
Sistema de Semáforos Inteligentes suportado por comunicações V2X

Motivação

Uma rede veicular permite a comunicação dos veículos entre si, espontaneamente, ou com a infraestrutura instalada ao longo das estradas. Atualmente, a maioria dos veículos já vem de fábrica equipado com dispositivos de comunicações e espera-se que num futuro muito próximo, as redes veiculares sejam tão ubíquas quanto o são as redes móveis celulares, atualmente.

Existem vários tipos de aplicações que podem ser usadas nas redes veiculares, que incluem as aplicações que melhoram a segurança rodoviária, as aplicações que melhoram a eficiência e fluidez do tráfego rodoviário e por fim as aplicações de entretenimento e conforto. De entre estes tipos de aplicações, são as relacionadas com a segurança rodoviária, as que mais expectativas têm gerado, graças à diminuição de acidentes e do número de fatalidades associadas que poderão proporcionar. Não obstante, as aplicações que visam melhorar a eficiência do tráfego, possuem também um enorme potencial na melhoria da qualidade de vida de todos os utilizadores das estradas. Este tipo de aplicações inclui as aplicações que controlam interseções e cruzamentos e as que reduzem e evitam o congestionamento de tráfego rodoviário. Dentro do controlo e gestão de interseções e cruzamentos, encontram-se aplicações como os semáforos inteligentes (que até podem ser virtuais), que controlam o fluxo de tráfego do cruzamento de uma forma dinâmica. Com este tipo de semáforos, além de controlar o tráfego, é possível diminuir o congestionamento, uma vez que estas aplicações têm a capacidade de se adaptar às condições do trânsito em tempo real e decidir qual a melhor opção a ser tomada para evitar grandes congestionamentos. Se for possível recorrer às comunicações V2X, os semáforos físicos podem inclusive não ser necessários, já que a ordem de avançar ou parar num cruzamento, pode ser transmitida por mensagens.

Objetivos

Neste trabalho, pretende-se contruir um protótipo funcional de um sistema de semáforos inteligentes suportado por comunicações veiculares. Os veículos enviam mensagens com dados sobre a sua mobilidade, que são recebidas não só pelos veículos seus vizinhos, como também pelas unidades de comunicação fixas instaladas nos cruzamentos/interseções. Ou seja, numa fase inicial do trabalho, os semáforos físicos continuarão a existir. Com base nas mensagens recebidas dos veículos, os semáforos devem apresentar verde ou vermelho de forma a melhorar a fluidez do tráfego. Este semáforo poderá ser facilmente removido se devolver essa informação aos veículos em forma de mensagens, a tempo destes, poderem agir em conformidade (parar ou passar, em segurança, no cruzamento/interseção). No entanto, o sistema continuará dependente de um semáforo (embora virtual), centralizado. Numa fase mais avançada do trabalho, o semáforo virtual poderá ser implementado de forma distribuída, pelos próprios veículos. Nessa situação, a decisão de passar ou parar num cruzamento deverá ser tomada pelos próprios veículos com base nas mensagens que recebem dos veículos à sua volta. Outra melhoria possível que poderá ser implementada na fase final é a coordenação entre os vários semáforos de uma zona, de forma a melhorar ainda mais a fluidez do tráfego.

O foco principal do trabalho está essencialmente na arquitetura global do sistema como um todo, com destaque para comunicações entre as várias entidades envolvidas e nos algoritmos a implementar pelos semáforos inteligentes.

Descrição geral

A Figura 1 mostra os componentes fundamentais da rede veicular a considerar. Os veículos (V_i) estão equipados com unidades de comunicação, designadas por OBU (*On-Board Unit*). Alguns, como o V1, V2, V3 e V4, estão no alcance rádio da unidade de comunicação no cruzamento/interseção, designada por RSU (*Road-Side Unit*). Outros, como o V5 e V6 estão fora de alcance da RSU e só comunicam com os vizinhos. V6 só consegue trocar mensagens com o V5, enquanto o V5 tem 3 veículos no seu raio de alcance: V3, V4 e V6. Se for necessário enviar mensagens do V6 para a RSU, é preciso que outros veículos no percurso (V5 e V3) reenviem as mensagens, ou seja, que atuem como routers.

Junto ao RSU, há uma pequena infraestrutura local, com um *switch*, um servidor local (*mini-cloud*) e um *router* para ligação ao resto da rede fixa. É no servidor local que o algoritmo do semáforo terá de ser implementado. As decisões tomadas pelo semáforo devem ter como objetivo otimizar a fluidez do tráfego, assim, as vias com mais veículos devem ter prioridade, ou pelo menos mais tempo de “sinal verde”. No entanto, os veículos que circulam nas vias menos movimentadas não devem ficar demasiado tempo à espera.

Este trabalho apresenta vários desafios e deve ser implementado em fases. Na fase inicial, podemos contar com o semáforo físico e usar as comunicações apenas para que os veículos transmitam ao semáforo a sua presença, ou seja, nesta fase as comunicações funcionam apenas como um sensor. Com base nesta informação, o servidor determina quem pode passar (acendendo o verde) e quem deve esperar (acendendo o vermelho).

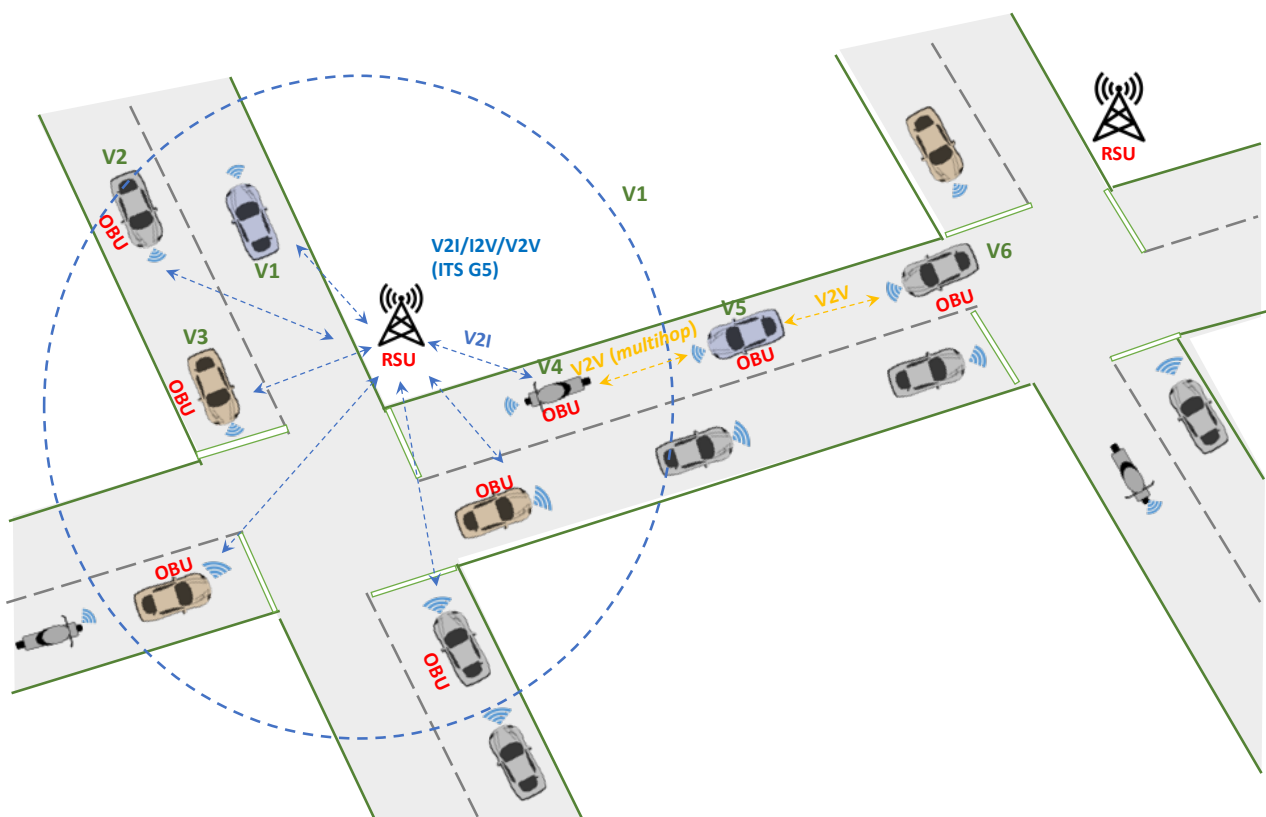


Figura 1: Visão geral da rede veicular e seus componentes principais

Na segunda etapa deste trabalho podemos tentar aumentar a perceção que o semáforo físico tem do ambiente à volta. Para isso é necessário construir um protocolo de encaminhamento multihop, ou seja, os veículos, além de originarem mensagem dando conta da sua presença e localização, encaminham as mensagens que recebem dos seus vizinhos. Obviamente que este encaminhamento tem de ser de alguma forma controlado para não sobrecarregar demasiado a rede. Assim, o RSU do nosso exemplo, além de conhecer os veículos V1, V2, V3 e V4 que estão dentro do seu raio de alcance, poderá ficar a conhecer também o V5 e até eventualmente o V6. Desta forma poderá tomar melhor decisões, relativamente à coordenação do tráfego no cruzamento.

Na terceira etapa pretende-se substituir o semáforo físico por um semáforo virtual, mas mantendo a implementação da lógica associada ao funcionamento do semáforo, no servidor que está associado ao RSU. Ou seja, o que será necessário implementar nesta fase é o envio de mensagens por parte do RSU, que darão ordens a alguns veículos para avançar (no caso daqueles que terão obtido o “sinal verde”) e aos outros para parar (no caso dos que obtiveram o sinal “vermelho”).

Finalmente na última etapa, pretende-se melhorar o funcionamento do sistema de semáforos.

Requisitos mínimos

- cada veículo deverá coligir informações a respeito de si próprio (tipo de veículo, dimensões, peso, etc), do seu movimento (posição, velocidade, aceleração, direção, etc), do estado dos seus sensores (piso molhado, nevoeiro, etc)
- cada veículo deverá executar uma app que envie periodicamente ou quando se justificar mensagens sobre o seu estado
- cada veículo deverá estar sempre pronto a receber mensagens de informação de outros veículos e mensagens com o estado do semáforo
- cada RSU no cruzamento/interseção deve estar sempre pronto a receber mensagens de informação dos veículos
- cada RSU no cruzamento/interseção deve enviar mensagens com ordens de parar ou avançar para os veículos que estão a chegar ao cruzamento/interseção
- as mensagens a enviar resultam do processamento realizado no servidor local diretamente ligado ao RSU no cruzamento/interseção
- a área de intervenção das unidades de comunicação e processamento nos cruzamentos/interseções pode ser alargada com recurso a encaminhamento adequado na rede veicular, a mais que um salto

Cada grupo deverá estender os requisitos de forma que considerar adequada ao sistema de semáforos que vai implementar.

Descrição detalhada

Nesta secção propõe-se um conjunto de 3 etapas, com objetivos bem claros em cada etapa, bem como uma sugestão de atividades que podem ser desenvolvidas para atingir esses objetivos. Além destas etapas, deixamos duas sugestões de melhoria que poderão ser implementadas opcionalmente, numa etapa final.

Etapla 1 – Protótipo simplificado do serviço limitado a um único semáforo físico e ao raio de alcance deste

Nesta etapa, espera-se poder testar uma versão simplificada do sistema. Os veículos limitam-se a adquirir os dados e a enviar por *multicast* a um salto de distância. É preciso especificar a informação a enviar, o formato e a periodicidade adequados. Os dados que a RSU conseguir receber do seu raio de ação, devem coligidos no servidor. Com base nestes dados o algoritmo a implementar no servidor deve decidir quais são as estradas que recebem o sinal verde e quais são as que recebem o sinal vermelho. O algoritmo deve procurar melhorar a fluidez do tráfego sem, no entanto, deixar os veículos demasiado tempo parados no semáforo.

Nesta primeira fase, é também necessário preparar o ambiente de teste e desenvolvimento, instalando as ferramentas necessárias e criando a topologia. Pode-se começar com uma topologia simples como a ilustrada na

Figura 1.

Em síntese, sugerem-se as seguintes atividades (a refinar pelos grupos de trabalho):

- criação da topologia híbrida de rede veicular ad-hoc e rede fixa de infraestrutura complementar
- definição dos dados que o veículo deve enviar, modo de os obter e formato de envio
- criação de uma aplicação para os veículos que implemente uma estratégia de difusão dos dados a um salto de distância
- criação de uma aplicação para receção dos dados na RSU, que se limita a recebe-los e enviá-los para o servidor
- criação da primeira versão do sistema de semáforos inteligentes que, com base no número de veículos detetados nas várias estradas que vão ter ao cruzamento controlado pelo semáforo, determina qual das estradas recebe o sinal verde e por quanto tempo. Os veículos são detetados através das mensagens recebidas no RSU.

Etapla 2 – Estabelecimento de encaminhamento MultiHop nos veículos com o objetivo de aumentar a perceção do semáforo da área circundante

Na segunda etapa, o objetivo é melhorar a perceção que o semáforo físico tem do ambiente à volta. A estratégia para isso é implementar um algoritmo de encaminhamento multihop que deverá ser executado pelos veículos. Um veículo quando recebe uma mensagem de outro veículo, deverá, em determinadas condições, reencaminhar essa mensagem. Dessa forma, consegue-se que o semáforo detete os veículos antes deles chegarem ao seu raio de alcance. O protocolo de encaminhamento a desenvolver deverá ser um protocolo “inteligente”, por exemplo, um veículo só deve reencaminhar pacotes de veículos que se estejam a aproximar do semáforo e não os daqueles que se estão a afastar. Adicionalmente devem ser implementados mecanismos que evitem sobrecarregar demasiado a rede.

Em síntese, sugerem-se as seguintes atividades (a refinar pelos grupos de trabalho):

- especificar e implementar um protocolo de encaminhamento multihop que permita uma melhoria da percepção do semáforo da área circundante
- implementar e avaliar mecanismos que evitem broadcast storms que resultem da execução do protocolo de encaminhamento nos veículos.

Nesta etapa pode ser necessário aumentar a topologia e/ou mexer no modelo de mobilidade dos veículos para testar convenientemente a solução desenvolvida.

Etapa 3 – Substituição do semáforo físico por um semáforo virtual

Na terceira etapa pretende-se evoluir o sistema de semáforos para que deixe ser necessário depender da existência de semáforos físicos. Assim, além da comunicação entre os veículos e o RSU, que já foi necessário implementar na primeira fase, nesta fase será necessário implementar a comunicação entre o RSU e os veículos. Dependendo do funcionamento do algoritmo que estabelece o verde ou vermelho em cada uma das vias, o RSU deverá enviar mensagens com ordens de avançar ou parar aos veículos respetivos.

Etapa 4 (Opcional) – Melhorias a introduzir ao sistema

Depois de concluídas as três etapas anteriores, cada grupo de trabalho deverá ter chegado a um protótipo de um sistema inteligente de semáforos virtuais, perfeitamente funcional. No entanto, como em qualquer sistema, será sempre possível melhorá-lo. Nesta quarta etapa, pretende-se que cada grupo implemente, à sua escolha, uma melhoria ao sistema desenvolvido nas três fases anteriores.

Como sugestão apresentamos duas alternativas:

- Implementar o semáforo virtual de forma distribuída, ou seja, a decisão de passar ou parar num cruzamento em vez de ser tomada centralmente pelo servidor local que está junto do RSU, deverá ser tomada de forma distribuída pelos próprios veículos com base nas mensagens que recebem dos veículos à sua volta. Desta forma o funcionamento do semáforo virtual deixa de estar dependente da infraestrutura, dependendo apenas dos veículos.

- Melhorar o funcionamento do semáforo virtual através da troca de mensagens entre vários semáforos contíguos (ou seja, semáforos que pertencem à mesma zona). O funcionamento cooperativo dos vários semáforos irá permitir uma melhoria na fluidez do tráfego rodoviário superior à que é conseguida quando a decisão é tomada por cada semáforo individualmente.

Ambiente de desenvolvimento e teste

A implementação e teste do sistema de semáforos inteligentes deverá ser efetuada na plataforma de simulação Eclipse MOSAIC - A Multi-Domain and Multi-Scale Simulation Framework for Connected and Automated Mobility (<https://eclipse.dev/mosaic/>). O Eclipse MOSAIC, é uma plataforma open-source de simulação usada para o desenvolvimento e avaliação de sistemas de mobilidade conectados e automáticos. O Eclipse Mosaic permite acoplar simuladores de vários domínios, como tráfego, comunicações, ambiente e aplicações, para criar um cenário de simulação abrangentes. O Eclipse Mosaic usa uma infraestrutura de tempo real (RTI) para gerir a troca de dados e o controlo do tempo entre os simuladores e pode ser usado

para modelar e avaliar diferentes cenários de mobilidade inteligente e conectada, como o uso de veículos elétricos, sistemas cooperativos de transporte inteligente, gestão de tráfego urbano, entre outros

Avaliação

O trabalho deve ser realizado em grupo, com grupos de três elementos, sendo a constituição dos grupos da responsabilidade dos alunos. O trabalho deverá ser demonstrado em duas fases (*demonstração intermédia* na semana antes da Páscoa, e *demonstração final* na semana de 27 de maio a 1 de junho). Na demonstração final, além da demo, os alunos deverão elaborar um relatório escrito que descreva o trabalho efetuado. Este relatório e o respetivo código deverão ser submetidos até ao dia 24 de maio na plataforma *elearning.uminho.pt*.

O relatório deve ser escrito em formato de artigo com um máximo de 8 páginas (recomenda-se o uso do formato LNCS - *Lecture Notes in Computer Science*). Deve descrever o essencial do desenho e implementação com a seguinte estrutura recomendada: Introdução; Especificação do sistema e dos protocolos de suporte (primitivas de comunicação, formato das mensagens protocolares (PDU), interações); Implementação (detalhes, parâmetros, bibliotecas de funções, etc); Testes e resultados; Conclusões e trabalho futuro.

Referências

- P. Alvarez Lopez *et al.*, “Microscopic Traffic Simulation using SUMO” in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Maui, USA, Nov. 2018, pp. 2575–2582. Accessed: Feb. 21, 2023. [Online]. Available: <https://www.itsc2019.org/>
- F. FOKUS. Eclipse MOSAIC – A Multi-Domain and Multi-Scale Simulation Framework for Connected and Automated Mobility. [Online]. Available: <https://www.eclipse.dev/mosaic/>
- R. Zhang et al. “Virtual Traffic Lights: System Design and Implementation”, in 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall). Chicago, IL, USA: IEEE, 2018-08, pp. 1–5. isbn: 978-1- 5386-6358-5. doi: 10.1109/VTCFall.2018.8690709.
- L. Jácome et al. “A Survey on Intelligent Traffic Lights”, in 2018 IEEE International Conference on Automation/XXIII Congress of the Chilean Association of Automatic Control (ICA-ACCA). 2018-10, pp. 1–6. doi: 10.1109/ICA-ACCA.2018.8609705.
- A. Bazzi et al. “A distributed algorithm for virtual traffic lights with IEEE 802.11p”, in 2014 European Conference on Networks and Communications (ECNC). Bologna, Italy: IEEE, 2014-06, pp. 1– 5. isbn: 978-1-4799-5280-9. doi: 10.1109/EuCNC.2014.6882621.
- H. Qin e H. Zhang. “Intelligent traffic light under fog computing platform in data control of real-time traffic flow”, in *The Journal of Supercomputing* 77.5 (2021-05), pp. 4461–4483. issn: 0920- 8542, 1573-0484. doi: 10.1007/s11227-020-03443-3.