



**Encontrando estacionamentos de carros:
uma aplicação de névoa computacional**

Douglas A. S. Costa

Supervisor: Edmundo Madeira

Trabalho de Conclusão de Curso

Engenharia de Computação

Instituto de Computação da Universidade Estadual de Campinas

Abstract. *Basing in a model of sensors and servers in the parking lots of establishments (shopping malls, restaurant, banks, etc), was designed a Cloud and Fog infrastructure that, along with the vehicular networks, are a fast and low latency communication network that allows to the common person a fast and reliable way to find a parking spot, saving time, fuel and generating less traffic. For this problem, I designed a simplified simulation of a vehicle that changes its routes and an application that show available parking spots near its destination. The simulation was made using SUMO (Simulation of Urban Mobility) by its Java interface TraCI4J.*

Resumo. *Baseando-se num modelo de sensores e servidores presentes nos estacionamento de estabelecimentos (shoppings, restaurantes, bancos, etc), foi desenvolvido uma estrutura de Nuvem e Névoa que, junto com as redes veiculares, formam uma rede de conexão de baixa latência que dá o usuário uma maneira rápida e confiável de achar uma vaga de estacionamento para seu veículo, possibilitando uma maior economia de tempo, combustível e diminuição do tráfego. Para este problema, montei uma simulação simplificada de um veículo que muda suas rotas e uma aplicação que mostra vagas estacionamento disponíveis próximo de seu destino. A simulação foi feita usando o SUMO (Simulation of Urban Mobility) através de sua interface para Java, TraCI4J.*

1. Introdução

Se te perguntarem neste momento aonde está seu carro, provavelmente a resposta será: "está estacionado". Segundo o *International Parking Institute* (IPI), veículos privados passam 95% do tempo parados numa vaga enquanto seus donos trabalham, estudam, comem, dormem, fazem compras ou alguma outra atividade rotineira. Estamos sempre nos movendo, somos afetados diretamente pela localização deixaremos nossos veículos nos destinos.

A cidades estão crescendo, estima-se que em 2030 60% da população mundial viverá em cidades[Institute 2014]. No terceiro trimestre de 2013 foi estimado que 247,9 milhões de carros caminhões circularam nas ruas, e é esperado que esse numero atinja 284 milhões próximo do ano de 2025. Um estudo realizado numa área de 15 quarteirões na área de Los Angeles, em um ano foram medidas 950 mil milhas rodadas de veículos a procura de vagas. Para se ter noção da quantidade, isso é equivalente a 38 voltas no Planeta Terra.

Não podemos deixar de citar que além dos problemas do trânsito, isso tem um impacto imenso na quantidade de emissão de gases de efeito estufa. Várias cidades e grandes centros já estão se movimentando e considerando vagas para estacionar uma grande preocupação em seus planejamentos.

Com isso em mente, foi pensado um modelo de aplicação que roda na nuvem e se comunica com os estabelecimentos na periferia da rede (Fog) para encontrar vagas para o motorista [Oanh Tran Thi Kim 2015] e feito uma simulação simplificada da aplicação usando a ferramenta SUMO [of Transportation Systems 2015]. Será desenvolvido o modelo proposto, metodologia e ferramentas usadas nas próximas seções.

2. Conceitos

Aqui usamos alguns conceitos de IoT e VANETs, Cloud e Fog. *Internet of Things* descreve uma visão da rede mundial de computadores como uma rede central acessada por todo tipo de *smart objects*, ou seja, qualquer dispositivo "esperto" o suficiente para se conectar a internet e trocar informações com outros dispositivos. Em nosso modelo temos variados deles, desde veículos e servidores até sensores de vagas, que conectados por Bluetooth ou WiFi informarão se temos ou não vagas disponíveis.

Restringindo para o caso dos veículos, podemos falar de VANET (Vehicular ad hoc network) que é uma rede distribuída de intercomunicação entre os veículos[Corser 2013]. Além de entre si, comunicam-se também com as RSUs (Roadside Units), unidades a beira da estrada que concentram informações locais, têm acesso a internet e se comunicam com a Roadside Cloud, responsável pelo maior processamento de informações. Podem entrar nas VANETs outros dispositivos, como semáforos, praças de pedágios e centrais de trânsito.

As VANETs foram pensadas para diminuir a latência da obtenção de informações: ao invés de todos se concentrarem numa Cloud central, as informações são obtidas localmente com requisições de baixas latências resultando em informações e tomadas de decisões mais rápidas. As comunicações entre veículos são usadas para traçar rotas de auto-guiados e para se evitar colisões. No caso de um veículo colidir no meio da via, este consegue imediatamente avisar os mais próximos da via a tempo de evitar acidentes, o que talvez não fosse possível se tivesse de aguardar a nuvem avisar os outros veículos.

As RSUs são para informações locais, no exemplo acima o veículo parado também informaria a RSU mais próxima (ou outro veículo informado) que passaria a informação adiante, fazendo com que fosse possível avisar para outros veículos antecipadamente sobre um possível engarrafamento e sugerir rotas alternativas. Estes dados também seriam sincronizados com a Cloud para que esta pudesse acionar as autoridades competentes (ambulância, bombeiros, polícia) e mais tarde sejam analisados mais de uma maneira mais específica pelo departamento de trânsito ou pela concessionária da via, como por exemplo detectar áreas de maior perigo e decidir onde colocar semáforos, rotatórias ou diminuir a velocidade máxima permitida no trecho.

Tudo isso nos leva ao conceito de Fog (Névoa) que deriva seu comportamento da Cloud (Nuvem) e o realiza de forma mais distribuída nas bordas da rede. O termo foi criado pela Cisco Com isso podemos ter um menor tempo de processamento pois desafoga a Cloud, menor tempo de resposta por ter uma menor latência nas requisições, e ainda a maior confiabilidade nos dados pelo fato de serem avaliadas informações mais específicas sobre o local.

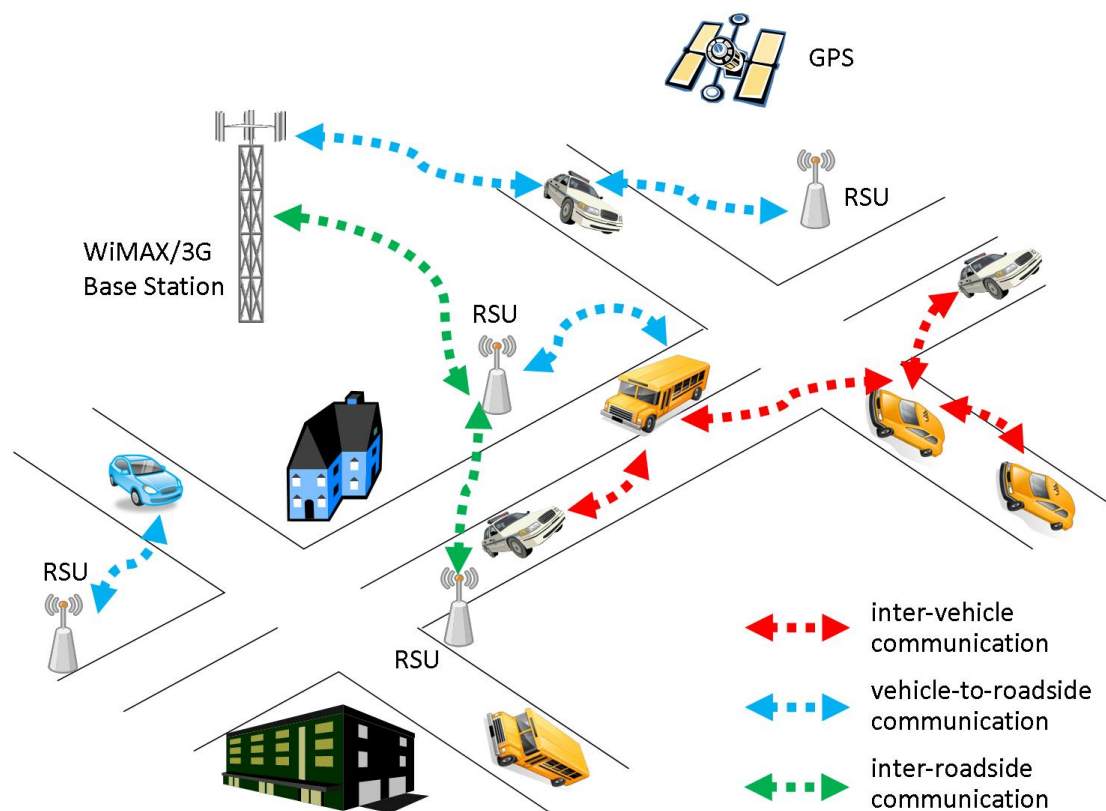


Figure 1. Comunicações numa VANET.

3. Modelo

O modelo proposto em [Oanh Tran Thi Kim 2015] foi pensado para vagas de estacionamento compartilhadas entre vários tipos de estabelecimentos. Sendo estes próximos mas de seguimentos variados poderiam compartilhar vagas de estacionamento, visto que operariam com maior público em horários diferentes. Por exemplo uma casa noturna ter vagas compartilhadas com um banco ou salão de beleza. O salão e o banco têm públicos diurnos, enquanto a outra possui maior movimento a noite. Na prática isto já existe, porém são apenas acordos e convênios fechados entre no máximo dois estabelecimentos.

O modelo engloba duas aplicações diferentes: uma rodando na nuvem que recebe as solicitações de vagas dos usuários e outra rodando na Fog que gerencia as vagas em um ou mais estabelecimentos próximos.

Para cada estabelecimento teríamos um servidor de baixo custo responsável pela gerência das vagas dos mesmos. A ideia é que sensores sejam usados nas vagas e se comuniquem com este servidor por tecnologias sem fio. Estes servidores estariam conectados a internet e previamente cadastrados num banco de dados acessado pela aplicação na nuvem e seriam requisitados quando fosse necessário encontrar vagas de estacionamento próximo àquele local. Os estabelecimentos próximos poderiam até compartilhar a mesma máquina física de servidor, podendo conter várias VMs com uma instância cada-com instância escutando em portas diferentes.

Os sensores seriam qualquer um deles que tenha capacidade de discernir se a vaga está ocupada ou não. Podem ser utilizados sensores simples de proximidade com

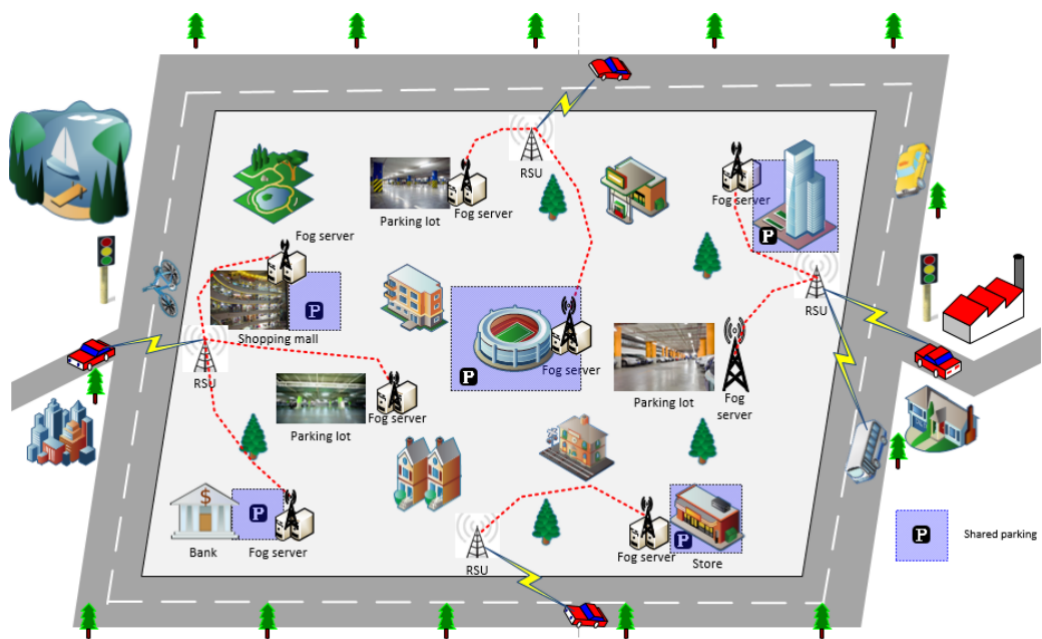


Figure 2. Esquema representando as comunicações entre servidores Fog, RSUs e a Cloud[Oanh Tran Thi Kim 2015].

comunicação Bluetooth com o servidor até sistemas de análise de imagem usando câmeras de vigilância.

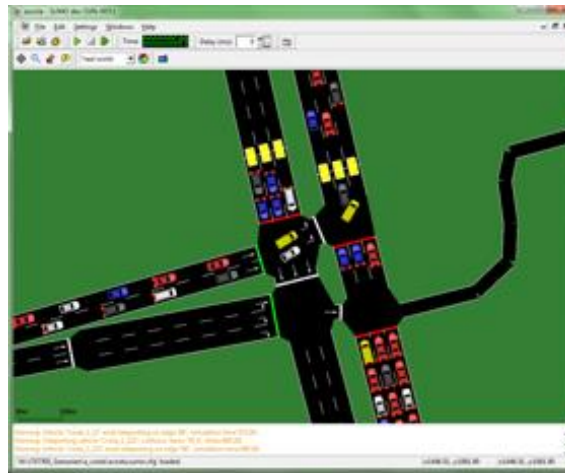


Figure 3. Interface gráfica SUMO.

4. Ferramentas

4.1. SUMO

A simulação de tráfego facilita a avaliação de modelos propostos antes de sua implementação, por exemplo a adição de um semáforo ou cruzamento na via pode ser testado e otimizado antes de ser aplicado no mundo real. O SUMO (Simulation of Urban Mobility) [of Transportation Systems 2015] é uma ferramenta open source criada em 2001 que possibilita a modelagem e testes do modelo como um todo, incluindo veículos, pedestres e transporte público.

5. Images

All images and illustrations should be in black-and-white, or gray tones, excepting for the papers that will be electronically available (on CD-ROMs, internet, etc.). The image resolution on paper should be about 600 dpi for black-and-white images, and 150-300 dpi for grayscale images. Do not include images with excessive resolution, as they may take hours to print, without any visible difference in the result.

6. References

Bibliographic references must be unambiguous and uniform. We recommend giving the author names references in brackets, e.g. [Knuth 1984], [Boulic and Renault 1991], and [Smith and Jones 1999].

The references must be listed using 12 point font size, with 6 points of space before each reference. The first line of each reference should not be indented, while the subsequent should be indented by 0.5 cm.

References

Boulic, R. and Renault, O. (1991). 3d hierarchies for animation. In Magnenat-Thalmann, N. and Thalmann, D., editors, *New Trends in Animation and Visualization*. John Wiley & Sons Ltd.

- Corser, G. (2013). Vanet introduction. <https://www.youtube.com/watch?v=DrH-1505-Mg>. [Online; accessed 09-November-2015].
- Institute, I. P. (2014). Why parking matters. *The Role of Parking in Our Future Cities*.
- Knuth, D. E. (1984). *The T_EX Book*. Addison-Wesley, 15th edition.
- Oanh Tran Thi Kim, Nguyen Dang Tri, V. N. N. H. T. C. S. H. (2015). A shared parking model in vehicular network using fog and cloud environment. *Network Operations and Management Symposium (APNOMS), 2015 17th Asia-Pacific*, pages 321–326.
- of Transportation Systems, I. (2015). Simulation of urban mobility. http://sumo.dlr.de/wiki/Main_Page. [Online; accessed 09-November-2015].
- Smith, A. and Jones, B. (1999). On the complexity of computing. In Smith-Jones, A. B., editor, *Advances in Computer Science*, pages 555–566. Publishing Press.