

Implementação e verificação formal de estratégias para desvio de obstáculos de veículos autônomos modelados como agentes racionais

Lucas Emanuel Ramos Fernandes
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Brasil
lucfer@alunos.utfpr.edu.br

Vinicius Custodio
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Brasil
viniciuscustodio@alunos.utfpr.edu.br

Gleifer Vaz Alves
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Brasil
gleifer@utfpr.edu.br

Resumo

Em um futuro não tão distante, veículos autônomos serão uma realidade no tráfego urbano. Empresas como Google, Tesla e Uber estão realizando pesquisas para o desenvolvimento de um software capaz de tomar decisões próprias sem intervenção humana. Uma possível abordagem para a criação dessa autonomia necessária para controlar o veículo é por meio da implementação de agentes racionais. Neste cenário, o agente deve ter a capacidade de guiar um veículo e decidir rotas a serem seguidas. Esta tarefa aparentemente simples se torna complicada quando se trata do desvio de obstáculos. Sendo, necessário implementar um agente capaz de identificar quais decisões dever ser tomadas nestas situações. Na área de agentes existem diversas técnicas utilizadas para o desvio de obstáculos. Porém, todo software é susceptível à erros, então é necessário validar as decisões tomadas pelo agente, e isto pode ser realizado através do uso da verificação formal. Portanto, este trabalho propõe uma pesquisa sobre a implementação de um agente racional e a verificação formal da tomada de decisão, onde tais agentes representam veículos autônomos em cenários específicos para desvio de obstáculos.

Palavras-chaves

Veículos Autônomos; Agentes Inteligentes; Verificação Formal

Abstract

Autonomous vehicles will soon be a reality in urban traffic. Companies such as Google, Tesla, and Uber are conducting research to develop and spearhead a software capable of making on-demand need-based decisions without human

interaction. One possible approach at creating the needed autonomy is through the implementation of rational agents with the ability to guide a vehicle and decide the routes to be followed. This seemingly simple task becomes far more complicated when dealing with preventative collision measures. These unforeseen events require the implementation of an agent capable of identifying what decision should be made in these unique and immediate situations. Rational multi-system agents have many varying techniques ready for object avoidance adoption; however, all software is prone to errors and therefore these techniques of decision making need to be validated as the "best course of action" in the given situation. Thus, this paper proposes a research on the implementation of a rational agent and a formal verification technique to oversee the decision making process, where such agents represent autonomous vehicles in specific scenarios for obstacle avoidance.

Keywords

Autonomous Vehicles; Intelligent Agents; Formal Verification

1. INTRODUÇÃO

O processo de urbanização presenciado no últimos anos tem alterado o cenário urbano mundial. Mais da metade da população hoje vive dentro das cidades, e a previsão para as próximas décadas é que o número de habitantes cresça ainda mais [18]. O aumento de automóveis em circulação é proporcional ao crescimento populacional nas zonas urbanas, e problemas como congestionamento são intensificados. Logo, é necessária a criação de novas estratégias para a administração do tráfego urbano. Uma das soluções propostas e em destaque nos últimos anos é o veículo autônomo, sendo considerada por muitos o próximo grande avanço da indústria automobilística [22].

Um veículo autônomo é um carro capaz de transitar no perímetro urbano, ou em rodovias, por meio de decisões tomadas pelo software. Os veículos autônomos trazem consigo uma grande mudança em toda a dinâmica da indústria e na forma como os carros são utilizados. Wallace *et al* [22] descrevem em seu trabalho o cenário a seguir, possível através da utilização de carros autônomos:

Imagine que você esteja saindo do trabalho às 6:25 da noite. Você ainda possui diversas tarefas a realizar antes de dormir, e um percurso de 25 minutos de carro até a sua casa. Com seu celular, você requisita um carro que, alguns minutos depois estaciona a sua frente sem motorista. Ao entrar no veículo, você diz "Casa" para indicar seu destino. No percurso, você faz suas atividades restantes, quando chega em casa pode usar o tempo para relaxar e curtir a família. Quando você sai do carro ele se move para o próximo cliente.

O exemplo anterior é um dos benefícios da adoção desta tecnologia, que é baseado em um sistema de partilha de automóveis. Com isso, além de trazer maior comodidade, será possível diminuir a quantidade de veículos em circulação e aumentar a disponibilidade de vagas em estacionamentos. Durante sua vida útil, um carro só está ativo 5% desse tempo. Por meio compartilhamento de carros autônomos este índice pode subir para 75% [8]. Consequentemente, a utilização de menos carros acarretará na diminuição de emissão de dióxido de carbono (CO₂).

Dentre as outras vantagens dos veículos autônomos, temos: (i) O potencial de diminuir o congestionamento nas cidades por meios de predição de tráfego e análise padrões no trânsito [12]; (ii) Jornadas mais rápidas e movimentação entre veículos com o uso comboios sincronizados de automóveis trocando dados [8]; (iii) Possível redução do número de acidentes de trânsito com um software como motorista, uma vez que o fator humano é a causa de 90% dos acidentes no trânsito [19].

Atualmente, existem 33 empresas desenvolvendo projetos para carros autônomos [5], e as referências na área são: Uber, Google e Tesla. O Uber prepara-se para realizar testes com carros autônomos [6], aonde será possível requisitar o carro utilizando o aplicativo, que o buscará e o levará ao destino. A Google possui diversos carros autônomos em fase de teste, aproximadamente 58 carros circulam por cidades dos Estados Unidos [15]. E a Tesla já possui um modelo semi-autônomo em comercialização, o Tesla Model S [20].

Uma das capacidades esperadas em um veículo autônomo é a habilidade de navegação autônomo, ou seja, não é necessário um motorista. A navegação dessa categoria de veículos requer que o automóvel esteja preparado para todas as situações possíveis dentro do trânsito. Além de controlar funções vitais como acelerar, frear e identificar sinais de trânsito, o automóvel estar apto a detectar e desviar de obstáculos em seu trajeto para evitar colisões [16]. Os obstáculos podem ser de diferentes formas e tamanhos, e serem estáticos ou dinâmicos. Pessoas atravessando a rua, animais na rodovia e veículos quebrados no meio do trânsito são possíveis cenários onde o automóvel deve ser capaz de desviar para garantir a segurança de seus passageiros.

O desvio de obstáculos pelo veículo está relacionado com a tomada de decisão do software controlador do mesmo. É necessário que o sistema seja capaz de interpretar o ambiente onde está e decidir qual ação é a melhor opção a ser executada. Logo, é imprescindível a correta compreensão dos dados. Além disso, o software deve ser de qualidade para garantir a segurança do veículo autônomo e seus passageiros.

Para entender como implementar e verificar as decisões por um veículo autônomo, é necessário compreender a tec-

nologia utilizada para o desenvolvimento do software controlador deste sistema. Uma das abordagens para o desenvolvimento do software controlador de veículos autônomos é por meio de agentes. Um agente é uma entidade computacional inserida em um ambiente capaz de tomar decisões por si próprio [23]. Para verificar as decisões realizadas por um agente, é necessário saber por que o agente decidiu executar determinada ação [13]. Então, a utilização de um agente capaz de explicar os motivos que o levou a tomar suas decisões é necessário, e esse é o propósito dos agentes racionais [24]. O desenvolvimento de agentes racionais pode ser realizado com Jason [4] ou Gwendolen [10] (derivada do Jason), ambas linguagens têm sua implementação baseada em Java.

Com base nos conceitos abordados anteriormente, este trabalho tem o intuito de apresentar propostas de pesquisas para implementar e verificar as decisões tomadas por um veículo autônomo, modelado através de um agente racional.

Na seção 2 são explorados os conceitos envolvendo a implementação da navegação do automóvel e as técnicas que podem vir a ser utilizadas. A seção 3 descreve a importância da verificação do agente controlador do veículo autônomo e de qual forma é possível garantir a corretude das ações realizadas pelo mesmo. Por fim, as considerações finais descrevem as conclusões deste trabalho.

2. ELABORAÇÃO E IMPLEMENTAÇÃO DE PLANOS DE DECISÕES PARA DESVIO DE OBSTÁCULOS

A navegação consiste em encontrar um caminho para atingir um determinado objetivo. O caminho deve ser livre de colisões [11], porém isso não é uma tarefa simples, diversos fatores influenciam na navegação. Um aspecto importante que se busca na navegação é que ela seja capaz de desviar de obstáculos. Quando se trata em navegação de veículos isso se torna ainda mais importante e também um fator de risco. Pois, assegurar que veículos encontra um caminho é livre de colisões é garantir a segurança dos ocupantes.

As técnicas do Obstáculos de Velocidade Recíproca [2] e o das Trajetórias Elípticas [9] são duas técnicas de desvio de obstáculos que são planejadas para serem utilizadas em conjunto com agentes e sistemas multiagentes.

O método das Trajetórias Elípticas tem como princípio as leis newtonianas, que através de interações de atração e repulsão o agente consegue desviar de obstáculos. O método Obstáculos de Velocidade Recíproca garante o desvio de forma reativa, ou seja, quando um obstáculo surge, existe um plano a ser executado para se realizar o desvio. Este método é uma extensão do Obstáculos de Velocidade, o qual garante que é possível evitar uma colisão por meio da análise de todas as possíveis velocidades no qual o agente pode colidir.

Para a aplicação das técnicas de desvio obstáculos é preciso realizar uma análise e compreensão dos possíveis cenários e obstáculos. Um cenário inicial considerado será utilizando somente obstáculos estáticos, desconsiderando velocidades e curvas, ou seja o veículo somente se locomove em linha reta. Considerando veículos autônomos, os obstáculos podem ser pedestres, animais, carros e diversos outros, sendo indispensáveis a incorporação deste conhecimento ao agente, para que este tenha a capacidade de reconhecê-los para então tomar a decisão adequada. Um agente com tais capacidades, possivelmente será um agente racional, aonde esse possui

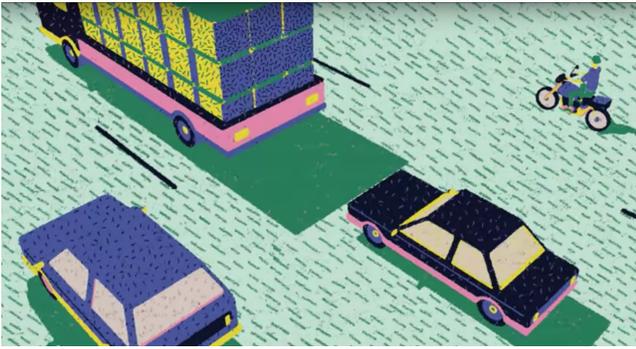


Figura 1: Cenário onde carro autônomo possui um SUV à esquerda, uma motocicleta à direita e um caminhão carregado à sua frente [17].

todos os conhecimentos para evitar obstáculos e e acidente ao mesmo tempo, sendo que uma decisão errada por gerar graves consequências.

Com a definição dos possíveis cenários e tomadas de decisões será possível a implementação do agente no Jason [4] incorporando as tomadas de decisões aos agentes. No Cartago [1] por sua vez será possível a implementação de alguns cenários, escolhendo obstáculos e o ambiente de inserção do agente que foi implementado com o Jason.

Considerando as decisões tomadas será necessário a realização da verificação de desta, para assegurar que são seguras e não suscetíveis a falhas.

3. VERIFICAÇÃO FORMAL DA TOMADA DE DECISÃO DE AGENTES RACIONAIS

Para entender a importância de verificar as decisões tomadas pelo sistema controlador do veículo autônomo, considere o seguinte cenário e a Figura 1, proposto por Lin *et al* [17]:

Um caminhão carregado de madeiras está se movendo na frente de um carro autônomo. Nas faixas ao lado encontram-se uma motocicleta e uma SUV (*sport utility vehicle*, ou carro utilitário esportivo). Uma das torras se desprende do carregamento do caminhão e irá colidir com o veículo. No entanto, não há tempo para frear antes da batida. Agora o automóvel tem três opções: desviar em direção a SUV ou para a moto, ou enfrentar o impacto.

A situação descrita anteriormente, e evidenciada na Figura 1, é uma versão moderna e real do famoso Dilema do Bonde (*Trolley Problem*) proposto por Phillipa Foot [14]. Independentemente da escolha feita em cenários como o anterior, deve-se garantir que o veículo autônomo irá se comportar conforme foi implementado. Como o veículo irá atuar no trânsito e pode causar danos à terceiros e aos seus próprios passageiros, é imprescindível que o software por trás dessa tecnologia funcione corretamente. Então, o automóvel não deve executar uma ação própria de um cenário crítico (e.g., eminência de colisão com outro veículo) em um cenário comum (e.g., deslizar e fazer uma leve curva à direita), ou vice-versa.

Por meio da verificação formal, é possível validar as escolhas feitas pelo agente controlador do veículo autônomo e suas especificações. A verificação formal é utilizada para

garantir a correção do algoritmo utilizado em um sistema com relação as suas propriedades. Esta metodologia é utilizada em sistemas de segurança crítica, onde é imprescindível o desempenho correto do software [7]. O *model checking* é uma das técnicas de verificação formal, que por meio da análise do conjunto finito de estados de um software valida os requisitos do mesmo [7]. Contudo, não é possível determinar com exatidão quais decisões serão tomadas por um agente antes de sua execução. Visser *et al.* [21] utilizando-se do *model checking program* (uma extensão do *model checking*) desenvolveram uma ferramenta denominada *Java PathFinder*, que permite realizar o *model checking* durante a execução do software. Como base no *Java PathFinder*, Bordini *et al.* [3] criaram o *framework Agent JPF*, uma versão específica para agentes racionais.

Assim, a análise e verificação do processo de tomada de decisão de um agente poderá ser realizada por meio do uso do *model checking* e do *model checking program*. E, com o auxílio da ferramenta *Agent JPF*, será possível verificar as estratégias e planos do agente, impondo cenários críticos e cotidianos, com o intuito de validar as especificações do mesmo.

A verificação formal será realizada após a conclusão do desenvolvimento do agente. O agente será exposto às situações para qual foi programado e terá suas tomadas de decisões analisadas.

4. CONSIDERAÇÕES FINAIS

Este trabalho mostra a relevância e importância dos trabalhos para desvio de obstáculos e a verificação das decisões. As tomadas de decisões de um carro são extremamente importantes e com os avanços na área de veículos autônomos tornam-se ainda mais relevantes. Um carro precisa estar preparado para desviar e reagir a todos os obstáculos que o afetam, além de não poder sofrer com defeitos de software. Sendo assim é necessário garantir a segurança das decisões tomadas por meio de métodos formais (*model checking*).

Para trabalhos futuros, temos o desenvolvimento das técnicas de desvio de obstáculos e a verificação formal do mesmo. Inicialmente serão considerados cenários, agentes e planos de decisões simples. Gradualmente será adicionado novas funcionalidades e capacidades aos agentes, e os cenários se tornarão mais complexos. De forma que a complexidade dos planos de decisões aumentará. E, durante todas as etapas, será realizada a verificação formal dos planos de decisões dos agentes.

5. REFERÊNCIAS

- [1] A. O. Alessandro Ricci, Mirko Viroli. Cartago: Framework for prototyping artifact-based environments in mas. *Environments for Multi-Agent Systems*, 4389:67–86, 2006.
- [2] V. D. Berg, M. C. Lin, and D. Manocha. Reciprocal velocity obstacles for real-time multi-agent navigation. *IEEE Conference Robotics and Automation*, pages 1928–1935, 2008.
- [3] R. H. Bordini, L. A. Dennis, B. Farwer, and M. Fisher. Automated verification of multi-agent programs. In *Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering, ASE '08*, pages 69–78, Washington, DC, USA, 2008. IEEE Computer Society.

- [4] R. H. Bordini, J. F. Hübner, and M. Wooldridge. *Programming Multi-Agent Systems in AgentSpeak Using Jason (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.
- [5] CBINSIGHTS. 33 corporations working on autonomous vehicles, 2016.
- [6] M. Chafkin. Uber’s first self-driving fleet arrives in pittsburgh this month, 2016.
- [7] E. M. Clarke, Jr., O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, Cambridge, MA, USA, 1999.
- [8] M. Cliffe. Driverless cars – the route to more than smart cities, 2016.
- [9] B. Dafflon, J. M. Vilca, F. Gechter, and L. Adouane. Adaptive autonomous navigation using reactive multi-agent system for control law merging. In S. Koziel, L. Leifsson, M. Lees, V. V. Krzhizhanovskaya, J. Dongarra, and P. M. A. Sloat, editors, *Proceedings of the International Conference on Computational Science, Computational Science at the Gates of Nature, Reykjavik, Iceland, 1-3 June, 2015, 2014*, volume 51 of *Procedia Computer Science*, pages 423–432. Elsevier, 2015.
- [10] L. A. Dennis and B. Farwer. *Gwendolen: A BDI Language for Verifiable Agents*. University of Aberdeen.
- [11] F. Ducatelle, G. A. D. Caro, A. Förster, M. Bonani, M. Dorigo, S. Magnenat, F. Mondada, R. O’Grady, C. Pinciroli, P. Rétoznaz, V. Trianni, and L. M. Gambardella. Cooperative navigation in robotic swarms. *Swarm Intelligence*, 8(1):1–33, 2014.
- [12] D. J. Fagnant and K. M. Kockelman. Bpreparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Eno Center for Transportation*, 2, 2013.
- [13] M. Fisher, L. Dennis, and M. Webster. Verifying autonomous systems. *Commun. ACM*, 56(9):84–93, Sept. 2013.
- [14] P. Foot. The problem of abortion and the doctrine of double effect. *Oxford Review*, 5:5–15, 1967.
- [15] Google. Self-driving cars reports, 2016.
- [16] J. Levinson. Towards fully autonomous driving: System and algorithms. *Intelligent Vehicles Symposium*, 4:164–128, 2011.
- [17] P. Lin, Y. Du, A. Adkins, K. O’Shea, J. Wang, and C. Misirlioglu. The ethical dilemma of self-driving cars, 2015.
- [18] U. Nations. *Department of Economic and Social Affairs: Population Division. World Urbanization Prospects: The 2014 Revision*. KPMG Center for Automotive Research, 2015.
- [19] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A. A. Hyder, E. Jarawan, and C. Mathers. *World report on road traffic injury prevention*. World Health Organization. 2004.
- [20] Tesla. Tesla models, 2016.
- [21] W. Visser, K. Havelund, G. Brat, S. Park, and F. Lerda. Model checking programs. *Automated Software Engg.*, 10(2):203–232, Apr. 2003.
- [22] R. Wallace and G. Silberg. *Self-Driving cars: The next revolution*. KPMG Center for Automotive Research, 2012.
- [23] M. Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley Sons, 2 edition, 2009.
- [24] M. Wooldridge and A. Rao. *Foundations of Rational Agency*. Kluwer Academic Publishers, 1999.