

REDES NEURAIS APLICADAS À BALIZA PARALELA EM AMBIENTE SIMULADO

André Carmona Hernandes, andre.hernandes@usp.br
André Salomão Brito Mendonça, erdan20@gmail.com
Kleber de Oliveira Andrade, pdjkleber@gmail.com
Marcelo Becker, becker@sc.usp.br

Escola de Engenharia de São Carlos – EESC-USP, Laboratório de Robótica Móvel, Av. do Trabalhador São-carlense, 400, Pq. Arnold Schimidt, CEP 13566-590, São Carlos – SP.

Resumo: *O interesse por automação veicular vem crescendo ultimamente, e devido ao grande desafio que existe em dotar um sistema móvel com capacidade para tomar decisões a partir de interações com o meio, o uso de sistemas inteligentes pode ser uma boa saída para esse problema. Além disso, a automação em veículos não precisa ser feita de modo a substituir o motorista completamente, mas sim poder auxiliá-lo em algumas tarefas, visando maior conforto e segurança desse e dos passageiros. Com essa motivação em mente, este trabalho tem por objetivo apresentar um controlador baseado em Redes Neurais Artificiais (RNAs) capaz de fazer com que um carro do tipo Sedan, em ambiente de simulação, realize a manobra de baliza paralela. Esse controlador recebe dados simulados de dois sensores SICK localizados na parte frontal e traseira do veículo. Esses dados servem de entrada para o controlador que é capaz de interpretá-los e de responder a esses estímulos de maneira correta. A busca por uma resposta adequada foi feita através de 3 abordagens de estruturas de dados e testadas em 3 tipos de arquiteturas de redes. A eficácia desse controlador foi posta à prova em ambiente de simulação até responder satisfatoriamente aos dados de entrada. A importância desse trabalho reside na idéia de ser o primeiro passo para a utilização desse tipo de sistema em modo embarcado a ser realizado futuramente pelo projeto SENA da EESC USP.*

Palavras-chave: *Robótica Móvel, Sistema Inteligente, Redes Neurais Artificiais, Cinemática Veicular, Carros autônomos*

1. INTRODUÇÃO

Atualmente, a automação veicular tem sido foco de muitas pesquisas, devido ao desafio que existe em dotar um sistema móvel com capacidade para tomar decisões a partir de interações com o meio ambiente. Veículos autônomos devem ser capazes de receber estímulos através de sensores e planejar, assim, suas ações. Essas qualidades permitem que estes sistemas possam ser utilizados para diversas funções (HEINEN et al, 2006).

Ainda segundo Heinen et al (2006), a automação em veículos não precisa ser feita de modo a substituir o motorista completamente, mas sim para poder auxiliá-lo em algumas tarefas ou realizar tarefas simples no veículo visando aumentar o conforto e segurança desse e dos passageiros. Essas tarefas vão desde, travar as portas quando o carro passar de uma velocidade determinada, a, até mesmo, indicar ao motorista a distância segura que deve estar de um veículo à frente. Alguns exemplos de veículos autônomos podem ser citados como: o sistema criado pela NavLab da CMU (Carnegie Mellon University), capaz de conduzir um automóvel pelas estradas estadunidenses; os robôs enviados para explorar o planeta Marte; o robô Dante explorador de vulcões entre outros. O desenvolvimento de veículos autônomos é também almejado na área militar, isso pode ser observado no DARPA Grand Challenge, patrocinado pelas forças armadas norte-americanas.

Muitas empresas têm investido em sistemas de automação capazes de realizar ou auxiliar o motorista em pequenas manobras, como por exemplo: o Parking Assistant System do Toyota Prius; o sistema de estacionamento instalado nos Sedan Lincoln MKS e no novo modelo de luxo Lincoln MKT da FORD; e o sistema do C4 Picasso da Citroën, que indica se o automóvel cabe ou não em uma determinada vaga.

A complexidade dos movimentos que devem ser realizados e o grande número de variáveis que devem ser levadas em consideração tornam essa tarefa desafiadora. Assim, a utilização de sistemas inteligentes dotados da capacidade de aprendizado, como as Redes Neurais Artificiais (RNA), pode ser uma boa tentativa de superar essa dificuldade, pois esses sistemas são capazes de se adaptar a várias situações, lidando bem com um número muito grande de fatores.

Trabalhos realizados por Osório, Heinen e Fortes (2002), Heinen, Osório e Heinen, (2005) e Heinen et al (2006), conseguiram utilizar controladores baseados em RNAs para realizar manobras de baliza paralela em um veículo, através de simulações 2D e 3D. No presente trabalho, foi desenvolvido um controlador baseado em RNAs o qual realiza a manobra de baliza paralela em um ambiente de simulação. Esse controlador recebia dados de dois sensores simulados SICK LMS 2XX, a partir dos quais decidia e gerava comandos de acionamento de volante e aceleração do veículo. O controlador cumpriu com o objetivo de completar a manobra de baliza paralela sem bater nos obstáculos ao redor.

2. METODOLOGIA

A manobra de baliza paralela é o método de estacionar um veículo em linha com outros veículos estacionados formando uma linha paralela ao meio-fio. Essa manobra, exigida nos exames de motoristas em muitos países, é considerada umas das mais difíceis de aprender por parte de muitos motoristas iniciantes. Apesar das dificuldades dos jovens condutores, percebe-se que esta manobra é composta de três passos ou etapas bem distinguíveis entre si e cada uma delas é caracterizada por uma posição fixa do volante acompanhada de um do movimento fixo do carro (ou para frente ou para trás). Tais passos, que a partir de agora nomear-se-ão de etapas, são descritos a seguir.

A primeira das etapas se inicia quando o carro alinha-se paralelo ao lado do veículo à frente da vaga. Nessa etapa, o motorista gira o volante totalmente para a direita (no caso de vagas à direita do motorista), e engata a ré, fazendo com que a traseira do carro entre na vaga até percorrer aproximadamente metade da distância entre a posição inicial do carro e o meio-fio. A sequência de imagens da Fig. (1) representa os instantes dessa etapa realizadas por um motorista.



Figura 1. Sequência de imagens ilustrando o primeiro passo na baliza paralela.

A meio caminho do meio-fio, tem-se início a segunda etapa da baliza, que consiste em girar o volante totalmente para a esquerda e manter a ré engatada, nessa parte da manobra, o carro continua entrando na vaga e começa a se alinhar paralelamente ao meio-fio. A sequência de imagens da Fig. (2) mostra os instantes dessa manobra.



Figura 2. Sequência de imagens trazendo instantes da segunda etapa

A terceira e ultima etapa da manobra de baliza paralela tem início quando o carro se alinha à vaga paralelamente ao meio fio. Essa parte da manobra tem como objetivo, manter o carro próximo ao meio da vaga, assim as ações nessa etapa consistem em engatar a primeira marcha, mantendo o volante na posição neutra (com pneus retos) fazendo o carro se mover apenas para frente. A sequência de imagens da Fig. (3) traz instantes dessa manobra.

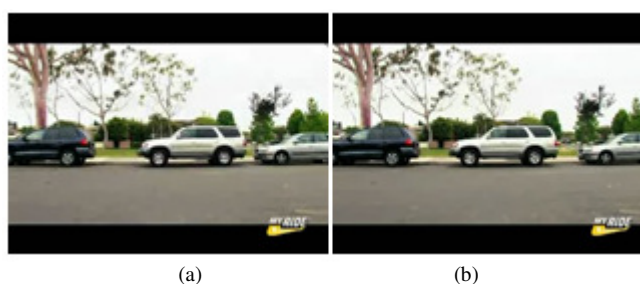


Figura 3. Sequência de imagens exemplificando a ultima etapa da baliza

Poder dividir a manobra de baliza paralela em três passos ou etapas bem distintas entre si é bastante relevante, pois afeta diretamente o tipo de saída que a RNA apresentará a suas entradas, já que os comandos do volante e do controle de velocidade não se alteram durante cada etapa, pode-se atribuir a rede o papel de decidir em que etapa da manobra se encontra o carro, simplificando a saída da RNA e tornando esta apenas um classificador.

Para a RNA conseguir classificar em qual etapa se encontra, é preciso selecionar as entradas necessárias para a mesma realizar tal tarefa. Assim, o uso de sensores que podem fornecer informações sobre meio em que o carro se encontra se faz necessário.

O sensor SICK LMS 2XX, usado na simulação, varre uma área em leque de 180° a sua frente e retorna um vetor-resposta de 181 posições de medidas de distância. Deste modo, o uso de dois desses sensores, um em cada extremidade do carro seria conveniente para o trabalho. Entretanto, se a saída do sensor fosse ligada diretamente à entrada da RNA, ter-se-ia uma rede com mais de 180 entradas por sensor, contudo boa parte da informação coletada pelos sensores não é significativa para a manobra de baliza, o que poderia induzir erros à rede.

Logo, foi necessária a filtragem de dados da saída do sensor antes de serem usadas como entradas da RNA. Considerou-se, então, significativos para a RNA os cantos ou linhas definidos pelos veículos que delimitavam a vaga da baliza. Optou-se por representar cada um desses objetos (cantos e linhas) por três pontos em coordenadas cartesianas (x e y).

Assim, considerando-se dois objetos significativos para a RNA, a entrada da rede seria composta por 12 coordenadas, sendo seis delas representando o veículo da frente e as outras seis, o veículo de trás. Então, um filtro para os dados do sensor teve de ser usado, convertendo os dados em coordenadas polares do sensor para coordenadas cartesianas com eixos no centro de gravidade do veículo. Com essas informações em mente, foi possível desenvolver o simulador, baseado em Becker *et al* (2009) com a finalidade de testar o controlador em RNAs. Esse simulador foi modificado para tornar-se mais leve e simples, algumas funções novas foram desenvolvidas e outras foram modificadas ou retiradas. As Figuras (4) e (5) trazem uma visão geral do simulador após as modificações. Algumas funções atuantes dentro da simulação foram modificadas e outras criadas, a fim de facilitar a aquisição de dados, o controle do veículo, os testes e os treinamentos.

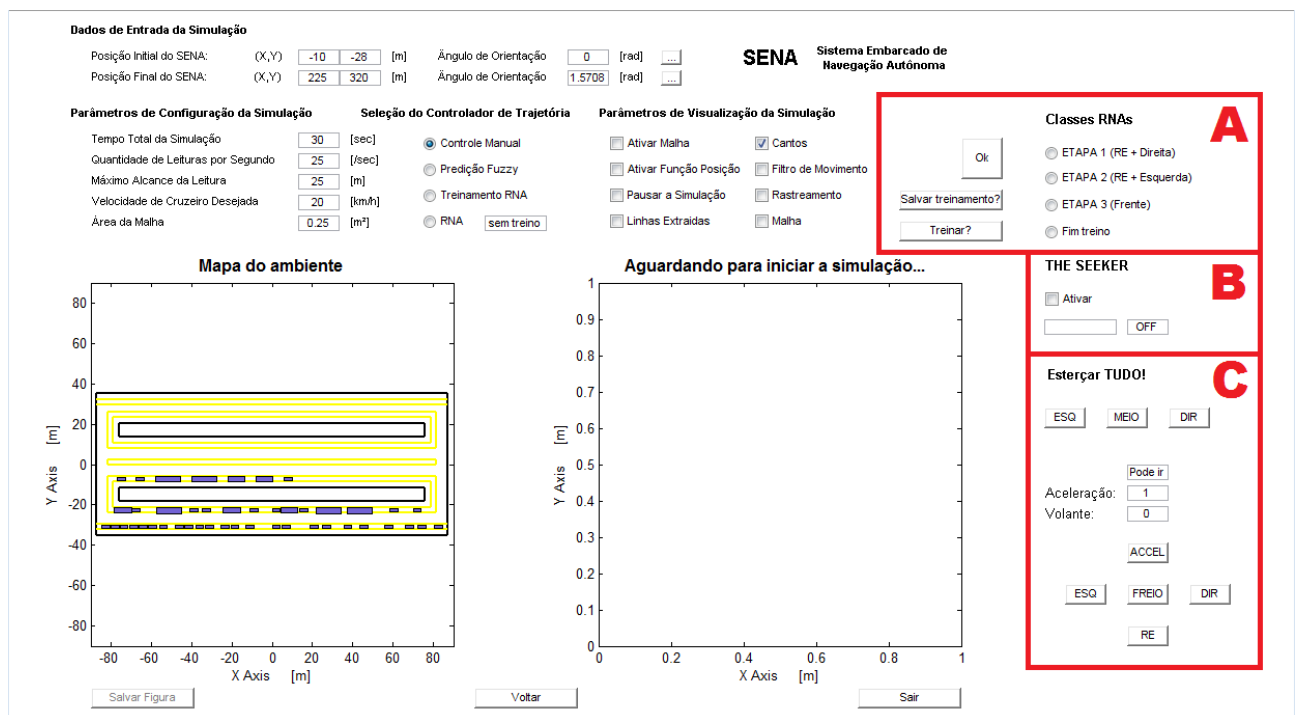


Figura 4. Tela Principal do Simulador Modificado. A: Interface para o treinamento da RNA. B: Interface para a ativação da função “buscador de vagas”. C: Interface para o controle manual do carro, possibilitando o acionamento da direção e da seleção da velocidade

O ambiente de simulação desenvolvido representava ruas de diversos tamanhos todas com área de estacionamento com veículos de vários portes estacionados com distâncias variáveis entre si. Um veículo em baixa velocidade, o qual parte de uma posição e orientação escolhida pelo usuário, busca uma vaga de tamanho suficiente para que a manobra de baliza possa ser realizada em apenas 3 etapas. Quando encontrada, o veículo alinha-se automaticamente na posição inicial da manobra de baliza e, em seguida, o controlador em RNAs é acionado para realizar a manobra.

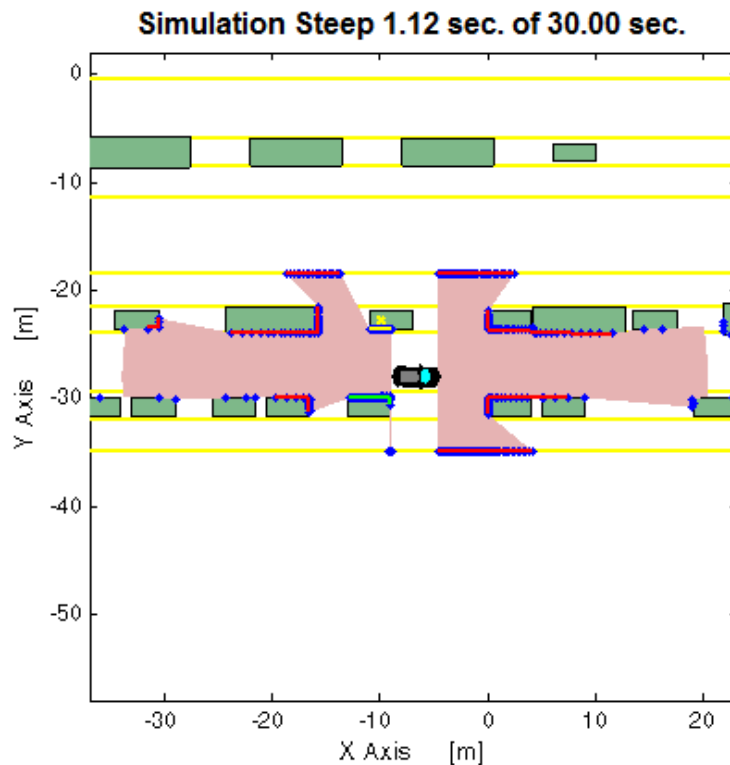


Figura 5. Tela de Simulação Modificada

Analisando-se a leitura dos sensores SICK, percebeu-se que os objetos identificados variam muito, tanto em quantidade como em tamanho o que poderia induzir erros a RNA. Usando funções que imitam a odometria do carro foi possível, assim que uma vaga é detectada, selecionar quatro pontos, considerados os limitantes da área dos dados de entrada. Deste modo, os objetos identificados pelo sensor deveriam sempre estar contidos ou parcialmente contidos nessa região, reduzindo a possibilidade de perder uma referência ou tomar referências errôneas. Utilizou-se a odometria do carro para realizar uma série de mudanças de coordenadas desses pontos em relação ao veículo, sendo possível assim, manter essa área fixa no espaço. A área delimitada e os pontos dos objetos identificados dentro dessa área estão mostrados na Fig. (6).

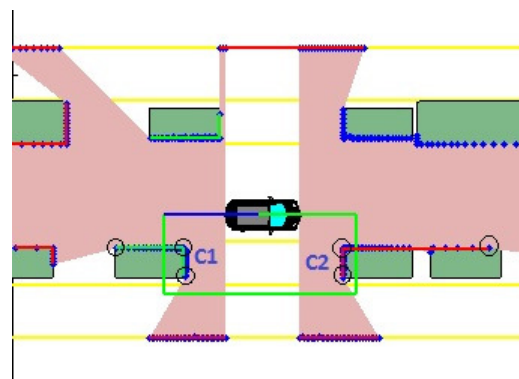


Figura 6. Retângulo delimitando área de significancia e cantos identificados nessa área.

Três abordagens foram tomadas quanto à formalização dos dados para servirem de entrada para o treinamento da RNA. Na primeira delas, foram considerados, durante toda a trajetória, os dados apenas dos pontos encontrados na região de interesse, como apresentado na Fig. (6), a matriz resultante dessa abordagem será referenciada como Matriz Normal.

Esperando-se que o controlador encontrasse uma ordem de mudança das etapas, considerou-se, como uma segunda abordagem, os valores da etapa anterior em que se encontrava o carro, juntamente com os valores dos pontos significativos durante toda a trajetória. A matriz gerada a partir dessa abordagem recebeu o nome de Matriz de Etapa Anterior. As ilustrações da Fig. (7) mostram a diferença entre essas estruturas.

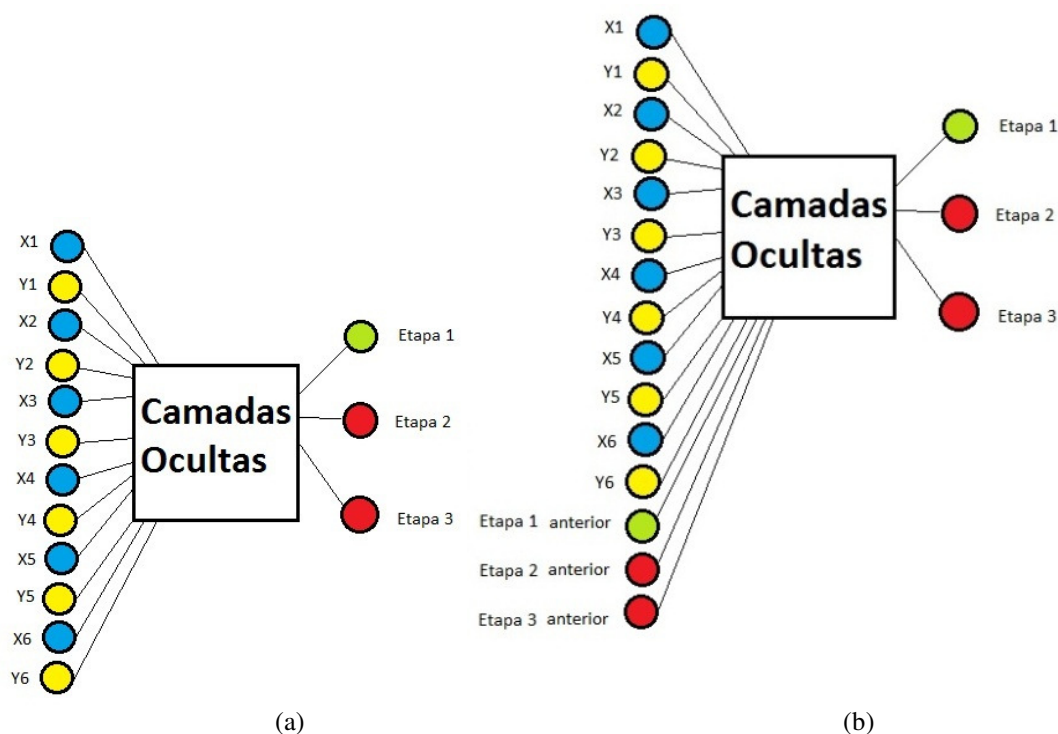


Figura 7. Esquema ilustrando as entradas e saídas da RNA sendo (a) com a Matriz Normal e (b) com a Matriz de Etapa Anterior

Acreditando-se que a utilização dos pontos obtidos durante toda a trajetória (como feito nas abordagens anteriores) poderia, de alguma forma, levar a RNA a uma má classificação das etapas, devido a proximidade entre o último ponto de uma etapa e o primeiro ponto da etapa seguinte. Considerou-se, na última abordagem, os 10 primeiros valores de cada etapa, ou seja, o primeiro valor apresentado com a mudança de etapa e seus 9 valores subsequentes, a matriz gerada possui uma estrutura de dados semelhante à Matriz de Etapa Anterior, e foi referenciada como Matriz de Gatilho.

Por fim, a arquitetura da rede também influencia muito em sua resposta, bem como o treinamento. Deste modo, algumas redes de múltiplas camadas alimentadas adiante foram testadas. Cada RNA era testada pelas 3 abordagens acima apresentadas, fornecendo assim um modo de comparação dos desempenhos obtidos. A função de ativação utilizada em todos os testes foi a função tangente hiperbólica.

Um método de segunda ordem foi escolhido para o treinamento, fornecendo assim, treinamentos mais rápidos. O método utilizado foi o algoritmo de *Levenberg-Marquart* (HAGAN e MENHAJ, 1994). A análise do desempenho foi feita em cima do erro gerado por cada RNA, assim como pela matriz de confusão de cada uma, apresentadas na seção de resultados.

3. RESULTADOS

A análise dos resultados teve por início a verificação do desempenho de cada rede perante o treinamento realizado. Cada grupo de treinamento foi separado em três partes, 80% do grupo de treinamento seria para o treinamento da RNA em si, 10% para a validação e 10% para testes. Para cada uma das abordagens acima, foram testadas redes com arquitetura com duas, três e quatro camadas ocultas. A Tabela (1) mostra o erro cometido pelas redes testadas, onde estão as três abordagens descritas na metodologia: a Matriz Normal (MN), a Matriz de Etapa Anterior (ME) e a Matriz de Gatilho (MG).

Tabela 1. Desempenho das RNAs testadas.

RNAs	MN			ME			MG		
	2	3	4	2	3	4	2	3	4
Treinamento	10^{-2}	10^{-10}	10^{-2}	10^{-1}	10^{-10}	10^{-10}	10^{-1}	10^{-11}	10^{-11}
Validação	10^{-2}	10^{-2}	10^{-2}	10^{-1}	10^{-5}	10^{-11}	10^{-1}	10^{-11}	10^{-11}
Teste	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-1}	10^{-11}	10^{-11}

A análise da Tab. (1) mostra que a matriz de gatilho consegue aproximar melhor a resposta se comparada com a Matriz Normal, entretanto, uma análise das matrizes de confusão deve ser feita, visto que trata-se de um problema de

classificação. As Figuras (8), (9) e (10) mostram as matrizes de confusão para as RNAs. Cada figura analisa um tipo de RNA.

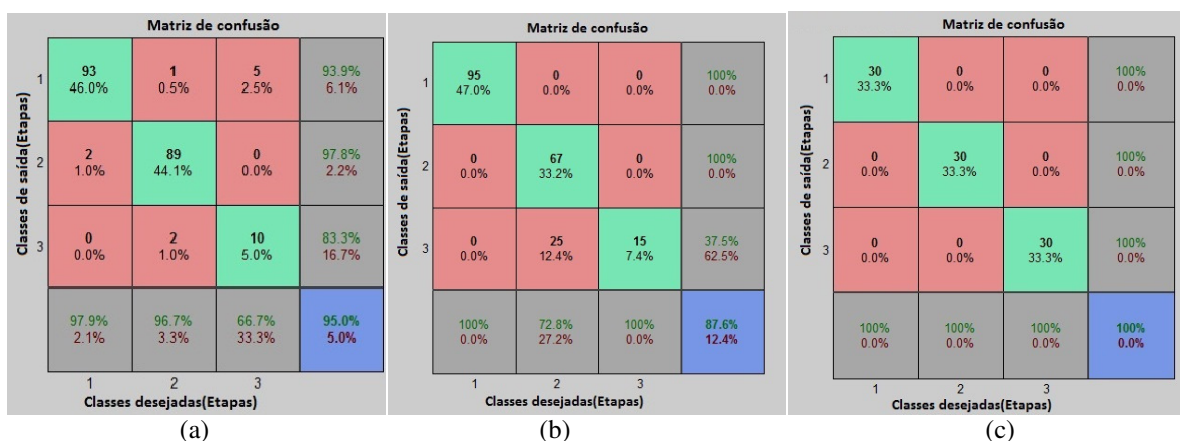


Figura 8. Matriz de Confusão da RNA com duas camadas ocultas. (a) Matriz Normal. (b) Matriz de Etapa Anterior. (c) Matriz de Gatilho.

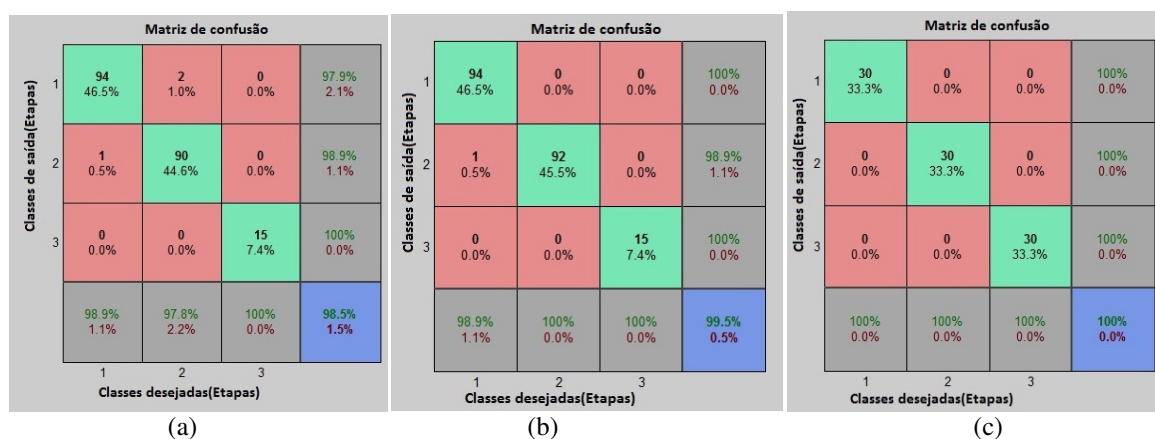


Figura 9. Matriz de Confusão da RNA com três camadas ocultas. (a) Matriz Normal. (b) Matriz de Etapa Anterior. (c) Matriz de Gatilho.

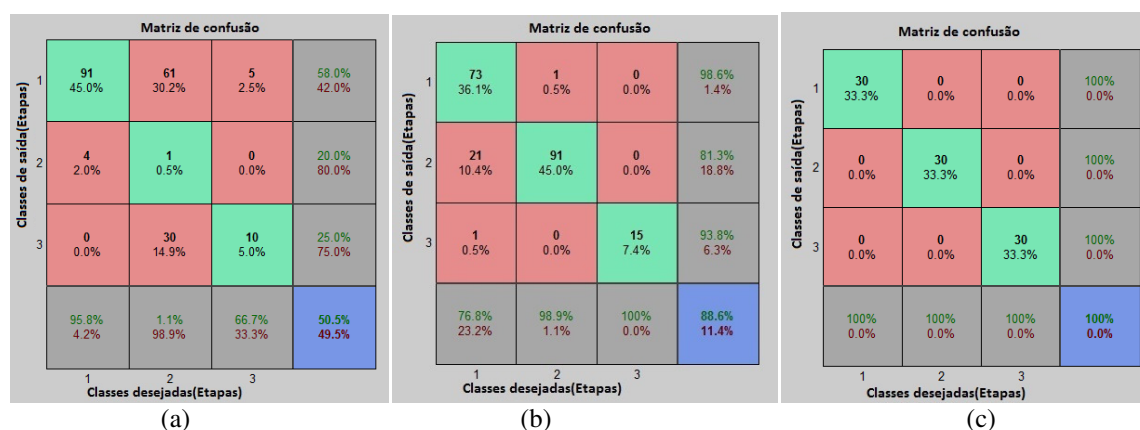


Figura 10. Matriz de Confusão da RNA com quatro camadas ocultas. (a) Matriz Normal. (b) Matriz de Etapa Anterior. (c) Matriz de Gatilho.

Assim, foi possível ratificar que a Matriz de Gatilho, como verificado pela Tab. (1), apresenta a melhor classificação dos dados apresentados a RNA. Considerando esses resultados, a Matriz de Gatilho foi escolhida como o modelo de matriz de entrada usado para o treinamento da RNA. Entretanto, a rede deve apresentar um funcionamento ótimo em ambiente de simulação e para isso, o grau de generalização deve ser o melhor possível.

Apenas os dados de desempenho e da matriz de confusão não são suficientes para garantir a resposta desejada dada pelo controlador no simulador. Tais dados foram úteis para excluir redes que provavelmente não dariam boas respostas

na simulação. As RNAs restantes foram testadas em ambiente de simulação, desde modo, apenas a RNA com 3 camadas ocultas, treinada com a matriz de gatilho foi capaz de estacionar o carro, como mostra a Fig (11).

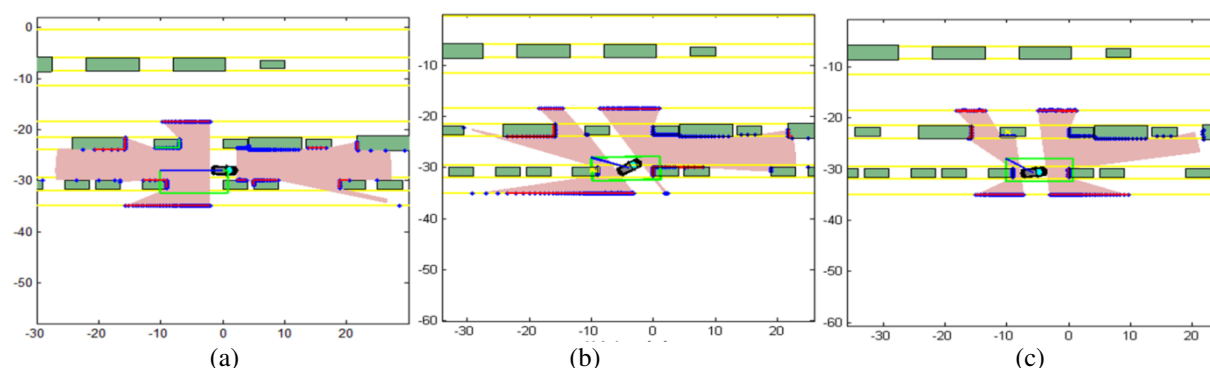


Figura 11. Realização das etapas pelo controlador RNA: (a) início da etapa 1, (b) início da etapa 2, (c) início da etapa 3.

4. CONCLUSÃO

Após a realização do presente trabalho, comprovou-se, em ambiente de simulação, que um controlador baseado em RNA, alimentado com dados fornecidos por dois sensores SICK, é capaz de controlar um veículo na manobra de baliza paralela.

Um grande ganho da RNA é sua capacidade de se tornar um controlador robusto, contra uma solução clássica. Ou seja, um controlador baseado em RNAs seria capaz de responder satisfatoriamente a casos não apresentados e deste modo, sendo mais efetivo que uma solução clássica que deveria se preocupar com todos os detalhes que podem variar a forma de executar o controle.

Ressalta-se a importância do pré-processamento dos dados de entrada do controlador, devido a sua influência direta no aprendizado da RNA. Haja vista a diferença nos resultados dos treinamentos de RNAs de arquiteturas equivalentes apenas substituindo-se os dados de entrada.

Acredita-se que o controlador descrito no presente trabalho possa ser embarcado em um veículo real, entretanto recomenda-se modificações e/ou melhoramentos em quesitos como substituição de sensor SICK traseiro, melhoramento na função localizadora de vagas e o tamanho do grupo de treinamento.

Quanto a questão da substituição do sensor SICK localizado na parte traseira do veículo, esse pode ser trocado por sensores de ultrassom, os quais já são bastante utilizados em veículos comerciais. Tal troca acarretaria em uma mudança nos dados de entrada do controlador, havendo a necessidade de mais adaptações, tanto no que toca a RNA como no pré-processamento dos dados.

É possível realizar melhoramentos na função localizadora de vagas, implementando a detecção de vagas em ambos os lados do veículo e também a identificação de vagas sem ser necessária a utilização da odometria do veículo. Essa função exige ainda um algoritmo mais robusto e resistente a erros, devido a sua importância em alinhar o veículo à vaga.

Por fim, há a necessidade de um conjunto de treinamento significativamente maior, pois a RNA no ambiente embarcado deve ter uma ótima generalização, visto que em ambiente simulado há a presença de simplificações do ambiente real.

Indica-se ainda a possibilidade de se inserir na rede dados sobre o tempo ou número de interações que a rede executou a fim de melhorar ainda mais seu desempenho.

5. AGRADECIMENTOS

Ao professor Fernando Santos Osório do Instituto de Ciências Matemáticas e de Computação (ICMC) da Universidade de São Paulo (USP) campus de São Carlos pelas sugestões na realização do trabalho. À FAPESP e ao CNPq pelo apoio financeiro. Ao INCT SEC - Sistemas Embarcados Críticos e INCT INOF – Instituto Nacional de Óptica e Fotônica.

6. REFERÊNCIAS

BECKER, M.; HALL, Richard; KOLSKI, Sascha; MACEK, Kristijan; JENSEN, Bjoern; SIEGWART, Roland . 2D Laser-based Probabilistic Motion Tracking in Urban-like Environments. Journal of the Brazilian Society of Mechanical Sciences and Engineering (Impresso), v. 31, p. 83-96, 2009.

HAGAN, M.T.; MENHAJ, M.. "Training feed-forward networks with the Marquardt algorithm," IEEE Transactions on Neural Networks, Vol. 5, No. 6, 1999, pp. 989-993, 1994.

HEINEN, Milton et al. Estacionamento de um Veículo de Forma Autônoma Utilizando Redes Neurais Arti. In: CONGRESSO DA SBC, 26., 2006, Campo Grande, Ms. Anais do XXVI Congresso da SBC. Campo Grande: SBC, 2006. p. 107 - 116.

HEINEN, Milton Roberto; OSÓRIO, Fernando Santos; HEINEN, Farlei Jose. Estacionamento de um veículo de forma autônoma simulado em um ambiente tridimensional realístico. In: SEMINCO, 14., 2005, Blumenau. Anais... . Blumenau: Furb, 2005. v. 1, p. 91 - 101.

OSÓRIO, F.; HEINEN, F.; FORTES, L.. Controle da Tarefa de Estacionamento de um Veículo Autônomo através do Aprendizado de um Autômato Finito usando uma Rede Neural J-CC. In: SIMPÓSIO BRASILEIRO DE REDES NEURAIAS, 7., 2002, Porto de Galinhas. Anais do SBRN.Recife: SBC, 2002.

NEURAL NETWORKS APPLIED TO PARALLEL PARKING IN A SIMULATED ENVIRONMENT

André Carmona Hernandes, andre.hernandes@usp.br

André Salomão Brito Mendonça, erdan20@gmail.com

Kleber de Oliveira Andrade, pdjkleber@gmail.com

Marcelo Becker, becker@sc.usp.br

Escola de Engenharia de São Carlos – EESC-USP , Laboratório de Robótica Móvel, Av. do Trabalhador São-carlense, 400, Pq. Arnold Schimidt, CEP 13566-590, São Carlos – SP.

Abstract. *The interest in vehicular automation is growing lately, and due to the existing challenge of providing to a mobile system the capacity of taking decisions from interactions with the environment, the use of smart systems can be a good solution for this problem. Moreover, automation in vehicles don't need to be made in a way to completely replace the driver, but to assist him in some tasks in order to increase comfort and safety to the driver and passengers. With this motivation in mind, this paper aims to present an Artificial Neural Networks (ANN)-based controller capable of perform a parallel parking with a Sedan-type car in a simulation environment. This controller receives simulated data from two SICK sensors, located at the front and rear of the vehicle. These data are used as input to the controller, which is able to analyze them and respond to these stimuli correctly. The search for an appropriated response was made through three data structures approaches and tested in three network-architecture types. The effectiveness of this controller was tested in simulated environment until respond satisfactorily to the input data. The importance of this work lies in the idea of being the first step for using this type of embedded system, in order to be held in future by SENA project of the EESC USP.*

Keywords: *Mobile robotics, Smart system, Artificial Neural Networks, Vehicles' Kinematics, Autonomous cars.*