham como propósito sobreviver através da competição por alimento. Como resultado as criaturas desenvolveram estratégias e contra estratégias capazes de garantir sua sobrevivência.

As duas pesquisas apesar de criarem criaturas com propósitos diferentes tem uma premissa básica similar, o corpo das criaturas é formado de uma série de objetos rígidos interligados em uma hierarquia e cuja ligação é articulada, sendo que existem vários tipos articulações possíveis. O

controle das criaturas é obtido através do que o autor chamou de Internal Neural Nodes capazes de processar diferentes dados vindos de sensores e produzir efeitos nos atuadores.

O trabalho de Sims deu origem a diversas outras pesquisas cujo objetivo era testar a evolução de criaturas criadas com diferentes propósitos e que também testaram diferentes métodos para criação da morfologia e do controle. Um novo estágio que pode ser verificado nas pesquisas atuais é a evolução destas criaturas de maneira que as mesmas possam ser montadas no mundo real. Para que esse objetivo possa ser alcançado é necessário que os mecanismos básicos utilizados na evolução da morfologia possam ser materializados como na proposta apresentada por Lipson e Pollack [Lipson, 2000]. Nesta pesquisa os autores demonstraram que é possível evoluir as criaturas de modo virtual e então materializá-las através de *rapid manufacturing technology*.

Os trabalhos citados até o momento tem por objetivo criar criaturas virtuais, entretanto, as mesmas ideias podem ser utilizadas na construção de robôs autônomos móveis que utilizem formas tradicionais de movimentação (rodas) e sensoriamento (sensores de distância infravermelho ou sonar).

Em outro trabalho [Chocron, 1999] definiu quatro mecanismos básicos para a morfologia, sendo eles *joins*, *segments*, *wheels* e *base*. Utilizando-se a combinação destes mecanismos foi possível evoluir a morfologia do robô para atingir mais de um objetivo. Esta pesquisa propôs dois objetivos finais sendo que nenhum dos dois necessitava de coleta de dados do ambiente. O objetivo com maior relação com esta pesquisa foi a de criar um robô capaz de se locomover em um ambiente plano com a maior eficiência possível. Os componentes básicos permitem a construção tanto de robôs movidos as rodas quanto movidos a pernas, sendo que o resultado da pesquisa apresenta como uma de suas conclusões a ideia que em robôs cujo objetivo é movimentação horizontal a melhor solução são rodas. Apesar de os componentes básicos da morfologia serem aparentemente materializáveis, os autores não apresentaram discussões a respeito do assunto.

Esta pesquisa tem por objetivo definir um framework para evolução de robôs móveis baseados em sensores genérico o suficiente para ser adaptado na solução dos mais diversos problemas envolvendo robótica móvel. Como principais diferenciais as pesquisas anteriormente descritas cita-se: construção de robôs móveis que se movimentem através de rodas, utilizem sensores como input, programação genética como mecanismo de controle e que possam ser materializados utilizando-se sensores de distância, motores e um microcontrolador de baixo processamento.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Algoritmo Genético

Algoritmo Genético (AG) é um algoritmo de computação evolutiva inicialmente proposto por [Holland, 1975]. AGs são capazes de identificar e explorar fatores ambientais e convergir para soluções ótimas, ou aproximadamente ótimas em níveis globais. Por ser baseado em fundamentos teóricos da biologia, o conceito de seleção natural está intimamente relacionado com os AGs.

Para [Darwin, 2009], seleção natural é um processo que atua continuamente em todos os indivíduos de um ambiente, selecionando as variações mais aptas para aquele ambiente, no que se refere ao melhoramento genético do indivíduo.. Na natureza, o processo evolutivo ocorre quando quatro condições fundamentais são satisfeitas [Koza, 1992]. Tais condições são:

- O indivíduo tem a capacidade de se reproduzir;
- Há uma população de indivíduos que se auto reproduzem;
- Há variedade entre os indivíduos que se autoreproduzem;
- A capacidade de sobrevivência no ambiente está associada à variedade.

A variedade citada anteriormente se manifesta como variação nos cromossomos dos indivíduos da população, ou seja, no nível molecular. Essa variação é considerada um gene pertencente a um determinado DNA.

Basicamente um AG é constituído de componentes como, a representação do indivíduo, o método de seleção, uma função de avaliação (fitness) e operadores genéticos [Koza, 1992].

3.2 Arquitetura de Subsunção

A tarefa do sistema de controle é fazer com que o robô alcance um determinado estado. O sistema de controle obtém informações do ambiente através dos seus sensores e transmite comandos para os atuadores do robô. Uma das principais técnicas de controle de robôs é a Arquitetura de Subsunção.

A arquitetura de subsunção foi proposta por [Brooks, 1986]. Nesta arquitetura, os comportamentos são definidos através de leituras dos sensores e envio de comandos para os atuadores. Os comportamentos são conectados uns aos outros formando uma rede organizada em camadas dispostas hierarquicamente. Cada camada é responsável por uma atividade do robô. Nas camadas mais altas estão os comportamentos mais abstratos e que levam o robô a efetuar um objetivo. Nas camadas mais baixas ficam os comportamentos responsáveis por ações básicas do robô.

Os comportamentos em cada uma das camadas funcionam de forma concorrente e independente. Estes comportamentos são coordenados de forma competitiva, sendo que os comportamentos em camadas mais baixas têm prioridade em relação aos comportamentos em camadas superiores. A coordenação é feita por meio de dois mecanismos principais chamados inibição e supressão de comportamentos. Na arquitetura de subsunção a hierarquia entre os comportamentos é definida de forma bastante específica [Brooks, 1986].

3.3 Simulação

O módulo de robótica projetado e implementado funciona como uma extensão da plataforma de robótica *GrubiBots* (ferramenta desenvolvida e mantida pelo Laboratório de Pesquisa *GRUBi*). Essa plataforma tem por objetivo ser utilizada no desenvolvimento e teste de algoritmos voltados a programação de robôs móveis autônomos baseados em sensores.

A plataforma *GrubiBots* utiliza a biblioteca de física *ODE - Open Dynamics Engine*. A *ODE* é uma biblioteca de alto desempenho, especialmente indicada para simulação de objetos móveis em ambientes dinâmicos [Zaeschke, 2014]. A *ODE4J* é a implementação da *ODE* para Java. A biblioteca *ODE* é útil para criar veículos simulados e robôs, além de estar sendo preparada para ser liberada como *open source* e tornar-se uma ferramenta aberta para auxiliar pesquisas na área.

4 MODELO PARA EVOLUÇÃO DA MORFOLOGIA E CONTROLE

Esta seção apresenta o problema abordado, a modelagem computacional da morfologia e controle dos robôs, os operadores de crossover e mutação, e a representação no DNA de cada indivíduo usada no presente trabalho.

4.1 Problema

A fim de validar o módulo desenvolvido, três casos foram propostos. Um total de dois cenários foram desenvolvidos, conforme apresentado na Figura 1. Por fim, duas funções objetivas foram determinadas.

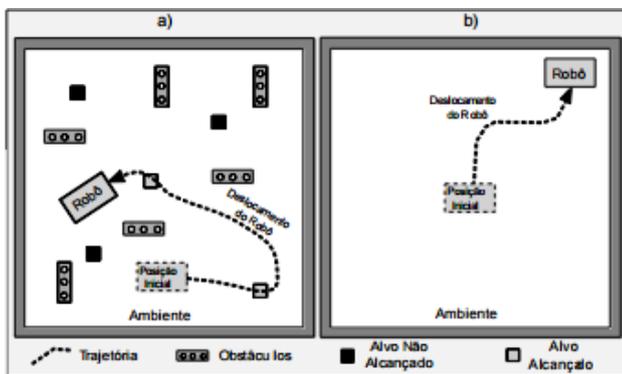


Figura 1 - Erro! Nenhuma sequência foi especificada. – Cenários utilizados na solução dos problemas.

O cenário ilustrado na Figura 1-(a) será chamado de *Cenário_1*. Este cenário visa encontrar um robô capaz de se locomover no ambiente com obstáculos passando por pontos de referência. Neste cenário foram definidos cinco pontos de referência. O cenário apresentado na Figura 1-(b) será chamado aqui de *Cenário_2*. Este cenário visa desenvolver um robô capaz de se locomover em ambiente livre de obstáculos. O robô encontra-se localizado no centro de uma região quadrada.

Duas funções de avaliação foram definidas, uma para cada cenário dado. A função de avaliação F_1 é a função de avaliação base para o *Cenário_1*. Esta função descreve que o robô que conseguir atingir o maior número de pontos de referência, colidir com a menor quantidade de obstáculos e gastar a menor quantidade de recursos de peças terá o melhor *fitness*, assim terá maior probabilidade de passar suas características para a próxima geração.

$$C_{morf} = W_{morf} \times \sum_{p \in P} (W_p \times Q_p)$$

$$C_{mark} = W_{mark} \times (Q_{markT} - Q_{markC})$$

$$C_{col} = W_{col} \times Q_{col}$$

$$\text{Min } F_1(\dots) = C_{morf} + C_{mark} + C_{col}$$

A função objetivo F_2 é a função de avaliação base para o *Cenário_2*. Esta função descreve que o robô que conseguir o maior deslocamento e a menor quantidade de recursos de peças gastos terá o melhor *fitness*.

$$C_{desl} = W_{desl} \times (Clado/2 - R_{desl})$$

$$\text{Min } F_2(\dots) = C_{morf} + C_{desl}$$

C_{morf} - Custo associado a morfologia do robô

C_{mark} - Custo associado aos marcadores

C_{col} - Custo associado às colisões do robô

C_{desl} - Custo associado ao deslocamento do robô

W_{morf} - Peso da morfologia

W_{desl} - Peso do deslocamento

W_{mark} - Peso dos marcadores

W_{col} - Peso das colisões

P - Conjunto de peças do robô

p - Peça pertencente ao conjunto de peças *P*

W_p - Peso atribuído à peça *p*

Q_p - Quantidade da peça *p*

Clado - Lado (comprimento) do cenário

R_{desl} - Deslocamento do robô

Q_{markT} - Quantidade de marcadores totais

Q_{markC} - Quantidade de marcadores capturados

Q_{col} - Quantidade de colisões

A definição dos três casos propostos são encontradas a seguir:
Caso 1 ↔ Testa a evolução do controle do robô. A estrutura morfológica do robô foi fixada. O cenário utilizado neste caso foi o *Cenário_1*. A função de *fitness* utilizada neste case é dada pela função F_1 .

Caso 2 ↔ Testa a evolução da morfologia do robô. As estruturas de controle são fixas. O cenário utilizado neste caso foi o *Cenário_1*. A função de avaliação utilizada é dada pela função F_1 .

Caso 3 ↔ Testa a evolução da estrutura morfológica em conjunto com as estruturas de controle do robô. O cenário utilizado neste caso foi o *Cenário_2*. A função objetivo utilizada neste case é dada pela função F_2 .

5 MODELAGEM DA MORFOLOGIA

Um robô é constituído por um conjunto de peças. Neste trabalho foram definidas as seguintes peças que constituirão o robô: Chassi, Processador, Roda Sem Motor, Roda Com Motor e Sensor de Distância. A Figura 2 mostra o conjunto de peças implementadas no módulo de robótica. Este conjunto de peças será representado neste trabalho por *P*.

Chassi (*P_{chs}*): peça que representa o corpo do robô. Esta peça serve de encaixe para todas as outras peças.

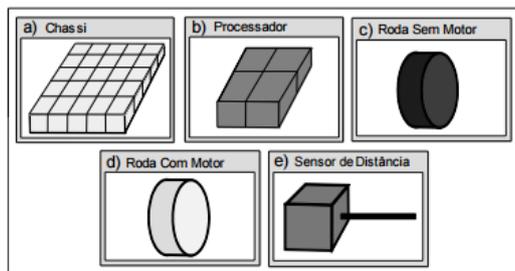


Figura 2 – Conjunto de peças P do robô.

Processador (*Pcpu*): peça que é responsável pelo controle do robô (comportamentos). Esta peça ocupará as células de um quadrado de dimensões 2 x 2. A Figura 2-(b) apresenta a morfologia do processador.

Roda Sem Motor (*Prsm*): peça responsável por auxiliar no deslocamento do robô, diminuindo o atrito com o chão. Este componente ocupará uma célula por completo e parcialmente as células vizinhas. A Figura 2-(c) mostra a morfologia desta peça.

Roda Com Motor (*Prcm*): peça responsável pelo deslocamento do robô no ambiente. Este componente ocupará uma célula por completo e parcialmente as células vizinhas. A Figura 2-(d) mostra a morfologia desta peça.

Sensor de Distância (*Psdd*): peça que extrai informações do ambiente e calcula distância entre o robô e obstáculos a sua frente. Este componente ocupará uma célula de dimensões de 1 x 1. A Figura 2-(e) mostra a morfologia desta peça.

O conjunto de peças apresentados foram acoplados sobre a estrutura do chassi. Para efetuar a acoplagem de tais peças, alguns pontos de conexão (junção) foram definidos. A Figura 3 ilustra onde estão localizados cada um dos pontos de junção.

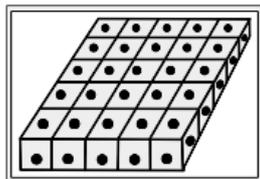


Figura 3 – Pontos de conexão das peças no chassi.

5.1 Montagem da Morfologia do Robô

Após a definição da população inicial de robôs, o AG (Algoritmo Genético) seleciona de um conjunto de peças, a seguir, o AG seleciona o lugar onde tais peças ficarão acopladas. A avaliação de cada um dos robôs é efetuada utilizando-se a função de fitness definindo desta maneira o desempenho do indivíduo gerado para um determinado cenário, em seguida, o método de seleção seleciona os pais para gerar o indivíduo filho. O método de recombinação será executado, e em seguida, mutações poderão ser executadas. Por fim, os indivíduos gerados são avaliados, este processo é repetido até que se esgote o tempo ou que o número de gerações definidas seja atingido.

Desta foram os melhores robôs de cada geração são selecionados para que sejam recombinados, gerando assim um novo indivíduo. Ao longo das gerações os piores robôs são descartados. Este processo tende a garantir que a cada geração os robôs melhorem a sua pontuação.

5.2 Operadores de Crossover

O operador de recombinação faz com que os descendentes recebam em seu código genético parte do código genético do Pai1 e parte do Pai2. A seguir, são descritos os operadores de *crossover* implementados.

Crossover de Processador - este *crossover* seleciona o processador de um dos pais para constituir o indivíduo filho.

Crossover de Roda Sem Motor - operador tem probabilidade de capturar metade destas peças do Pai1 e metade destas peças do Pai2.

Crossover de Roda Com Motor - este operador é análogo ao operador de *crossover* de Roda Sem Motor.

Crossover de Sensor - a implementação deste operador garante que sejam selecionados sensores de ambos os pais, porém desde que tais sensores não estejam em setores iguais.

5.3 Operadores de Mutação

O operador de mutação tem como objetivo garantir maior variabilidade genética na população, impedindo que a busca fique estagnada em um mínimo local. Adiante, são descritos os operadores de mutação implementados.

Adicionar Peça - este operador é responsável por adicionar uma peça ao robô. As peças que podem ser adicionadas dependem do case a ser resolvido. No *Case_1* e *Case_2* as peças permitidas são: *Prsm*, *Prcm*, e *Psdd*. Já no case *Case_3* as peças permitidas são: *Prsm* e *Prcm*.

Remover Peça - operador responsável por remover uma peça do robô.

Trocar Peças - este operador é responsável por efetuar trocas de peças do robô. A chance de ocorrer a troca de uma peça é de 50% das vezes que ocorrer a mutação. Primeiramente sorteia-se uma peça a ser trocada por outra peça. Caso esta peça seja um *Psdd* a troca ocorrerá na orientação do sensor. Já no caso da peça ser *Prsm*, ou ainda, *Prcm*, é verificado se a peça encontra-se nas laterais ou no topo do chassi. Caso esteja nas laterais então troca-se o tipo de roda. Caso esteja no topo do chassi então a peça poderá se transformar em *Prsm*, *Prcm* ou *Psdd*.

6 MODELAGEM DO CONTROLE

Para a modelagem do controle do robô foram definidos uma série de componentes que serão descritos a seguir:

Trigger - componente responsável por ativar um controle específico do robô. Este componente é ativado se a condição lógica existente nele for satisfeita.

Ação - Uma ação é tudo aquilo que um robô pode fazer como andar, parar, virar, etc.

Supressão - assim como na ação, a supressão tem a capacidade de ativar um componente ou desativar o mesmo. Esse comportamento é desejável para ocorrer à subsunção.

6.1 Operadores de Crossover

Para a ocorrência do operador de *crossover*, primeiramente são selecionados dois pais pelo método do torneio. Foi escolhido esse método por ser um método comumente utilizado. Após a seleção de dois pais, o método de

recombinação copia toda a lista de comportamentos de ambos os pais e a mistura de forma aleatória.

Com a lista de comportamentos finalizada, o operador genético seleciona de forma aleatória um ponto de corte para essa lista e seleciona também uma das partes desta lista, com 50% de chance para cada. Logo em seguida, a parte da lista selecionada é copiada para o DNA do filho.

6.2 Operadores de Mutação

Neste trabalho foram utilizados três tipos de mutação. Mutação de *trigger*, de ação e de supressão. Basicamente o princípio das mutações são os mesmos.

Para a mutação de *trigger*, o operador de mutação seleciona aleatoriamente uma *trigger* e modifica seu tipo entre o conjunto de tipos possíveis discutidos anteriormente. Esse conceito pode ser propagado para os outros tipos de mutação.

6.3 Representação do DNA

O controle do comportamento do robô é a parte responsável pela mobilidade do robô. Se o robô tiver rodas, mas não tiver comportamentos capazes de ligar estas rodas, o robô não poderá exercer a função de movimentar-se.

Na Figura 4 apresenta-se a representação de um indivíduo. Cada robô é composto por um DNA, e cada DNA é constituído de morfologia e comportamento. O DNA é o objeto que será manipulado pelo algoritmo genético, por esse motivo ele é constituído pela parte morfológica e comportamental do robô.

A morfologia do robô é composta pelas seguintes partes: peça, orientação e posição. O controle do comportamento, por sua vez, é constituído de *triggers*, ações e supressões.

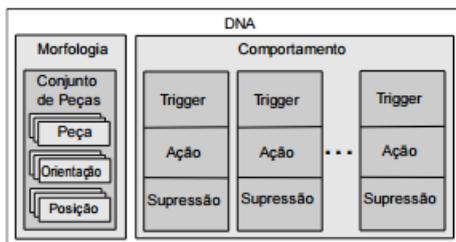


Figura 4 – Representação do DNA do Indivíduo.

7 RESULTADOS E DISCUSSÕES

A seguir são discutidos os resultados e discussões deste trabalho.

7.1 Desenvolvimento Manual da Solução Próxima do Ótimo

Os testes do Caso 1 fazem a evolução do controle, porém mantém a morfologia do robô fixa. Assim para efetuar as simulações foram definidos que o robô possuiria toda a estrutura morfológica definida da seguinte forma: o robô possuiria um chassi, um processador, dois sensores de distância colocados estrategicamente sobre o chassi, duas rodas com motor e duas rodas sem motor. Com o robô contendo esta estrutura básica foi feita a evolução do controle do robô sem, no entanto alterar os componentes morfológicos do robô.

Os testes do Caso 2 envolvem a evolução da morfologia, sendo que o controle foi mantido. O controle fixo foi definido da seguinte forma. Um conjunto com três comportamentos foram desenvolvidos: um comportamento responsável por andar (comportamento mais básico), um responsável por virar para direita, caso algum obstáculo apareça na esquerda do robô e outro comportamento responsável por virar para esquerda, caso algum obstáculo seja encontrado a direita do robô.

No Caso 3 foi avaliada a capacidade do modelo de evoluir tanto a morfologia quanto o controle do robô, neste caso o problema definido foi simplificado para diminuir a área de busca.

7.2 Resultados do Caso 1

Para este caso foi evoluído o controle, sendo que a morfologia foi mantida fixa contendo quatro rodas (duas com motor) e dois sensores de distância conforme figura 5.

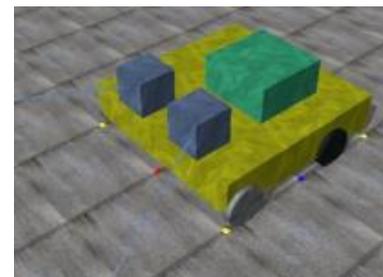


Figura 5 – Morfologia utilizada no Caso 1.

A melhor solução encontrada foi capaz de controlar o robô atingindo um total de 4 pontos de referência dos 5 definidos no cenário conforme pode ser verificado na figura 6.

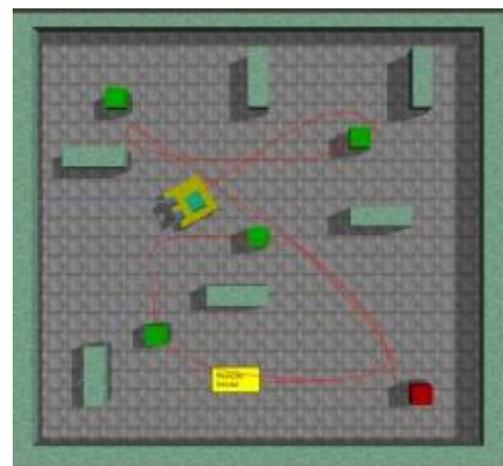


Figura 6 – Comportamento no Caso 1.

7.3 Resultados do Caso 2

Para este caso foi evoluída a morfologia mantendo-se o controle fixo. A melhor solução encontrada foi capaz de atingir 3 dos 5 pontos de referência determinados, além de ter pior desempenho que o Caso 1, o robô não foi capaz de desviar de todos os obstáculos. A Figura 7 apresenta a solução encontrada neste caso.

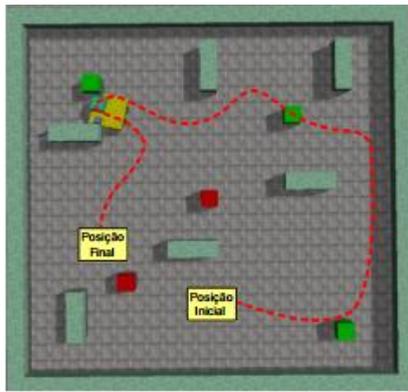


Figura 7 – Trajetória descrita pelo robô no Caso 2.

7.4 Resultados do Caso 3

A melhor solução encontrada neste caso foi um robô composto de um chassi, um processador, uma roda com motor e duas rodas sem motor, foram necessárias três rodas para manter o equilíbrio do robô, entretanto, a característica mais importante é que o robô foi capaz de deslocar-se linearmente conforme proposto no cenário. Uma representação da solução pode ser verificada na figura 8.

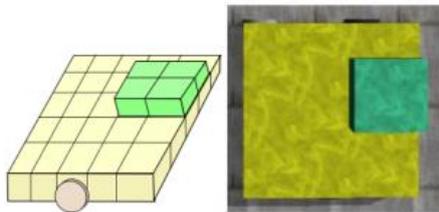


Figura 8 – Comportamento no Caso 3.

8 CONCLUSÕES

Este trabalho apresentou um módulo de robótica evolutiva capaz de evoluir a morfologia de um robô, com base em um conjunto de peças, um cenário e uma função objetivo desenvolvida. Este módulo foi implementado dentro da plataforma de programação de robôs móveis *GrubiBots*.

Foram propostos três casos para testes. Dois dos casos avaliaram a capacidade de criação de robôs móveis capazes de deslocar-se em um ambiente com obstáculos sem colidir passando por pontos de referência predeterminados. O outro caso avaliou a capacidade de criação de um robô capaz de deslocar-se linearmente em um ambiente sem obstáculos.

Todos os casos avaliados apresentaram resultados promissores abrindo espaço para novas pesquisas mais aprofundadas sobre o tema. Outro resultado significativo foi o desenvolvimento de uma ferramenta de simulação genérica o suficiente para permitir a criação de diversos tipos de cenários e problemas de robótica móvel que podem ser base para o desenvolvimento de novas morfologias e estruturas de controle.

AGRADECIMENTOS

Os autores deste artigo agradecem à Fundação de Amparo à Pesquisa do estado de Minas Gerais (FAPEMIG) pelo apoio financeiro parcial ao projeto de pesquisa e ao Grupo de Redes Ubíquas do Departamento de Ciência da Computação da Universidade Federal de Lavras (DCC-UFLA) pelo apoio técnico.

REFERÊNCIAS BIBLIOGRÁFICAS

- Brooks, Rodney A., A Robust Layered Control System for a Mobile Robot, IEEE Journal of Robotics and Automation, 1986.
- Chocron, O. Bidaud. P. Evolving walking robots for global task based design Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99., 1999.
- Darwin, C. A Origem das Espécies. [S.l.]: Editora Martin Claret Ltda, 2009.
- Holland, J. H. Adaptation in Natural and Artificial Systems. Ann Arbor, MI, USA: University of Michigan Press, 1975.
- Koza, J. R. Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA, USA: MIT Press, 1992. ISBN 0-262-11170-5.
- Lipson, H. POLLACK, J. B. Automatic design and manufacture of robotic lifeforms. Nature, International Weekly Journal of Science, 2000.
- Lund, H. H. Co-evolving Control and Morphology with LEGO Robots Morpho-functional Machines: The New Species 2003, pp 59-79, 2003.
- Sims, K. Evolving Virtual Creatures. Computer Graphics (Siggraph '94 Proceedings), 1994.
- Sims, K. Evolving 3D Morphology and Behavior by Competition. Artificial Life IV Proceedings, Artificial Life IV Proceedings, 1994. (b)
- Tilman Zäschke, ode4j A Java 3D Physics Engine & Library, <http://www.ode4j.org/>, Jan, 2014.

Observação: O material multimídia deste trabalho encontra-se disponível em: www.mnr.org.br/mostravirtual.