

Sistema de Navegação Autônoma inspirado em Robótica de Enxame

Márcio Mendonça^{1,3}/Ivan R. Chrun³

3-Laboratório de Automação e Sistemas de Controle
Avançado (LASCA)
Universidade Tecnológica Federal do Paraná
Curitiba, Brasil
mendonca@utfpr.edu.br; ivanchrun@gmail.com;

Marcelo C. G. Regatieri²/Marco A. F. Finocchio¹

1-Laboratório de Segurança, Iluminação e Eficiência
Energética (LABISEE)
2-Programa de Pós-Graduação de Engenharia Elétrica
Universidade Tecnológica Federal do Paraná
Cornélio Procópio, Brasil
marcelocregatieri@gmail.com; mafinocchio@utfpr.edu.br

Resumo—Neste trabalho é apresentada uma estratégia de navegação em bando baseada em técnicas de Inteligência Coletiva para o controle de robôs autônomos utilizando Redes Cognitivas Dinâmicas Simplificadas (*Simplified Dynamic Cognitive Networks*, sDCN), uma evolução dos Mapas Cognitivos Fuzzy (*Fuzzy Cognitive Maps*, FCM). A construção e fundamentos dos controladores sDCNs serão apresentados no desenvolvimento desse trabalho. E, finalmente, resultados simulados utilizando um ambiente gráfico com duas dimensões em três diferentes cenários para observar resultados dos múltiplos objetivos: navegação em formação (seguindo o líder do grupo) e desvio de obstáculos.

Palavras-chave—Robótica De Enxame; Mapas Cognitivos Fuzzy; Redes Cognitivas Dinâmicas; Navegação Robótica Autônoma.

I. INTRODUÇÃO

Na natureza, podem-se encontrar diversos tipos de interação em grupo dentre os seres vivos, desde uma simples colônia de bactérias até mamíferos maiores e mais complexos vivendo em conjunto. Alguns exemplos destes comportamentos podem ser vistos em cardumes, rebanhos, bandos de pássaros, colônias de formigas, enxame de abelhas, entre outros. Cada indivíduo, se analisado individualmente, é simples e praticamente desprovido de inteligência, mas quando estes são governados por regras, cria-se uma inteligência coletiva que demonstra ser capaz de se adaptarem a diversos tipos de ambientes [1]. Desse modo, trabalhos com grupos de robôs inspirados em comportamento de colônias de formigas, ou em voos de pássaros, nos quais o bando segue o líder, mesmo que em fase inicial é o objetivo dessa pesquisa.

Nesse contexto, Brooks [2] descreveu uma arquitetura de controle de robôs móveis autônomos, Subsunção, na qual várias camadas de um sistema de controle são construídas para melhorar o nível de competência de suas operações inspiradas no comportamento de navegação de insetos. Esta arquitetura demonstrou ser robusta e flexível por possuir um comportamento de realização de tarefas dividido por camadas de nível alto e camadas de nível baixo, onde as camadas de nível mais alto teriam um peso maior sobre as camadas de nível mais baixo, outro diferencial da arquitetura de Subsunção comparado

a arquiteturas de controle clássicas está na capacidade de processamento paralelo.

Pesquisas utilizando sistemas computacionais inteligentes têm sido utilizadas na construção de sistemas de navegação autônomos, demonstrando uma capacidade de execução de tarefas complexas, em especial em aplicações com pouco ou nenhum conhecimento do ambiente [3, 4].

O termo “autonomia” se refere não só à capacidade de ação e decisão de um sistema de controle artificial, mas também à sua capacidade de adaptação do mecanismo que tomadas de decisões [5]. De outro modo; Sistemas Autônomos têm a capacidade de realizar tarefas complexas com um elevado grau de sucesso [3]. Neste contexto, a complexidade envolvida na tarefa de geração de trajetórias é reconhecidamente de alta complexidade e, em muitos casos, exige que o sistema autônomo seja capaz de aprender uma estratégia de navegação através da interação com o ambiente.

Há um interesse crescente no desenvolvimento de robôs (agentes) e veículos autônomos, principalmente devido à diversidade de tarefas, especialmente aqueles que expõem em perigo a saúde humana e / ou para o ambiente, [6] e [7]. Como exemplos, podemos citar trabalhos, que descrevem um robô móvel autônomo para uso em soldagem [8], meio ambiente exploração [9], subaquático [10], [11] e outros.

Diferente da utilização de apenas um robô autônomo, um grupo de robôs homogêneos podem ser empregados para solucionar problemas com melhor desempenho, dependendo da aplicação; em especial quando se necessita a exploração de uma maior área, ou até mesmo quando o tempo de execução da tarefa é uma variável objetivada [1]. Este trabalho trata problemas com decisões sequenciais em ambientes totalmente desconhecidos com múltiplos objetivos para robôs. A arquitetura proposta apresenta capacidade de sintonia e adaptação, gerenciamento de ações, e finalmente capacidade de interação entre os robôs por utilizar um algoritmo inspirado em inteligência coletiva.

Robótica de enxame (*Swarm Robotics*) é uma abordagem dos princípios de Inteligência Coletiva aplicados a um sistema

de coordenação para múltiplos robôs autônomos, capazes de se comunicarem e possuírem uma forma de operação inspirada biologicamente [1, 12].

A Robótica de Enxame apresenta características de otimização do sistema, como: melhor desempenho, tolerância a falhas e decréscimo da complexidade individual. Um problema pode ser resolvido apenas adicionando mais robôs que compartilhem a mesma regra ao processo, aumentando assim sua chance de sucesso [13]. Outra justificativa é a robustez proporcionada ao sistema, pois caso algum robô se perca no processo, este problema pode não ser significativo para a obtenção do objetivo esperado [1].

A modelagem do sistema de navegação utilizada nos robôs é baseada nas Redes Cognitivas Dinâmicas Simplificadas (*Simplified Dynamic Cognitive Networks*, sDCN). A DCN, por ser uma técnica evoluída dos Mapas Cognitivos, consiste em um algoritmo com baixa complexidade computacional e portátil, com possibilidade de ser embarcado em diversos tipos de microcontroladores [14]. Seu modelo cognitivo final tem semelhanças com os modelos do Mapa Cognitivo *Fuzzy* (FCM), com a inclusão de outros tipos de relações e conceitos com a possibilidade de implementação de uma lógica de reajuste em tempo real dos pesos entre as relações causais dos conceitos. Neste trabalho foi utilizado um Algoritmo de Aprendizado por Reforço com uma heurística de sintonia. A sDCN será abordada no desenvolvimento desse artigo.

O objetivo deste trabalho consiste em desenvolver uma lógica de navegação para controlar robôs em grupo utilizando as Redes Cognitivas Dinâmicas Simplificadas. Somente o algoritmo de exploração em grupo será apresentado neste trabalho. Esta estratégia de navegação tem como base seguir um líder estabelecido, inspirado em robótica de enxame.

Este trabalho está organizado em diversas seções. Na Seção II serão descritos alguns conceitos relacionados ao controle da robótica de enxame e a fundamentação teórica a respeito dos Mapas Cognitivos *Fuzzy* (*Fuzzy Cognitive Maps*, FCM), e de sua distinção para as Redes Cognitivas Dinâmicas e/ou as Redes Cognitivas Dinâmicas Simplificadas (utilizada neste trabalho); na Seção III será demonstrado o desenvolvimento do trabalho para a obtenção dos resultados; na Seção IV serão apresentados os resultados obtidos em três cenários diferentes; as considerações finais serão apresentadas na Seção V.

II. FUNDAMENTOS TEÓRICOS

A. Movimentação em Grupo entre Robôs

O controle de um enxame de robôs foi claramente dividido entre controle centralizado e controle distribuído [15]. O controle distribuído é quando a organização dos agentes autônomos não depende de um líder, enquanto que o controle centralizado é um sistema que possui um agente líder responsável pela coordenação do grupo, os líderes em alguns casos podem ser até virtuais.

Lynne [16] em seu trabalho realizou experimentos relacionando as vantagens e desvantagens entre as abordagens de controle para um enxame de robôs e demonstrou que para obter um bom controle do grupo, seria recomendado a utilização

de uma estratégia desenvolvida através da combinação entre o controle centralizado e o distribuído.

B. Mapas Cognitivos *Fuzzy*

Os Mapas Cognitivos [17] representam conexões entre porções de informação sobre um tema ou tarefa, e seus elementos são arranjados de forma intuitiva de acordo com o grau de importância dos conceitos. Eles são organizados em ramificações, grupos ou áreas.

Mapas Cognitivos *Fuzzy* é uma metodologia de modelagem [18], que estendeu os mapas cognitivos para a inclusão da lógica *Fuzzy*, apresentando assim um sistema com características semânticas dos sistemas *Fuzzy* e propriedades de estabilidade das Redes Neurais Artificiais [19]. Mais recentemente os FCMs foram conceituados como uma combinação de Lógica *Fuzzy*, Redes Semânticas, Redes Neurais Artificiais, Sistemas Especialistas [20]. A ilustração gráfica de um FCM é representada por um grafo cíclico ou acíclico, com suas relações causais determinadas por números *Fuzzy* para quantificar os pesos entre os arcos de conexões. Os nós são usados para descrever a característica de comportamento principal do sistema, os conceitos, e são conectados uns aos outros com um valor fixo de peso representando o nível de relação ou causa e efeito entre os conceitos (na prática podem ser variáveis de estados de um problema), no caso específico desse trabalho, os conceitos serão sensores e atuadores de um robô ou agente móvel. O grafo da Fig. 1 demonstra um exemplo de FCM, no qual é possível identificar as conexões causais e os conceitos do mapa, Conceitos (A1 até A5) com suas respectivas relações de causa e efeito, representadas pelas flechas no grafo.

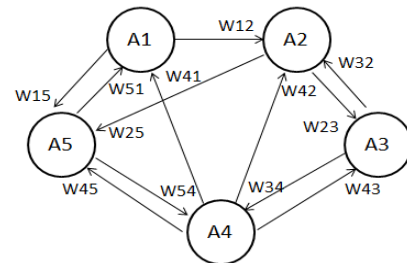


Fig. 1. Exemplo de um FCM.

$$W(j, i) = \begin{bmatrix} 0 & w_{12} & 0 & 0 & w_{15} \\ 0 & 0 & w_{23} & 0 & w_{25} \\ 0 & w_{32} & 0 & w_{34} & 0 \\ w_{41} & 0 & w_{43} & 0 & w_{45} \\ w_{51} & 0 & 0 & w_{54} & 0 \end{bmatrix} \quad (1)$$

Por conseguinte, a inferência FCM é obtida após várias iterações, os valores dos conceitos evoluem como mostra em (2) até estabilizarem em um ponto fixo ou em um ciclo limite [21].

$$A_i(t) = f\left(\sum_{j=1, j \neq i}^n A_j(t-1) * W_{ji}\right) \quad (2)$$

onde:

- n é o número de nós no grafo;

- j é o contador das iterações;
- W_{ji} é o peso do arco que conecta o conceito A_j ao conceito A_i ;
- A_i é o valor do conceito na iteração atual;
- A função f é uma função do tipo sigmoide (3).

$$f(x) = \frac{1}{1+e^{-\lambda x}} \quad (3)$$

O FCM pode não estabilizar e oscilar em alguns casos, ou até apresentar um comportamento caótico [22]. Em sistemas normalizados pode-se observar que os valores dos conceitos atingem um ponto de equilíbrio fixo ou um ciclo limite, após um número finito de iterações.

C. Redes Cognitivas Dinâmicas

As Redes Cognitivas Dinâmicas surgem como uma evolução dos Mapas Cognitivos que oferecem possibilidades mais amplas na gestão de estruturas causais e na modelagem de sistemas que apresentam não linearidades fortes e fenômenos temporais acentuados [23], [24]. Os valores dos pesos associados aos arcos variam no tempo de acordo com alguma função, diferente dos Mapas Cognitivos E Mapas Cognitivos *Fuzzy* que possuem seus valores dos pesos fixo, permitindo assim a construção de modelos cognitivos dinâmicos que se adaptam naturalmente.

As Redes Cognitivas Dinâmicas, da qual foi abstraído o modelo simplificado utilizado nesse trabalho, agregam novos tipos de relações ao modelo cognitivo clássico dos FCM, em resumo permitem tratamento da ocorrência de eventos, o tempo de forma indireta e não linearidades de um modo geral. As quais são duas grandes desvantagens dos modelos clássicos, os quais não tratam o tempo devido à ocorrência simultânea de todas as relações causais; e as funções modeladas pelos FCMs clássicos são somente monotônicas, maiores detalhes construtivos da DCN podem ser conferidos na Tese de doutorado de Mendonça [14]. O modelo cognitivo empregado nessa pesquisa é uma simplificação da versão original, porque só aplica as relações de seleção que chaveiam os estados de operação dos controladores e algoritmo dinâmico de sintonia, explicados na próxima seção.

III. DESENVOLVIMENTO DA NAVEGAÇÃO AUTÔNOMA

Os agentes autônomos nesse trabalho utilizam um controle centralizado, e seu modelo de hierarquia possui uma operação semelhante à arquitetura de subsunção [25], em que as ações de alto nível desativam as de baixo nível (no sentido de baixo para cima, como no modelo de Brooks).

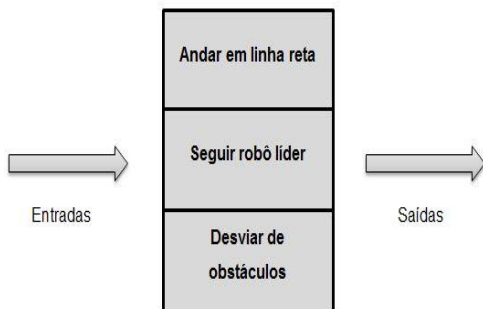


Fig. 2. Camadas das tomadas de decisões.

Ou seja, nesse trabalho as prioridades são: 1° – desviar de obstáculos; 2° – seguir o líder; 3° – andar em linha reta.

Deve-se ressaltar que as camadas de tomadas de decisões na Fig. 2 são aplicadas aos robôs seguidores, porque o líder só explora o ambiente com somente duas decisões, desviar de obstáculos e seguir em linha reta. O mesmo se aplica ao modelo cognitivo que o líder não tem as entradas **DX.E.** e **DX.D.** que são destinadas a verificar distancia que os seguidores estão do líder do grupo, em outras palavras, pode-se utilizar as entradas **DX.E.** e **D.X.D.** com valores zeros, fisicamente a distância do líder para ele próprio é zero.

TABELA I
DESCRÇÃO DAS RELAÇÕES CAUSAIS DA DCN

Relação Causal	Conceitos	Efeito	Intensidade
W_1	S.F. \rightarrow V.E.	Positivo	0,5
W_2	S.F. \rightarrow S.L.R.	Positivo	0,5
W_3	S.F. \rightarrow V.E.	Positivo	0,5
W_4	S.D. \rightarrow V.E.	Positivo	0,8
W_5	S.E. \rightarrow V.D.	Positivo	0,8
W_6	S.D. \rightarrow V.D.	Negativo	0,05
W_7	S.E. \rightarrow V.E.	Negativo	0,05
W_8	DX.E. \rightarrow V.E.	Positivo	0,8
W_9	DX.R. \rightarrow V.D.	Positivo	0,8

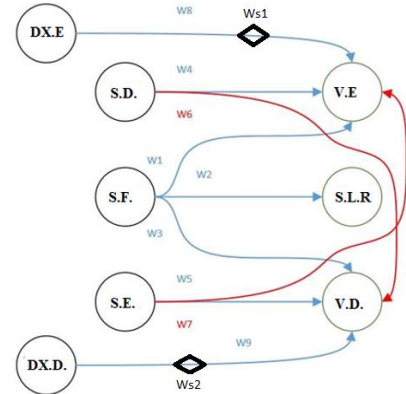


Fig. 3. DCN para Navegação em Grupo.

Os procedimentos iniciais para o desenvolvimento de uma DCN são os mesmos para a criação de um FCM, no qual se identificam os conceitos e as relações causais do problema. Os conceitos podem ser tantas ações como variáveis de controle. As relações causais precisam ser analisadas para definir se possuem uma causalidade positiva ou negativa entre si e quais seriam seus ajustes iniciais, podendo estes ser definidos através de um chute inicial e, posteriormente, reajustados de uma forma aleatória para se obter o resultado esperado do sistema. Depois de identificados os conceitos e as relações causais, o próximo passo é o de desenvolver um modelo de sintonia para respostas dinâmicas no controlador. Desta forma, o controlador será capaz de atribuir regras orientadas a eventos que possam ocorrer durante a navegação. A estrutura da DCN (Fig. 3) para navegação autônoma é constituída de cinco entradas e três saídas. Os conceitos de entrada são S.E. (sensor esquerdo), S.D. (sensor direito), S.F. (sensor frontal), DX.E. (detecção do líder à esquerda) e DX.D. (detecção do líder à direita). Os conceitos de saída são V.E. (virar à esquerda), V.D. (virar à direita) e S.L.R.

(seguir em frente) para os atuadores. A Tabela I apresenta a intensidade inicial utilizada entre as relações causais. Como o líder tem somente dois estados e só necessita de desvio de obstáculos, quando os mesmos surgirem durante a trajetória, o mesmo pode ser modelado por um FCM clássico, sem as relações de seleção W_{s1} e W_{s2} , e, conseqüentemente ter comportamento reativo (resposta direta aos sensores de entrada, de forma similar aos FCMs).

O algoritmo responsável pelo ajuste *on-line* dos pesos das relações causais é inspirado no Algoritmo de Aprendizagem por Reforço Baseado em Heurística do trabalho [4]. Entretanto o algoritmo é semelhante ao utilizado em [14]. Porém a sDCN desse trabalho considera o líder como um alvo móvel para os outros agentes (seguidores). O líder a princípio não utiliza os sensores dos alvos DX.E e DX.D, entretanto poderia ser uma funcionalidade incrementa em futuros trabalhos com cenários mais complexos com coleta de alvos.

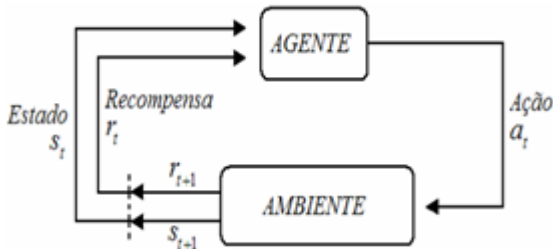


Fig. 4. Modelo Padrão de Aprendizagem Por Reforço

O algoritmo clássico (Q-learning) proposto por [26], como mostra a Fig.4. E sintoniza dinamicamente as ações dos agentes da seguinte forma: o ambiente pode ser parcialmente ou completamente observável e, através de sensores (em aplicações robóticas), o agente identifica os estados “S” atual do sistema. Em geral, o agente pode interagir com o ambiente, utilizando assim seus atuadores para efetuar ações “a” e mudar de estado. Para cada ação tomada, o agente pode receber uma recompensa “r”, que será utilizada para avaliar o seu comportamento. Assim, uma função de reforço $r(S(t), a(t))$ é um mapeamento de estados/ações para recompensas, num instante de tempo t . À medida que se dá a interação com o ambiente, o agente aprende a tomar decisões em determinadas situações que o levam a maximizar a soma de prêmios recebidos [14].

A cada iteração, o agente móvel atualiza seu valor de acordo com um gerenciamento de penalidades, determinando se o peso deve receber um valor de recompensa positivo ou negativo pelas suas ações.

$$W_{i(t+1)} = W_{i(t)} + \alpha * [r + (W_{lim} - W_{i(t)})] \quad (4)$$

O α é o fator que determina o grau da aprendizagem, o r é o valor da recompensa e W_{lim} é o mínimo/máximo valor que o peso pode alcançar. Para os pesos W_5 , a faixa varia entre [0,60 – 1,00] e para o peso W_2 , entre [0,35 – 0,65]. O valor adotado para o fator de aprendizagem foi de 0,01. A Tabela II relaciona a interação entre os sensores com o fator de recompensa de acordo com cada caso.

IV. VERIFICAÇÃO EXPERIMENTAL

Com o intuito de validar a DCN do Sistema Autônomo de Navegação apresentado, três simulações foram realizadas utilizando cenários com situações diferentes. Nos três cenários, o objetivo dos robôs é explorar ambiente junto ao robô líder. E, caso ele esteja próximo deve priorizar o desvio de obstáculos de sua trajetória. Dependendo da situação do ambiente, os agentes móveis podem ou não se distanciar do líder, desfazendo assim o grupo.

O ambiente de simulação adotado para a realização dos testes deste trabalho consiste em uma animação 2-D desenvolvida no MATLAB, representada por um plano X Y. As

TABELA II
FATOR DE RECOMPENSA ENTRE RELAÇÕES CAUSAIS

Interação	Relação Causal	Recompensa
+S.F.	W_2	+0,1
-S.F.	W_2	-0,1
+S.E.	W_5	+0,2
+S.E.	W_2	+0,1
-S.E.	W_5	-0,1
+S.D.	W_4	+0,2
+S.D.	W_2	+0,1
-S.D.	W_4	-0,1

+S.F. = Sensor frontal detectando obstáculo.
+S.E. = Sensor esquerdo detectando obstáculo.
+S.D. = Sensor direito detectando obstáculo.

dimensões deste plano no eixo X vão de 0 até 100, enquanto no eixo Y varia de -10 até 270. Os veículos estão simbolizados como “*”, e estão divididos por cores diferentes. A velocidade da animação é determinada pelo número de iterações e a distância de desvio de obstáculos dos sensores é de 11 unidades e para detecção do líder é de 30 unidades. O rastro das três cores representa a trajetória percorrida pelos três agentes diferentes, sendo o azul o veículo líder, e o vermelho e o verde os veículos que constituem o grupo, observa-se nas simulações que os veículos (agentes) buscam uma formação em triângulo após o desvio dos obstáculos distribuídos nos cenários. Já os pontos “●” pretos simbolizam os obstáculos.

A. Primeiro Cenário

No primeiro cenário, os três robôs partem próximos um do outro.

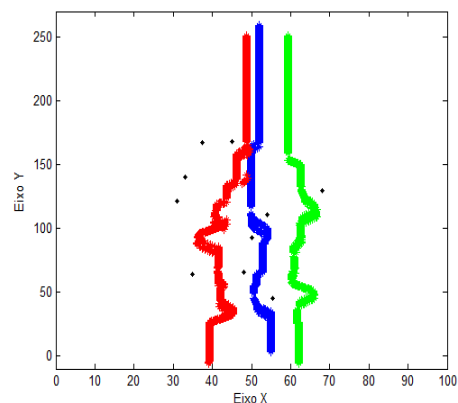


Fig. 5. sDCN no Primeiro Cenário.

As coordenadas X Y, o vermelho começa no ponto [41, -3], o azul no ponto [57, 5] e o verde no ponto [64, -3]. A Fig. 5 representa a DCN navegando no primeiro cenário.

Neste cenário os robôs conseguem desviar dos obstáculos e manter a formação em “V” no final do percurso, porém nota-se que o verde se posiciona de uma forma assimétrica com o vermelho. Isto ocorre devido tanto ao algoritmo de posicionamento possuir um grau de aleatoriedade no momento de navegação quanto aos sensores para desvio acusarem veículos próximos.

B. Segundo Cenário

Neste cenário, os veículos começam distantes uns dos outros. O vermelho no ponto [4, -5], o azul em [50, 5] e o verde em [100, -5] do plano.

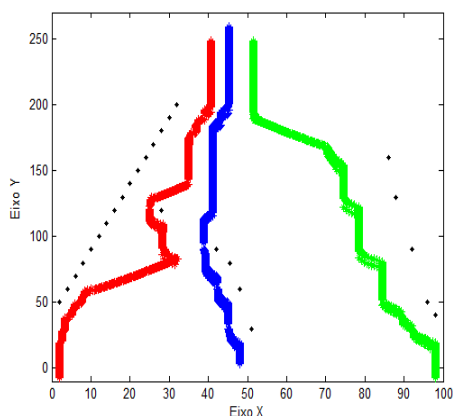


Fig. 6. sDCN no Segundo Cenário.

Este cenário foi designado para induzir que os robôs se encontrem durante a navegação, permitindo assim uma análise da resposta para este caso em que os robôs iniciam distantes do líder. A Fig. 6 demonstra o resultado obtido da navegação. Pode-se analisar o comportamento dinâmico reação no robô vermelho, em que ele fornece prioridade ao desvio do obstáculo, causando certo desvio na sua trajetória até o líder.

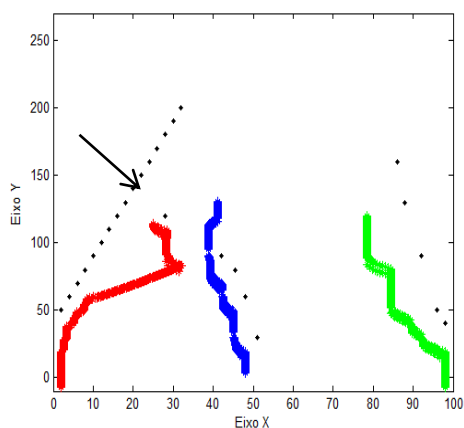


Fig. 7. sDCN no Segundo Cenário – Desvio de Trajetória.

A Fig. 7 aponta este momento, através de uma pausa na animação. Isto ocorre devido ao fato de que caso o sensor frontal do robô acusar um obstáculo e o sensor esquerdo não, o robô dará preferência de virar para a esquerda.

C. Terceiro Cenário

Os veículos começam próximos neste cenário, onde o vermelho inicia em [46, -3], o azul em [52, 5] e o verde em [58, -3]. O resultado obtido pela sDCN através da navegação no cenário está representado na Fig. 8.

O cenário da Fig. 8, foi desenvolvido para testar a capacidade de reestruturação do grupo após desviarem de obstáculos com maiores distâncias no cenário.

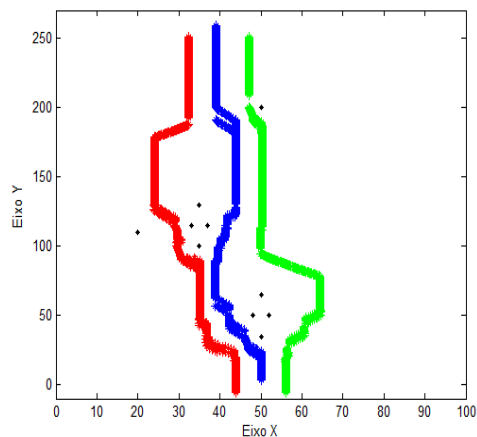


Fig. 8. sDCN no Terceiro Cenário.

De acordo com os resultados obtidos, ainda que iniciais e em cenários de baixa complexidade. Foi possível observar através do comportamento dinâmicos dos agentes (robôs). Controlados pelas sDCNs navegando nos três cenários demonstrados, foi possível notar que a movimentação em grupo funciona de maneira adequada na maior parte do tempo de simulação, porém, em alguns momentos, nota-se que a manobra de movimentação em grupo ocorre apenas quando há um desvio de obstáculo tanto por parte do líder quanto por parte dos outros membros do grupo, além de estarem a 30 unidades de distância do líder (valor adotado previamente). Futuras melhorias no controlador podem ser a inclusão mais conceitos da DCN original como memória de navegação, para se reduzir ações de controles “bruscas” e sim uma média das 3 últimas inferências, para se obter uma tendência de movimento nos agentes ou robôs móveis [14]. Em outra via da pesquisa. Testar os controladores em cenários mais complexos, como por exemplo, com obstáculos fixos ou móveis [27].

V. CONCLUSÃO

Com os resultados das simulações, obteve-se um Sistema de Navegação Autônomo Móvel em grupo, capaz de desviar de obstáculos e navegação em formação, seguindo um líder. O trabalho atingiu os objetivos previstos, navegar em grupo seguindo um líder desviando de obstáculos em pelo menos 3

cenários diferentes para sugerir autonomia nos controladores. No entanto, foi possível perceber que o sistema necessita de algumas melhorias. Como por exemplo, aperfeiçoamento do modelo cinemático de simulação dos robôs utilizados, como por exemplo, para um modelo com pulsos em duas rodas; objetivando maior realismo e precisão nos movimentos dos robôs ou agentes.

A vantagem do uso desta arquitetura de controladores robóticos é que ela pode ser modelada para que se desenvolvam estratégias com novos objetivos para o grupo, como por exemplo, o de buscar um determinado alvo no cenário, além do desvio de obstáculos e formação em grupo, funcionalidades úteis para aplicações reais, como coleta de minas em aplicações bélicas. Esta flexibilidade se dá devido ao “baixo custo computacional” e a capacidade de se adaptar o Modelo Cognitivo por meio dos conceitos e as relações causais, de seleção entre outras, utilizando-se de conhecimento heurístico. Desse modo, espera-se ter contribuído com uma proposta de um controlador flexível e de baixo custo computacional para aplicações em plataformas robóticas.

Futuros trabalhos, endereçam fazer uma quantificação da complexidade computacional das sDCN. E, comparar os controladores propostos, com outros similares desenvolvidos com técnicas clássicas como por exemplo, a Lógica *Fuzzy*. Aumentar a complexidade dos cenários com inserção de alvos e/ou obstáculos dinâmicos.

E, finalmente, embarcar o sistema em plataformas móveis reais, porém será necessário utilizar por exemplo o uso de câmeras, previamente calibradas, para mapear o ambiente e enviar as posições dos robôs, determinando assim um “plano cartesiano” com coordenadas X e Y. Ou se utilizar sensores nos robôs seria uma das possíveis alternativas para a solução do problema de reconhecimento de distância, como por exemplo, o uso de sensores de luz para diferenciar o robô líder do bando como um alvo móvel no cenário, de acordo com a estratégia utilizada no desenvolvimento da sDCN modelada nesse trabalho.

REFERÊNCIAS

- [1] Bonabeau, E.; Dorigo, M. and Theraulaz, G. , “Swarm intelligence: from natural to artificial systems.” Oxford, NY: Oxford University Press, 1999.
- [2] Brooks, R. A., “A robust layered control system for a mobile robot.” IEEE Journal of Robotics and Automation, v. 2, n. 1, Mar, 1986, p. 14-23.
- [3] Russel, S. J. and Norvig, P. “Artificial intelligence: a modern approach.” Englewood Cliffs: Prentice Hall, 1995.
- [4] Costa E. D. S. and Gouvea, M. M. “Autonomous navigation in dynamic environments with reinforcement learning and heuristic.” Ninth International Conference on Machine Learning and Applications, 12-14 Dez, 2010, pp. 37-42.
- [5] Smithers, M. “Autonomy in robots and others agents”, Brains and Cognition, vol. 34, 1997, pp. 88-106.
- [6] Shaikh, M.H.; Kosuri, K.; Ansari, N.A. and Khan, M.J.”The state-of-the-art intelligent navigational system for monitoring in mobile autonomous robot, “Information and Communication Technology (ICoICT), 2013 International Conference of, pp.405,409, 20-22 (2013)
- [7] Maki, T.; Kume, A.; Ura, T.; Sakamaki, T. and Suzuki, H., "Autonomous detection and volume determination of tubeworm colonies from underwater robotic surveys, “OCEANS 2010 IEEE - Sydney, pp.1, 8, 24-27 May, 2010.
- [8] Schroth, G.; Stork, I.; Wersborg, G. and Diepold, K., "A cognitive system for autonomous robotic welding," Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, pp.3148,3153, 10-15 Oct. 2009
- [9] Salan, S.; Drumwright, E. and King-Ip Lin, "Minimum-Energy Robotic Exploration: A Formulation and an Approach," Systems, Man, and Cybernetics: Systems, IEEE Transactions on , vol.45, no.1, pp.175,182, Jan. 2015
- [10] Spears, A.; Howard, A.M.; Schmidt, B.; Meister, M.; West, M. and Collins, T., "Design and development of an under-ice autonomous underwater vehicle for use in Polar regions," Oceans - St. John's, 2014 , pp.1.6, 14-19 Sept. 2014
- [11] Broggi, A., Zelinsky, A., Parent, M., and Thorpe, C. E.: “Intelligent vehicles. Springer Handbook of Robotics,” pp. 1175–1198. 2008
- [12] Sharkey, A. J. and Sharkey, C. N. “The application of swarm intelligence to collective robots.” Advances in Applied Artificial Intelligence, 2006.
- [13] Pessin, G.; Osório, F. S.; Wolf, D. F. and Brasil, C. R. S. “Improving efficiency of a genetic algorithm applied to multi-robot tactic operation”. Advances in Artificial Intelligence. Lectures Notes in Computer Science. vol. 6433, pp. 50-59, 2010.
- [14] Mendonça .M., “Uma contribuição ao desenvolvimento de sistemas inteligentes utilizando redes cognitivas dinâmicas.” Tese (Doutorado em Engenharia de Automação e Sistema), Universidade Tecnológica Federal do Paraná, Curitiba, Brasil, 2011.
- [15] Iocchi, L.; Nardi, D. and Salerno, M. “Reactivity and deliberation: a survey on multi-robot systems”, Balancing Reactivity and Social Deliberation in Multi-Agent Systems, From RoboCup to Real-World Applications, Springer-Verlag, London, UK, 2001, pp. 9-34.
- [16] Parker, L. E. “Current research in Multirobot Systems”, Artificial Life Robotics, 2003, pp. 1-5.
- [17] Axelrod R., “Structure of decision: the cognitive maps of political elites.” New Jersey: Princeton University Press, 1976.
- [18] Kosko, B. “Fuzzy cognitive maps.” International Journal Man-Machine Studies, vol. 24, no. 1, 1986, pp. 65-75.
- [19] Stylios, C. D.; Georgopoulos, V. C.; Malandraki, G. A. and Chouliara, S. “Fuzzy cognitive map architectures for medical decision support systems.” Applied Soft Computing, vol. 8, no. 3, Jun. 2008, pp. 1243-1251.
- [20] Papageorgiou E.I. and Salmeron J.L. (2013). A Review of Fuzzy Cognitive Maps Research During the Last Decade. Fuzzy Systems, IEEE Transactions on, vol.21, n.1, pp.66–79, Feb.
- [21] Stylios, C. D. and Groumpos, P. P. “Mathematical formulation of fuzzy cognitive maps.” Proceedings of the 7th Mediterranean Conference on Control and Automation (MED99), Haifa, Israel, June 28-30, 1999.
- [22] Stylios, C. D. and Groumpos, P. P. and Georgopoulos, V. C. “An fuzzy cognitive maps approach to process control systems.” J. Advanced Computational Intelligence, no. 5, 1999, pp. 1-9.
- [23] Miao, Y., Z.Q. Liu, C.K. Siew and C.Y. Miao “Dynamical cognitive network — an xxtension of fuzzy cognitive”. IEEE Trans. on Fuzzy Systems, Vol. 9, no. 5, pp. 760-770. 2001
- [24] Koulouriotis, D. E.; Diakoulakis, I. E.; Emiris, D. M. and Zopounidis, C. D. “Development of dynamic cognitive networks as complex systems approximations: validation in financial time series.” Applied Soft Computing, v. 5, n. 2, Jan, 2005, p. 157-179.
- [25] Brooks, R. A. “A robust layered control system for a mobile robot.” IEEE Journal of Robotics and Automation, v. 2, n. 1, Mar, 1986, p. 14-23
- [26] Sutton, R.; Barto, A. “Reinforcement learning: an introduction.” Cambridge: MIT Press, 1998.
- [27] Kluge, B.; Prassler, E., "Reflective navigation: individual behaviors and group behaviors," in Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on , vol.4, no., pp.4172-4177 Vol.4, April 26-May 1, 2004.