

Mecanismo de Negociação Descentralizado entre Agentes em um Estacionamento Inteligente

Felipe Felix Ducheiko¹, Gleifer Vaz Alves¹, André Pinz Borges¹

¹Departamento Acadêmico de Informática
Universidade Tecnológica Federal do Paraná (UTFPR)
Rua Doutor Washington Subtil Chueire, 330 - Jardim Carvalho - Ponta Grossa - PR
felipeducheiko@alunos.utfpr.edu.br, {gleifer, apborges}@utfpr.edu.br

Abstract. *In a Smart Parking, agents could negotiate in order to obtain a parking spot. Usually, this sort of negotiation is centralized, using a manager agent. Here, we present a decentralized approach, where a given agent (seller) can offer a parking spot for n agents (buyers). A negotiation mechanism is presented together with some results which show how the Multi-Agent system works considering the number of exchanged messages (between agents), the time to park (when a negotiation is done), and the selling price (of parking spaces).*

Resumo. *Em um Smart Parking agentes podem negociar para obter uma vaga de estacionamento. Normalmente, esse tipo de negociação é centralizada, usando um agente de centralizador. No presente trabalho apresentamos uma abordagem descentralizada, em que um determinado agente (seller) pode oferecer uma vaga de estacionamento para n agentes (buyer). Com isso, é descrito um mecanismo de negociação em conjunto com alguns resultados experimentais, os quais evidenciam o funcionamento do Sistema Multi-Agente, em termos de quantidade de mensagens (entre os agentes), do tempo para estacionar (quando uma negociação é bem sucedida) e o valor de venda das vagas.*

Keywords: Smart Parking, Multi-Agent Systems, Negotiation Protocol.

Palavras-chave: Estacionamento Inteligente, Sistemas Multi-Agente, Protocolo de Negociação.

1. Introdução

O conceito *Smart City* surgiu durante a última década com a fusão de várias ideias, onde a essência do conceito é integrar as tecnologias que até agora têm sido desenvolvidas separadamente, mas que tem ligações claras em seu funcionamento e podem ser desenvolvidas de forma integrada [Batty et al. 2012]. Dentre os desafios a serem enfrentados pelas cidades destacam-se os de mobilidade urbana. Em Nova York 40% dos congestionamentos decorrem de motoristas buscando vagas de estacionamento [Koster et al. 2014].

Quando se percebe que a demanda de vagas de estacionamentos não está sendo satisfeita a solução adotada normalmente é um aumento quantitativo do número de vagas. Porém, a utilização das mesmas vagas de modo mais inteligente pode amenizar ou

até solucionar o problema. *Smart Parkings* são sistemas compostos por dispositivos de *hardware*, capazes de detectar o nível de ocupação do estacionamento e *softwares* integrados, para gerir a atribuição desses espaços de estacionamento [Nocera et al. 2014]. Tais sistemas são concebidos para auxiliar os motoristas na localização de vagas disponíveis, colaborando com a solução de problemas relacionados à mobilidade urbana. Dentre os vários modelos computacionais que podem ser utilizados para implementar um *Smart Parking* destacam-se os Sistemas Multiagentes (SMA).

Segundo [Wooldridge 2009] SMA são sistemas compostos de elementos computacionais que realizam interações ente si, de modo a atingirem seus objetivos, sendo tais elementos conhecidos como agentes. Esses agentes possuem duas características: são capazes de ações autônomas e têm a capacidade de interagir uns com os outros pela interação análoga às interações sociais humanas [Wooldridge 2009].

Por meio das habilidades sociais agentes devem ser capazes de negociar uns com os outros, afim de solucionarem problemas de forma distribuída, como ocorre em sociedades. Negociação é um processo complexo de tomada de decisão em que cada parte representa de forma autônoma seus pontos de vista e interage com as outras para resolver conflitos e chegar a acordos, maximizando os ganhos de todas as partes [Choi et al. 2001].

Para desenvolver um mecanismo de negociação é possível utilizar três configurações: i) *one-to-one*: onde um agente negocia somente com um agente, neste caso os agentes possuem preferências simétricas; ii) *many-to-one* (centralizado): nesta configuração, um único agente negocia com vários agentes, como acontece em um leilão; e iii) *many-to-many* (descentralizado): neste caso, muitos agentes negociam com muitos agentes simultaneamente, como acontece em um mercado [Wooldridge 2009].

Com o objetivo de aplicar métodos e técnicas usados em Sistema Multiagente na criação de soluções para alocação de vagas e gerenciamento de um *Smart Parking* foi concebido um projeto, que possui um SMA com mecanismo de negociação centralizado [Castro et al. 2017]. No sistema todos os agentes *drivers* negociam exclusivamente com o agente administrador. Esta abordagem possui vantagens, como o fato dos agentes *drivers* trocarem mensagem apenas com o agente administrador, diminuindo o custo computacional. Mas também possui desvantagens, visto que o agente administrador pode falhar, deste modo comprometendo todo o SMA. Outro ponto negativo é a questão da autonomia dos agentes, visto que os agentes *drivers* em um sistema com mecanismo de negociação centralizado ficam sujeitos as imposições do agente administrador.

Para o desenvolvimento de agentes autônomos com capacidades sofisticadas e flexíveis de negociação três itens devem ser definidos: i) protocolo de negociação, o qual define as ações que os agentes podem tomar em uma negociação; ii) o problema que a negociação quer solucionar; e iii) modelo de raciocínio, o qual define as ofertas iniciais, a gama de ofertas aceitáveis, as contraofertas, quando a negociação deve ser abandonada e quando um acordo deve ser fechado [O'Hare and Jennings 1996].

O objetivo do presente trabalho é estender o projeto, analisando o SMA para alocação de vagas de estacionamento com um mecanismo de negociação descentralizado, onde não existe a figura de um agente centralizador que administra o sistema e os agentes

drivers negociam as vagas entre si. Este artigo apresenta: (i) protocolo de negociação e modelo de raciocínio descentralizados; (ii) brevemente a implementação do SMA descentralizado; e (iii) uma parte dos experimentos realizados para validar o sistema.

2. Modelo de Raciocínio e Protocolo de Negociação

O modelo de raciocínio assume a existência de apenas um tipo de agente no SMA, o agente *driver*, o qual pode assumir dois papéis distintos: *seller*, quando o agente está deixando uma vaga e *buyer*, quando o agente está procurando uma vaga. Este modelo de raciocínio é apresentado em maiores detalhes em [Ducheiko et al. 2018].

O modelo de raciocínio obedece a um protocolo de negociação, que é o conjunto de regras as quais especificam a gama de movimentos legais disponíveis para cada agente em qualquer fase de um processo de negociação [Endriss et al. 2006]. A Figura 1 apresenta o protocolo de negociação que é utilizado para implementar o SMA.

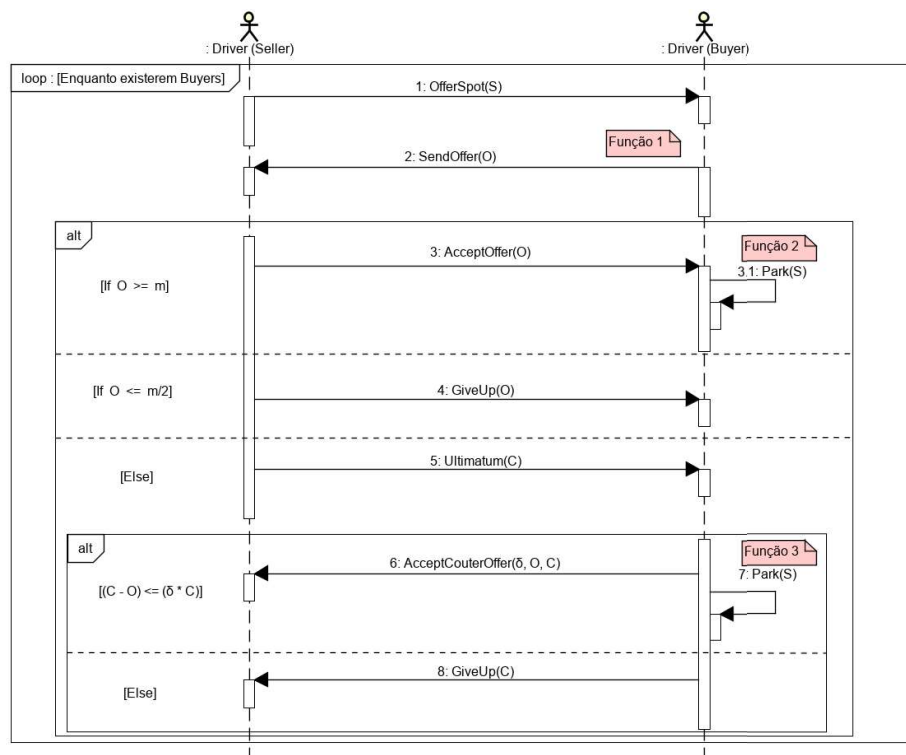


Figura 1. Protocolo de Negociação

Este protocolo define que sempre a negociação deverá ser iniciada pelo agente *driver seller*, informando os outros *drivers* do sistema que possui uma vaga para negociar (na Figura 1, item 1: *OfferSpot(S)*). Então, os agentes *drivers* que estão procurando uma vaga (*buyer*) geram uma proposta, utilizando a Função 1, e a enviam para o *driver seller* (na Figura 1, item 2: *SendOffer(O)*). Então, o *driver seller* analisa estas propostas utilizando a Função 2, podendo tomar as seguintes decisões: (i) aceitar proposta (na Figura 1, item 3: *AcceptOffer(O)*), onde a negociação acaba e o acordo é fechado; (ii) rejeitar proposta (na Figura 1, item 4: *GiveUp(O)*), onde a negociação entre estes dois agentes acaba; ou

(iii) enviar contraproposta (na Figura 1, item 5: *Ultmatum(C)*), onde o agente *seller* gera uma contraproposta utilizando ainda a Função 2 e envia para o agente *buyer*.

Quando o agente *buyer* recebe esta contraposta ele a analisa utilizando a Função 3 e pode tomar duas ações: (i) aceitar a contraproposta (na Figura 1, item 6: *AcceptCounterOffer(δ , O, C)*), onde um acordo entre os dois *drivers* é fechado e a negociação acaba; ou (ii) rejeitar contraproposta (na Figura 1, item 8: *GiveUp(C)*), onde a negociação entre estes dois agentes acaba e o agente *seller* inicia uma negociação com outro *driver buyer*, isto se repete até que não haja mais agentes para negociar.

3. Implementação

Para implementar o sistema foram utilizadas as três camadas do *framework* JaCaMo. Na camada do *Jason* foram definidos dois tipos de agentes, os agentes *drivers* que utilizam o sistema e negociam vagas entre si e os agentes *parkingSpotController*, que possuem um artefato *Spot* relacionado e controlam o mesmo. Na camada do *Cartago*, foram definidos dois artefatos: o artefato *ParkingSpot*, que é a vaga a qual o agente *parkingSpotController* controla e o artefato *ProposalGenerator*, que é um artefato gerador de proposta, o qual os agentes *drivers* utilizam para calcular as propostas da negociação. Na camada do *Moise*, foi feita a especificação organizacional do sistema, onde foram definidos os dois papéis que o agente *driver* pode assumir, *seller* e *buyer*.

A Figura 2 mostra dois trechos do código da especificação da organização do SMA, utilizando o *Moise*. No primeiro trecho existe a definição dos papéis (definido em *role-definitions*), onde é possível observarmos a especificação dos dois papéis já mencionados: *seller* e *buyer*. No segundo trecho é possível visualizar as normas do sistema.

```

16<role-definitions>
17  <role id="driver" />
18<role id="buyer">
19  <extends role="driver"/>
20  </role>
21<role id="seller">
22  <extends role="driver"/>
23  </role>
24 </role-definitions>
96<normative-specification>
97  <norm id="norm1" type="obligation" role="buyer" mission="arrival"/>
98
99  <norm id="norm2" type="permission" role="driver" mission="negotiate"/>
100
101  <norm id="norm3" type="permission" role="seller" mission="departure"/>
102 </normative-specification>

```

Figura 2. Trechos código *Moise*

A primeira norma é a *norm1* que define que o agente *buyer* execute a missão *arrival*, que consiste em o agente chegar no estacionamento e verificar se há vagas livres que obedeçam os requisitos solicitados. A segunda norma é a *norm2* que define que os agentes iniciem a negociação das vagas. A terceira norma é a *norm3* que define que o agente *buyer* execute a missão *departure* e passe a ser um *seller*. Portanto, os agentes *buyer* que entram no sistema, procurando uma vaga e após conseguirem esta vaga trocam o seu papel e passam a ser *seller*, permanecendo na vaga pelo tempo desejado e depois tentando vender a vaga para outro motorista.

É importante ressaltar que no início da execução do SMA não existem agentes *sellers* e as vagas estão livres, isto é, não pertencem a nenhum agente. Nesta etapa a alocação de vagas é feita sem utilização de negociação. O agente *buyer* envia uma mensagem para os agentes *parkingSpotController* verificando se a vaga que ele é responsável está livre e o agente *buyer* estaciona na primeira vaga que receber a resposta que está livre que obedeça os requisitos solicitados, esta vaga passa a ser sua e o motorista assume o papel de *seller*.

4. Resultados

Nesta seção são apresentados os resultados obtidos por meio da execução de cenários de testes, com o objetivo de validar o funcionamento do SMA e verificar como o mecanismo de negociação se comporta com a alteração do número de *drivers*. Para tanto, foram executados 10 cenários de testes, todos com 160 vagas de estacionamento. O número de *drivers* em cada cenário variou de 50 a 500, sendo acrescido de 50 *drivers* a cada cenário.

O Gráfico 1 (Figura 3) apresenta a variação do número médio de mensagens enviadas e o tempo médio para estacionar por *driver* em cada cenário de teste. No Gráfico 1 os valores médios do tempo para estacionar são multiplicados em 10 vezes, para que a escala fique mais próxima dos valores médios de mensagens enviadas e assim fazendo com que a análise fique mais clara.



Figura 3. Gráficos de Análise

A partir da análise dos dados coletados da execução dos cenários de testes, foi possível constatar que o tempo para alocação da vaga de estacionamento tende a crescer conforme o número de *drivers* no sistema, como é possível verificar no Gráfico 1. Este fenômeno acontece devido ao fato de que, quanto maior o número de agentes no sistema mais mensagens serão enviadas para realizar a negociação das vagas. A troca de mensagens influi no tempo de alocação das vagas, pois quanto maior o número de mensagens trocadas para fechar uma negociação, mais demorada é esta negociação.

O Gráfico 2 (Figura 3) apresenta a variação da distância média entre o ponto onde o agente deseja a vaga e o ponto onde o agente efetivamente estacionou, juntamente com valor médio de venda das vagas em cada cenário. No Gráfico 2, os valores médios de venda das vagas são multiplicados em 12 vezes, para que a escala fique mais próxima dos valores de distância média e assim fazendo com que a análise fique mais clara.

Por meio da análise do Gráfico 2, foi possível constatar que, a distância média entre o ponto onde o agente deseja a vaga e o ponto onde o agente efetivamente estacionou diminui conforme o número de *drivers* no sistema aumenta. Conseqüentemente, o valor médio de venda das vagas tende a crescer conforme o número de *drivers* cresce, devido ao fato de que, quanto menor a distância entre estes pontos maior será o valor que um *driver* estará disposto a pagar. Este fenômeno acontece pois quanto maior o número de *drivers* no sistema, mais negociações irão acontecer. Portanto, os *drivers buyers* terão mais opções de vagas para negociar e como eles tendem a escolher vagas mais próximas ao ponto onde desejam estacionar, a distância média entre o ponto de desejo e o ponto onde estacionou diminui. Esse comportamento é o esperado conforme fórmulas do modelo de raciocínio (maiores informações podem ser consultadas em [Ducheiko et al. 2018]).

5. Conclusão

O desenvolvimento de projetos relacionados com o tema de Cidades Inteligentes é importante, melhorando a qualidade de vida das pessoas e contribuindo no desenvolvimento sustentável e inteligente das cidades. Estacionamentos Inteligentes contribuem neste aspecto, buscando alternativas para automatizar o gerenciamento de estacionamentos.

Com o propósito de validar o funcionamento do SMA e aperfeiçoar o modelo de raciocínio e protocolo de negociação, foram desenvolvidos testes com o Sistema Multiagente. Conforme apresentado, foram feitos experimentos para analisar o funcionamento do sistema, aqui destacam-se as seguintes características: i) existe uma curva de tendência de crescimento de mensagens trocadas entre os agentes conforme aumenta o número de agentes no sistema, ocasionando um maior tempo para a alocação das vagas de estacionamento; e ii) conforme o número de agentes cresce existe uma tendência de que os agentes estacionem em uma vaga mais próxima do ponto de desejo. Como trabalhos futuros podemos elencar: (i) realizar comparações com abordagens centralizadas; (ii) rodar experimentos utilizando dados reais; e (iii) desenvolver outros modelos de raciocínio.

Referências

- Batty, M., Axhausen, K., Fosca, G., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., Ouzounis, G., and Portugali, Y. (2012). Smart cities of the future.
- Castro, L. F. S. D., Alves, G. V., and Borges, A. P. (2017). Using trust degree for agents in order to assign spots in a Smart Parking. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, 6(2):45–55.
- Choi, S., Liu, J., and Chan, S.-P. (2001). A genetic agent-based negotiation system. *Comput. Netw.*, 37(2):195–204.
- Ducheiko, F. F., André, P. B., and Gleifer, V. A. (2018). Implementação de Modelo de Raciocínio e Protocolo de Negociação para um Estacionamento Inteligente com Mecanismo de Negociação Descentralizado. *Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia*, 1(19):25–32.
- Endriss, U., Maudet, N., Sadri, F., and Toni, F. (2006). Journal of artificial intelligence research. *Negotiating Socially Optimal Allocations of Resources*, 25:315–348.
- Koster, A., Koch, F., and Bazzan, A. (2014). Incentivising crowdsourced parking solutions.
- Nocera, D. D., Napoli, C. D., and Rossi, S. (2014). A social-aware smart parking application. *eur workshop proceedings*. 1260.
- O’Hare, G. M. P. and Jennings, N. R., editors (1996). *Foundations of Distributed Artificial Intelligence*. John Wiley Amp; Sons, Inc., New York, NY, USA.
- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. J. Wiley, New York, 2nd edition.