

Uma Colônia Artificial de Formigas Modelada por Sistema Evolutivo

Danilo Vasconcellos Vargas¹, Mauro Miazaki²,

Eduardo do Valle Simões², Alexandre C. B. Delbem²

Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo

Cx. Postal 668 - CEP 13560-970 -São Carlos - SP - Brasil

¹ask@grad.icmc.usp.br, ²{mauro, simoes, acbd}@icmc.usp.br

Abstract

This paper applies an Evolutionary System to develop a distributed decentralized controller to navigate a team of robots that should behave like an artificial Ant Colony. The targeted application is navigation with exploration of the environment, “foraging” (search for objects) and regrouping, where complex ant-based behaviors should emerge from the interaction of simple individuals. A set of behavioral functions have been written according to field observation of biological ant colonies. Therefore, an evolutionary algorithm is applied to select and configure these functions that define individual behaviors, optimizing robot actions and the way they interact to produce a coordinated navigation control system.

1. Introdução

A natureza tem recentemente inspirado muitos cientistas com seus seres vivos, que são de muitas formas bem superiores às máquinas, tomando como exemplo os insetos que são muito diversificados e que têm se mostrado eficientes soluções, sobrevivendo ao contínuo processo de seleção natural e se espalhando pelo meio ambiente [11]. Este artigo apresenta um projeto que tem como objetivo observar os insetos (formigas) para modelar seu comportamento e construir robôs com comportamento similar, almejando uma solução (heurística) para problemas tradicionais da robótica, como navegação, exploração e busca por objetos no ambiente [15].

A comunicação das formigas na navegação baseia-se em produzir ferormônios e soltá-los por onde passam, servindo de referência para as outras que por ali passaram [10]. A partir de simples comportamentos como produzir ferormônios e segui-los, as formigas só percebem a sua vizinhança, como autômatos celulares [7]. Porém, comportamentos mais complexos e até

mesmo inteligentes podem emergir da interação entre elas.

Então, ao se ter robôs parecidos com as formigas pode-se conseguir um comportamento de exploração com algoritmos de controle mais simples e mais econômicos em termos de hardware. Desse modo, com robôs mais simples, pode-se utilizar um maior número de robôs e se obter uma solução mais tolerante a falhas, já que a perda de um robô não fará tanta diferença no cumprimento da tarefa [6].

Este trabalho apresenta métodos de exploração otimizados, capazes de buscar recursos específicos no meio ambiente (“Foraging”) [16] e sem se afastar muito da “colônia” (sempre buscando voltar para a colônia que pode representar tanto o lugar onde ocorrerá a reposição de energia quanto o lugar onde os robôs encontram a maior parte dos outros indivíduos para “induzi-los” a segui-los). Assim, procura-se manter nos robôs um comportamento um pouco caótico, porém controlado, permitindo que um robô pouco valorizado (com uma função de avaliação de valor pequeno) seja produzido. Este robô, por ser um robô de certa forma “ruim”, pode se afastar do grupo e assim promover a perda de um indivíduo, mas é preciso levar em conta que se nesse meio tempo ele encontrar um objetivo (uma “comida”, por exemplo) e voltar para avisar a colônia, este robô terá contribuído muito com o sucesso da colônia [4].

Para atingir este objetivo, este trabalho combina técnicas de Inteligência Artificial, autômatos celulares “dinâmicos” [7] (cada formiga como sendo um autômato celular), sistemas evolutivos [19] (para otimização dos códigos de controle de cada robô-formiga) e sistemas multi-agente [15] para produzir um sistema de controle para times de robôs exploradores.

2. Comportamentos Emergentes

Os robôs-formiga apresentam vários comportamentos complexos (este trabalho procura gerar um algoritmo capaz de produzir funções

comportamentais semelhantes às aquelas discutidas em [9], mas de uma maneira mais “bottom up”) resultantes da união de um algoritmo evolutivo e um novo sistema de autômatos celulares com mudança de estados dinâmica em função do tempo, e não somente dos outros autômatos vizinhos. Esta solução híbrida proporciona uma exploração do meio e uma aglomeração dos “robôs-formiga”, formando uma colônia (“formigueiro”) que se movimenta em conjunto, explorando o ambiente e coletando objetos específicos. Dentro deste contexto, procurou-se:

- Conseguir controlar uma população de pelo menos 10 robôs;

- Aplicar o sistema em tarefas de exploração/coleta de objetos (“foraging”);

Esta pesquisa tomou como ponto de partida os trabalhos de [8], [10], [11], [12], [13], [14] e [17], onde foram realizados experimentos com formigas naturais, identificando como elas se comportam. Outro trabalho que serviu de fonte de inspiração para o desenvolvimento deste foi realizado por G. Baldassarre, S. Nolfi e D. Parisi em [2]. Neste artigo fala-se de vários robôs navegando em um ambiente de forma que fiquem próximos uns dos outros, mas que ao mesmo tempo explorem o ambiente. Neste experimento, todos os robôs são iguais aos outros (clones), proporcionando uma certa linearidade. Cada um deles possui sensores de navegação para evitar colisões e no começo é iniciada uma formação (por exemplo, a de uma cobra) e os robôs tentam evitar colisões e continuar naquela formação. O sistema proposto em [2] foi executado em simulação.

Da mesma forma, Lynne E. Parker em [3] faz uma experiência com robôs e sensores tentando limpar uma área, necessitando a cooperação de todos e quando ocorria uma falha em algum robô o outro ocupava seu lugar. Para limpar o lugar foi utilizada uma organização na qual todos os robôs sabem fazer todas as tarefas, mas cada um tem uma tarefa que faz melhor que os outros (pois os robôs são diferentes entre si), e assim normalmente após a divisão das tarefas um ou outro robô fica sem fazer nada, então este escolhe uma tarefa aleatória e começa a competir com o robô que já a está realizando, então, o melhor é designado para a sucessão naquela tarefa por uma função de avaliação. Então o robô que perder voltara a escolher aleatoriamente a tarefa que irá executar, voltando ao ciclo.

Este trabalho investiga a aplicação destas técnicas para o controle de sistemas robóticos reais. Das observações realizadas sobre o comportamento de formigas naturais durante um período de seis meses e das considerações feitas por K. Vittori e A. F. R.

Araújo [1], G. Baldassarre, S. Nolfi e D. Parisi [2] e Lynne E. Parker [3], além das análises dos experimentos com formigas naturais descritos em [8-14], surgiram várias idéias que originaram os comportamentos descritos abaixo.

3. Sistema Evolutivo

O sistema evolutivo (SE) é elitista e utiliza a simulação para avaliar a população [20]. Cada população contém 10 indivíduos, na qual apenas um sobrevive (o “pai”) e o restante dos indivíduos têm seus códigos genéticos substituídos, onde cada “gene” do cromossomo tem uma chance de 50% de vir do pai e 50% de chance de vir da mãe. Após essa operação, o código genético da mãe é substituído pelo código do filho, ou seja, após reproduzir-se, a mãe “morre” para ser reconfigurada com os parâmetros do filho gerado.

Além disso, tem-se uma taxa de mutação de 4%, ou seja, a cada 100 genes do cromossomo, 4 são invertidos pela mutação (se for 1 torna-se 0 e vice-versa) e também tem 3% de chance por geração de ocorrer a presença de um órfão, onde um indivíduo (o pior fitness) é substituído por outro que possui um cromossomo formado aleatoriamente, almejando a inclusão de mais diversidade genética na população [18].

O processo de avaliação é realizado da seguinte maneira: os robôs-formiga se movimentam em simulação pelo seu ambiente e de acordo com suas atitudes eles são avaliados pelas seguintes funções:

3.1. Função de Exploração

Ao iniciar a simulação, cada indivíduo é avaliado isoladamente. É gerado um círculo de raio R em volta da “formiga” e esta se move por conta própria dentro dele, onde perde 1 ponto a cada iteração, até ultrapassar o círculo (a avaliação pode ficar negativa). Neste momento então esta recebe pontos (+10 pontos) por tê-lo ultrapassado e então é gerado um novo círculo de raio R em volta da “formiga”, reiniciando o processo (ver figura 1).

3.2. Função Sociável

A cada iteração as “formigas” fazem um círculo em volta de si mesmas e contam o número de outras “formigas” que se encontram dentro do círculo, deste modo aumentam a função sociável em 4 pontos por “formigas” presente no círculo. A figura 2 exemplifica essa função.

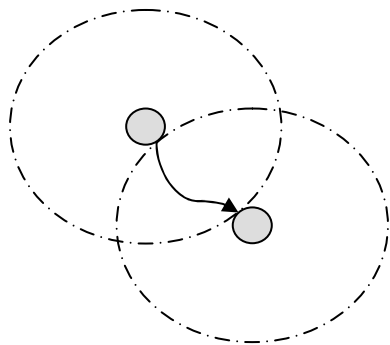


Figura 1 – Diagrama representando o círculo de pontuação da Função Exploração.

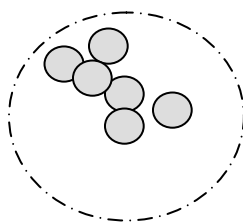


Figura 2 – Diagrama representando o círculo de pontuação da Função Sociável.

3.3. Função Alimentação

O robô recebe um bônus na função de avaliação por encontrar “comida” (objetos na tela) igual a 500 pontos (prêmio). Porém, o robô que encontrar a comida começará a perder pontos (-10 por iteração) na sua função avaliação até o momento em que este encontrar pelo menos outros três robôs e dividir com estes sua “comida”, valorizando a volta para a colônia ou a exploração conjunta, onde ganha 5000 pontos. Esta função se soma à função exploração formando uma única, denominada exploração, para fins didáticos ela foi separada.

3.4. Função de Avaliação

A Função de Avaliação é a soma das duas outras funções: Sociável e Exploração. Valorizando os robôs-formiga que apresentarem desenvolvimento com todas as funcionalidades nestas três áreas, ou seja, valorizando os robôs multi-especialistas. Como o elitismo neste caso não garante que o melhor encontrado não seja superado na próxima geração, isso dificulta determinar o melhor indivíduo da geração por elitismo.

4. Módulos Comportamentais

Os comportamentos dos robôs-formiga dependem da situação onde se encontram os indivíduos, de seus estados e seus respectivos cromossomos, os quais serão explicados em detalhes a seguir:

Estados: São regras que proporcionam uma orientação ao movimento dos robôs (características de autômatos celulares [7]) que todas as formigas possuem, diferentemente das funções que são selecionadas pelos cromossomos. E cada um desses estados apresenta um valor respectivo (uma variável) que funciona como um limiar, ou seja, quando um valor de um estado se torna maior que o outro a formiga muda de estado, desabilitando o último e seguindo as regras do novo estado que ela se encontra. Existem funções que modificam essa troca de estados, aumentando ou diminuindo os valores dos estados, e assim proporcionando uma mudança de estado diferente para cada formiga, dependendo como é seu cromossomo.

Os estados que cada indivíduo pode assumir são:

ESTADO1 - neste estado, a formiga tende a explorar o meio, afastando-se dos ferormônios colocados pelas outras.

ESTADO2 - neste estado a formiga procura ir de encontro a outras formigas, seguindo as trilhas de ferormônios das outras.

Funções: Cada robô “formiga” apresenta diferentes tipos de comportamentos, definidos pelo seu cromossomo (ver figura 3). Os cromossomos por sua vez possuem 15 “genes” que caracterizam a presença ou não de funções ou aumento da intensidade destas (o cromossomo é um vetor Booleano).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----

Figura 3 – Cromossomo do robô-formiga contendo os bits que configuram suas funções comportamentais.

O cromossomo tem a seguinte estrutura, normalmente, “0” significa que a função não é selecionada e “1” que a função está presente no robô. Em outros casos, o bit configura a função, modificando a intensidade ou a influência da mesma.

0- Impulsivo – Se impulsivo, analisa a situação e mantém a decisão por um tempo igual a 10 iterações.

1- Ferormônio – Sempre deposita ferormônios não importando o estado que se encontra o robô.

2- Decidido – Escolhe um lado após colidir com a parede e segue sempre para esse lado por um período de tempo igual a 20 iterações.

3- Solidão – Quando estiver longe de ferormônios o robô-formiga muda de estado para ESTADO2.

4 e 5- Variação na Função SOLIDÃO – diminui a intensidade da função solidão (00 – Intensidade normal; 01 ou 10 – Intensidade + 1; 11 – Intensidade + 2).

6- ESTADO2/ESTADO1 – Este bit diz se o robô-formiga coloca ferormônios quando está no estado ESTADO2 (bit = 0) ou ESTADO1 (bit = 1) (lembrando que se o cromossomo 1 estiver desabilitado, ele sempre depositará ferormônios).

7- Amigo – determina se o robô ficará ESTADO2 ou ESTADO1 quando estiver perto de ferormônios (ficará ESTADO2 ou ESTADO1, dependendo do bit 10).

8 e 9- Variação na função Amigo – Aumenta a intensidade da função Amigo (tem mais chances de entrar na função Amigo: 00 – Intensidade normal; 01 ou 10 – Intensidade + 1; 11- Intensidade + 2).

10- ESTADO1 ou ESTADO2 se Amigo – Diz se a função Amigo fará o robô-formiga ficar ESTADO2 ou ESTADO1.

11- Movimento Limitado – O robô-formiga diminui suas possibilidades de movimento, apenas escolhendo os movimentos: à frente, à direita e à esquerda.

12- Indeciso – O robô-formiga tem uma chance de mudar o movimento pré-escolhido para um movimento deslocado um pouco mais para a direita ou para a esquerda, apresentando então uma variação nos seus movimentos.

13- Aumenta a influência da função Solidão (diminui o limiar da função solidão).

14- Aumenta a influência da função Amigo (diminui o limiar da função Amigo).

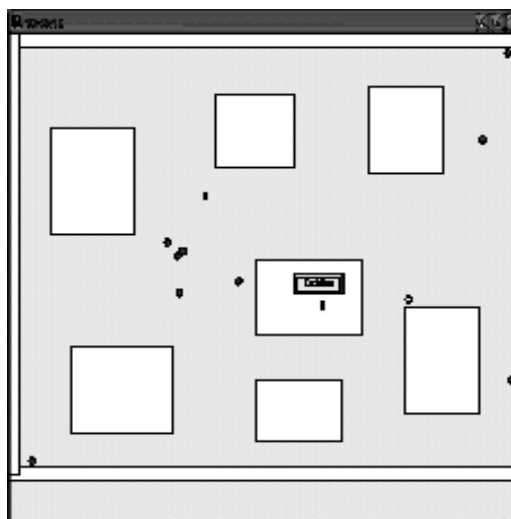
Como cada função interfere indiretamente ou diretamente nas outras, o sistema possui um número grande de possíveis soluções e para controlar e avaliar a relação de uma com as outras, utiliza-se um Sistema Evolutivo (selecionando as melhores relações e descartando as outras) para a convergência do sistema.

5. Resultados Obtidos

Obteve-se como resultante do conjunto de comportamentos evoluídos pelo algoritmo evolutivo uma certa similaridade com o comportamento exploratório das formigas. Assim foi possível a geração de comportamentos complexos a partir de regras simples controladas por algoritmos evolutivos.

5.1. Similaridade com as formigas

As figuras 4 e 5 trazem duas imagens da tela do simulador do experimento, no qual tenta-se simular um ambiente semelhante ao operacional dos robôs, para que a partir das funções de comportamento selecionadas pelo SE possam surgir robôs-formiga com comportamentos exploratórios similares aos das formigas naturais.



Figuras 4 – Simulação após 300 gerações: os pequenos círculos são os indivíduos evoluindo e os retângulos brancos são os obstáculos do ambiente.

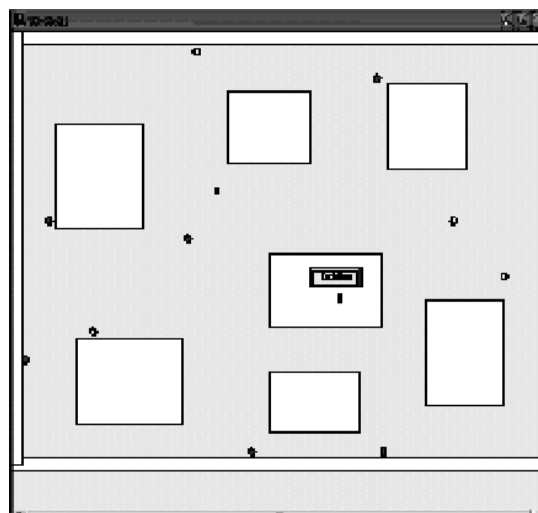


Figura 5 – Robôs-formiga explorando o ambiente de uma forma mais dispersa, porém mantendo-se próximos a pelo menos um outro robô.

Os círculos pequenos são os robôs-formiga, os tons mais escuros e mais claros dos círculos indicam os estados (os mais claros estão no estado ESTADO1 e os

mais escuros estão no estado ESTADO2) e os blocos brancos são paredes. Pode-se notar o agrupamento centralizado e a dispersão de alguns robôs pelo ambiente, demonstrando um grau de similaridade entre o agrupamento obtido (visto na figura) e o formigueiro natural, a dispersão obtida (vista na figura) e a busca por alimentos das formigas naturais.

A figura 5 mostra um outro aspecto comportamental dos robôs-formiga, uma dispersão regular pelo local onde eles estão estabelecidos, dependendo da quantidade de robôs-formiga eles podem cobrir uma área maior do ambiente. Isto foi verificado nas formigas naturais em observações de laboratório.

5.2. Objetivos Conflitantes

Buscam-se robôs-formiga que fiquem próximos uns dos outros e que ao mesmo tempo explorem o ambiente de forma a encontrar “comida” e trazerem estas para a “colônia” sem conhecimento global do ambiente. Esses dois objetivos são conflitantes e por isso mesmo um sistema evolutivo apresenta problemas de convergência entre dois possíveis ótimos locais, onde os indivíduos tornam-se muito bons em um dos dois objetivos e passam a desprezar o outro. Isso acontece porque na maioria dos casos uma melhor exploração resulta do espalhamento das formigas pelo ambiente e assim as formigas deixam de estar próximas e de marcar pontos na função Social. Da mesma forma, ao se manterem juntas (Sociais), elas perdem muitos pontos na função Exploração.

5.3. Experimentos com o Sistema Evolutivo

O gráfico da figura 6 demonstra uma melhoria significativa nas primeiras gerações, uma característica presente na maioria dos Sistemas Evolutivos, e apresenta súbitas melhorias que se mantém por um certo período de tempo e depois acabam por voltar ao nível anterior. Estes saltos nos gráficos acontecem porque o algoritmo evolutivo seleciona o melhor indivíduo (elitismo) para cruzar com todos os outros e como o melhor pode num determinado momento não apresentar uma solução satisfatória por influência da “sorte”, um outro robô um pouco pior por mero acaso pode encontrar mais comida que o anterior. Assim, esse robô aparentemente melhor (“sortudo”) pode ser considerado superior ao melhor pela função de avaliação e ser selecionado para reproduzir, provocando um decréscimo do fitness para as gerações futuras. No início do experimento, apresentado na figura 7, pode-se ver o crescimento rápido do fitness

inicial da população. Esta figura apresenta os mesmos dados da figura 5, mas em outra escala, para possibilitar a visualização das características evolutivas do sistema nas primeiras gerações.

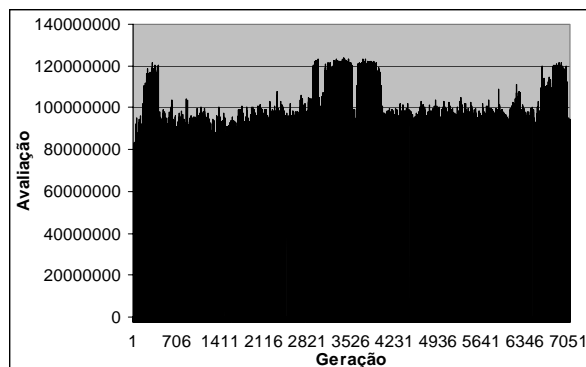


Figura 6- Gráfico do sistema evolutivo

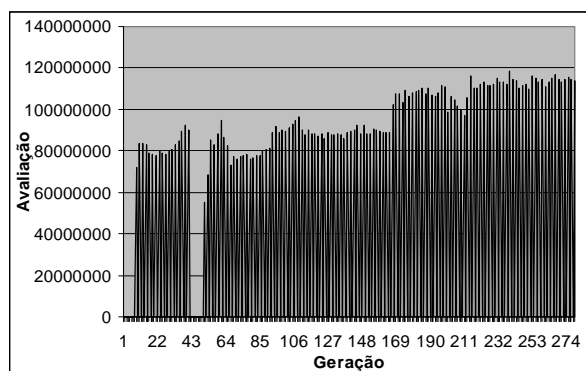


Figura 7- Gerações iniciais do experimento

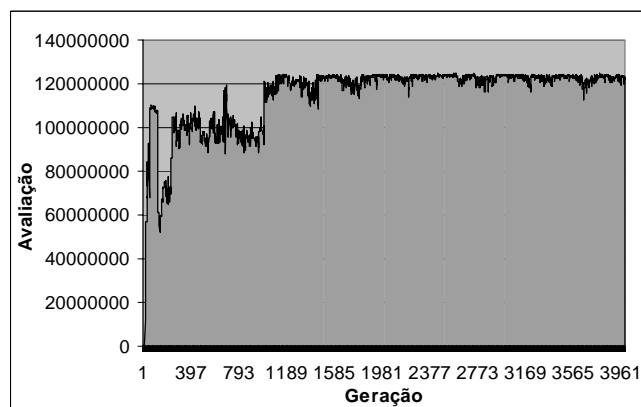


Figura 8- Seleção com hereditariedade.

O gráfico da figura 8 mostra uma estabilidade maior que a figura 6 e apresenta como única diferença a modificação da função de seleção, que agora inclui

uma função de hereditariedade [18], onde os indivíduos são selecionados comparando-se a média do fitness das últimas 6 gerações. Assim, mesmo que por “azar” o melhor indivíduo obtenha menor pontuação na geração atual, sua média dificilmente será menor que a de um “sortudo”. Demonstra-se desta maneira a necessidade de operadores genéticos mais robustos, capazes de lidar com os ruídos do sistema.

6. Conclusões

Este trabalho apresentou um sistema de controle de navegação de robôs móveis autônomos distribuído e descentralizado modelado a partir de observações realizadas em campo do comportamento de formigas naturais. Este sistema é uma colônia artificial de formigas onde vários comportamentos biológicos foram reproduzidos e um algoritmo evolutivo é utilizado para selecionar as funções comportamentais mais adequadas e configurar seus parâmetros. Assim, obteve-se um sistema de controle onde os robôs-formiga permanecem próximos uns dos outros e ao mesmo tempo exploram o ambiente para encontrar objetos específicos, trazendo-os para o grupo (“colônia”) sem comunicação global ou conhecimento global do ambiente.

7. Referências

- [1] Vittori, K., and Araújo, A. F. R., Agent-Oriented Routing in Telecommunications Networks. IEICE TRANS. COMMUN., Vol.E84-B, no.11, 2001.
- [2] Baldassarre, G., Nolfi, S., and Parisi, D., Evolution of Collective Behavior in a Team of Physically Linked Robots. In Applications of Evolutionary Computing, R. Gunther, A. Guillot, and J.-A. Meyer, editors, Springer Verlag, Heidelberg, Germany, pp. 581-592, 2003.
- [3] Parker, L. E., An experiment in mobile robotic cooperation. Proceedings of Robotics for Challenging Environments, Albuquerque, New Mexico, 1993.
- [4] Dorigo, M. and Caro, G. D., The Ant Colony Optimization Meta-Heuristic. D. Corne, M. Dorigo, and F. Glover, editors, New Ideas in Optimization. McGraw-Hill, London, pp. 11-32, 1999.
- [5] Dorigo, M. and Gambardella, L. M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. In: IEEE Transactions on Evolutionary Computation, vol. 1, no. 1, pp. 53-66, 1997.
- [6] Cao, Y. U., Fukunaga, A. S., and Kahng, A. B. Cooperative Mobile Robotics: Antecedents and Directions. Autonomous Robots, N. 4, Arkin, R.C. and Kluwer, G. A. B., editors, Academic Publishers, Boston, pp. 1-23, 1997.
- [7] Oliveira, G. M. B. D., Autômatos Celulares: Aspectos Dinâmicos e Computacionais. In: Anais da 3 Jornada de Mini-cursos de Inteligência Artificial, pp. 297-339, 2003.
- [8] Aron, S., Beckers, R., Deneubourg, J. L., and Pasteels, J. M., Memory and Chemical Communication in the Orientation of Two Mass-Recruiting Ant Species. In: Insectes Sociaux 40, pp. 369-380, 1993.
- [9] Sumpter, D. J. T., and Pratt, S. C., A Modeling Framework for Understanding Social Insect Foraging. Behav. Ecol. Sociobiol., 53, pp. 131-144, 2003.
- [10] Nonacs, P., and Soriano, J. L., Patch Sampling Behaviour and Future Foraging Expectations in Argentine Ants. The association for Study of Animal Behaviour, pp. 519-527, 1997.
- [11] Bonabeau, E., Theraulaz, G., Deneubourg, J. L., Aron, S., and Camazine, S., Self-organization in Social Insects. In: Trends in Ecology & Evolution (TREE), Elsevier Science, editor, Vol. 12, no. 5, 1997.
- [12] Fernandes, N., and Rust, M. K., Site Fidelity in Foraging Argentine ants (Hymenoptera:Formicidae). Sociobiology Vol. 41, no. 3, 2003.
- [13] Julianne D. Halley and Mark A. Elgar, The Response of Foraging Argentine Ants to Disturbance. Australian Journal of Oology, 49, pp. 59-69, 2001.
- [14] Rouston, T. H., and Silverman, J., The Effect of Food Size and Dispersion Pattern on Retrieval Rate by the Argentine Ant. Journal of Insect Behaviour, Vol. 15, no. 5, 2002.
- [15] Burgard, W., Fox, D., Moors, M., Simmons, R., and Thrun, S., Collaborative Multi-robot Exploration. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), San Francisco, CA, 2000.
- [16] Schweitzer, F.; Lao, K.; Family, F., Active Random Walkers Simulate Trunk Trail Formation by Ants. BioSystems, 41, pp. 153-166, 1997.
- [17] Goss, S., Beckers, R., Deneubourg, J. L., Aron, S., and Pasteels, J. M., How Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies. Unit of Behavioural Ecology, CP 231, Université Libre de Bruxelles, Bruxelles, Belgium, pp. 661-677, 1990.
- [18] Simões, E. D. V. and Dimond, K. R. Embedding a Distributed Evolutionary System into a Population of Autonomous Mobile Robots. In: IEEE International Conference on Systems, Man and Cybernetics, Tucson-AZ, USA, ISBN 0-7803-7089-9, pp. 1069-1074, 2001.
- [19] Floreano, D. and Mondada, F. Evolution of Homing Navigation in a Real Mobile Robot. In: IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, v. 26, n. 3, pp. 396-407, 1996.
- [20] Bramlette, M. F., Initialization, mutation and selection methods in genetic algorithms for function optimization. In: Proceedings of the Fourth International Conference on Genetic Algorithms, Belew, R. K., and Booker, L. B., editors, Morgan Kaufmann, San Mateo, CA, pp. 100-107, 1991.