

Transporte de Agentes Cognitivos entre SMA Distintos Inspirado nos Princípios de Relações Ecológicas

Vinicius S. de Jesus¹, Fabian Cesar P. B. Manoel¹, Carlos Eduardo Pantoja^{1,2}, Jose Viterbo²

¹Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET-RJ)
Campus Maracanã – 20271-110 – Rio de Janeiro – RJ – Brasil

²Universidade Federal Fluminense (UFF)
Campus Praia Vermelha – 24210-346 – Niterói – RJ – Brasil

{souza.vdj,fabiancpbm,viterbo}@gmail.com, pantoja@cefet-rj.br

Abstract. *In biology, living beings are able to establish relationships that can be classified according to the behavior of those involved. Agents are autonomous entities capable of making decisions, cognitive reasoning and socializing with other agents in a Multi-Agent System (MAS). Some agents are able to move to other MAS, thus being able to relate to agents, in a similar way to living beings. This work aims to propose protocols inspired by biological relations with the purpose of exploring the movement of agents to a MAS embedded in a physical and autonomous device to another MAS in a different device. Three protocols will be addressed: predatism, mutualism and inquiline, where the transference is made with the goal of dominating, exchanging knowledge and surviving in the destination MAS, respectively. These protocols aim to preserve and/or share the indispensable knowledge obtained during the existence of the agents. In this case, an MAS can use one of the proposed protocols to migrate to another embedded system. Finally, some initial experiments will be presented, in which two prototypes (one leader and one host) were created where the leader is damaged and the predatism relationship is activated to preserve the acquired knowledge.*

Resumo. *Na biologia, os seres vivos são capazes de estabelecer relações que são classificadas de acordo com o comportamento dos envolvidos. Agentes são entidades autônomas capazes de tomar decisões, raciocínio cognitivo e socializar com outros agentes em um Sistema Multi-Agente (SMA). Alguns agentes são capazes de se moverem para outros SMA, podendo, assim, se relacionar com outros agentes, de forma similar aos seres vivos. Este trabalho tem como objetivo propor protocolos inspirados nas relações biológicas com a finalidade de explorar a movimentação de agentes de um SMA embarcado em um dispositivo físico e autônomo para outro SMA em um dispositivo distinto. Serão abordados três protocolos: predatismo, mutualismo e inquilinismo, onde a transferência é feita com o objetivo de dominar, trocar conhecimentos e sobreviver no SMA de destino, respectivamente. Estes protocolos visam preservar e/ou compartilhar os conhecimentos obtidos durante a existência dos agentes. Neste caso, um SMA pode utilizar um dos protocolos para migrar para outro sistema embarcado. Por fim, serão apresentados alguns experimentos iniciais, nos quais foram criados dois protótipos (um líder e um hospedeiro) onde o líder é danificado e a relação de predatismo é acionada para preservar os conhecimentos adquiridos.*

1. Introdução

Para a ecologia, relações ecológicas é o estudo dedicado a explicar a forma que os seres vivos se relacionam a fim de manterem suas vidas [Tissot-Squalli 2009]. Existem subdivisões estabelecidas para as relações ecológicas tais como: mutualismo, inquilinismo e predatismo. O mutualismo é o tipo de relação ecológica onde os seres envolvidos possuem características que beneficiam os outros seres da relação, sendo considerada positiva para todos; o inquilinismo é a relação em que apenas um dos seres envolvidos se beneficia da relação, sem causar prejuízo a outro ser; já o predatismo é a relação em que um dos seres se beneficia, prejudicando o outro.

Os Sistemas Multi-Agentes (SMA) [Wooldridge 2009] são sistemas compostos de agentes autônomos, onde estes podem reagir a estímulos, tomar decisões e interagir com outros agentes. Os SMA são oriundos da sociologia e etologia [Hübner et al. 2004] e, por isso, se baseiam na ideia de que um indivíduo isolado não é tão forte quanto um conjunto de indivíduos. Dessa forma, pode-se considerar que a abordagem de agentes é um paradigma que estuda o comportamento de um grupo de agentes que busca a resolução de problemas coletivamente ou compete por um objetivo comum. Então, ao se falar de SMA, leva-se em consideração o relacionamento entre as suas partes e, assim como podem existir diferentes espécies se relacionando em um habitat, podem existir diferentes agentes em um SMA, inclusive, SMA diferentes se relacionando em um mesmo ambiente. Os SMA podem ser abertos ou fechados [Chebout et al. 2016], onde os sistemas abertos são capazes de se comunicarem com outros SMA, enquanto que, os fechados não. Os sistemas abertos também permitem a entrada e saída de agentes, estes agentes com essa capacidade de se moverem para diferentes SMA são chamados de agentes móveis.

O Ambient Intelligence (AmI) [Augusto Wrede et al. 2010] é um ambiente que faz uso de técnicas e sistemas computacionais alinhados a dispositivos físicos para interagir de forma ubíqua e imperceptível com os usuários visando auxiliá-los em tarefas específicas no dia-a-dia. A Internet of Things (IoT) [Zhang et al. 2012], é a interligação de diversos dispositivos autônomos e proativos em uma mesma rede e na internet com o objetivo de resolver problemas do dia-a-dia das pessoas. Para o desenvolvimento de aplicações para a IoT existe um *middleware* chamado de ContextNet [Endler et al. 2011] que proporciona a comunicação entre dispositivos ou nós móveis visando atividades com a geração de contexto e que suportam milhares de conexões simultâneas. Para o desenvolvimento de SMA existem diversos *frameworks*, como o Jason [Bordini et al. 2007] e Jade [Bellifemine et al. 2007], que possuem interpretadores de linguagens de programação orientadas a agentes que são utilizadas para criação de AmI integrados com a IoT.

Através de uma pesquisa na literatura de SMA abertos e agentes móveis foi possíveis observar que os principais trabalhos não focam na transferência de agentes entre diferentes SMA, mas sim, na segurança dos SMA, como nos trabalhos THOMAS [Ossowski et al. 2007] e em modelo de confiança entre agentes e SMA (FIRE) [Dong-Huynha et al. 2004], ou até mesmo, no consenso e adaptação dos agentes em um novo SMA, como, em análise de estratégias intra-equipe [Sanchez-Anguix et al. 2012] e estratégia de agrupamento de sistemas bio-inspirado em colônias de formigas [Calvo 2012]. Sendo assim, deixando uma lacuna na literatura que este trabalho pretende explorar.

Os trabalhos citados focam em soluções sobre o relacionamento entre os agentes de um SMA e agentes externos. Sendo assim, o objetivo deste trabalho é proporcionar o transporte de agentes de um SMA para outro baseado nos conceitos de relações ecológicas da biologia [Tissot-Squalli 2009]. Para isso, foram propostos 3 protocolos para a transferência de agentes entre SMA distintos, conhecidos como predatismo, inquilinismo e mutualismo, onde permitem dominar, sobreviver e trocar experiências com outros SMA, respectivamente. Será realizado um experimento com dois protótipos de carros que possuem sensores ultrassônicos, motores e LEDs, cada um com um SMA embarcado, onde um carro ativará o protocolo predador e, portanto, pregará o outro carro para dar continuidade as suas atividades.

O trabalho está estruturado da seguinte maneira: na seção 2, o referencial teórico será apresentado; na seção 3, o protocolo de transferência de agentes será apresentado; na seção 4, os testes executados serão expostos; na seção 5, os Resultados e discussões serão abordados; na seção 6, os trabalhos relacionados serão discutidos; na seção 7, a conclusão e trabalhos futuros serão exibidos; e, por fim, as referências bibliográficas serão apresentadas.

2. Referencial Teórico

Nesta seção, serão explorados os conceitos básicos para a compreensão do trabalho, tais como, o *framework* Jason, a arquitetura ARGO, assim como as tecnologias utilizadas para o desenvolvimento de tais ambientes e sistemas (*middleware* ContextNet).

2.1. Desenvolvimento de SMA

O Jason [Bordini et al. 2007] é um *framework* que interpreta uma linguagem orientada a agentes chamada AgentSpeak em Java para a programação de SMA e, também, implementa o modelo Belief-Desire-Intention (BDI) [Bratman 1987], permitindo assim, a programação de agentes reativos e cognitivos. O modelo BDI é composto basicamente por crenças, desejos e intenções: as crenças são informações tidas como verdades pelo agente; os desejos são as motivações que um agente possui para realizar algum objetivo e intenções são ações que o agente se compromete a realizar.

O ARGO [Pantoja et al. 2016b] é uma arquitetura customizada que permite programar agentes com a capacidade de interagir e controlar microcontroladores através do *framework* Jason. Os agentes com esta capacidade podem ser chamados também de agentes robóticos [Matarić 2007], pois, são eles que fazem a interação direta com o *hardware* fornecendo leitura de sensores, controle de acionamentos e interação com o meio físico. A arquitetura customizada ARGO é formada por dois principais módulos: o *middleware* Javino e os filtros de percepções. O Javino [Lazarin and Pantoja 2015] permite a comunicação entre linguagens de alto nível e controladores que possuem uma conexão serial, por exemplo, o Arduino; além disso, o Javino possui uma estrutura de identificação de erro que impede a perda de dados nas mensagens enviadas ou recebidas em tempo de execução. Os filtros de percepções [Stabile Jr. and Sichman 2015] são mecanismos que fazem com que o agente bloqueie as percepções do ambiente com base em filtros projetados pelo desenvolvedor em tempo de design, porém, os filtros podem ser alterados em tempo de execução pelo agente que possui o arquivo.

2.2. Tecnologias para a Internet das Coisas

O ContextNet [David et al. 2012] é um *middleware* criado para a IoT e serviços de contexto, que visa aplicações colaborativas, coordenação de atividades de entidades móveis e compartilhamento de informações. As entidades móveis podem ser qualquer dispositivo móvel, por exemplo, veículos, *smartphones*, *tablet*, *notebook* ou até mesmo robôs autônomos com conexão a redes IoT. Entre as tecnologias utilizadas para o desenvolvimento do ContextNet, o *Scalable Data Distribution Layer* (SDDL), [Vasconcelos et al. 2013], é uma camada do ContextNet responsável pela capacidade de comunicação e de distribuição de contexto. As demais funcionalidades do ContextNet foram desenvolvidas em módulos de *software* no topo da camada de distribuição. Além disso, o ContextNet permite a entrada de dispositivos em sua rede e o tratamento adequado.

O *middleware* ContextNet [Endler et al. 2011] permite o controle de milhares de nós móveis conectados ao mesmo tempo. Em [David et al. 2012] simula-se uma situação para o rastreamento em tempo real de milhares de cargas sendo transportadas com o posicionamento sendo atualizado. Por conta disso, o *middleware* ContextNet apresenta características que podem ser exploradas para a conexão entre diversos SMA embarcados e prover um ambiente para as relações biológicas propostas neste trabalho.

3. Protocolo de Transferência de Agentes

Nesta seção serão apresentadas a metodologia do protocolo de transferência de agentes, as possíveis relações estabelecidas entre SMA de origem e SMA de destino, além da sua implementação e tecnologias utilizadas. O protocolo de transferência de agentes permite a transferência de SMA inteiros ou agentes considerando a relação que será estabelecida com o SMA de destino, que são baseadas nos conceitos de relações ecológicas oriundas da biologia: Inquilinismo, Mutualismo e Predatismo. Na relação de Inquilinismo, o SMA inteiro se transfere para outro SMA com o intuito de fazer parte do mesmo; no Mutualismo, um agente ou o SMA de origem sai com a intenção de adquirir e transmitir novos conhecimentos, e posteriormente, voltar; já na de Predatismo, o SMA de origem visa a preservação da integridade de todos os seus agentes, assim, quando esta relação é ativada, o SMA predador se transfere para outros SMA com o intuito de dominar e transferir todos os seus agentes.

Para acontecer à transferência de um agente ou SMA para outro SMA existem duas condições: os SMA precisam estar conectados em uma rede IoT utilizando o *middleware* ContextNet para estabelecer uma conexão entre os diferentes SMA, dando a eles e aos seus agentes identificadores no ambiente; e a segunda condição é que o SMA de destino precisa permitir a entrada do novo SMA ou do agente transmitido.

O transporte de um agente é realizado enviando um arquivo do estado mental atual do agente para outro SMA. Primeiramente, é preciso criar um arquivo com as crenças, desejos e intenções atuais do agente (para isso, é preciso acessar a base de crenças, desejos e intenções, assim como a biblioteca de plano deste agente e acrescentar tudo que foi adquirido em tempo de execução); segundo, quando o agente chega ao SMA de destino, a sua permissão de entrada é avaliada, e é enviada uma mensagem de notificação para o SMA de origem; terceiro, se a notificação indicar que a entrada foi permitida, o SMA de origem apaga todas as informações dos agente no sistema de origem.

Para a transmissão dos agentes ou SMA planeja-se criar uma ação interna no Jason (.moveOut), onde são passados três parâmetros: o primeiro é o identificador do SMA de destino no ContextNet; o protocolo de transferência de agente utilizado (Inquilinismo, Mutualismo ou Predatismo); e o terceiro é o agente que será enviado caso o protocolo utilizado for o mutualismo e deseja-se enviar somente um agente. Com isso, o SMA de destino pode conhecer a intenção da transferência dos agentes e permitir ou não a entrada destes, contudo uma ontologia mais apropriada se faz necessário. A figura 1 apresenta as etapas para o protocolo de Predatismo.

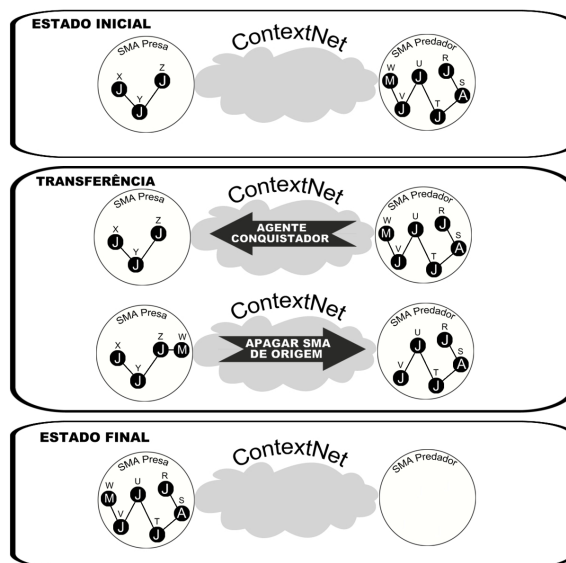


Figure 1. Relação de Predatismo e a dominação do SMA.

O protocolo de transferência de agentes foi proposto compreendendo a necessidade de mais mecanismos para a preservação da integridade dos conhecimentos dos agentes de um SMA, pois, os conhecimentos adquiridos por um agente em tempo de execução podem ser importantes, uma vez que, este pode adquirir ou transmitir conhecimentos, realizar desejos, intenções ou planos. Com isso, todas as relações do protocolo consideram a preservação dos conhecimentos dos agentes e em manter o agente sendo executado. A implementação do protocolo de agentes será feita como uma extensão da arquitetura de agentes presentes no *framework* Jason para possibilitar a utilização dos protocolos de transferência de agentes com as relações de inquilinismo, mutualismo e predatismo.

4. Testes executados

Nesta seção serão abordados os testes executados para avaliar o funcionamento dos protocolos de transferência de agentes aplicados em protótipos físicos no mundo real. Os testes de viabilidade dos protocolos de transferência de agentes foram realizados com o intuito de verificar a possibilidade de mover um agente de um SMA para outro, que consistiu em enviar um arquivo na extensão .asl (arquivo-fonte de um agente Jason) de um computador para outro através do *middleware* ContextNet.

Os testes executados neste trabalho foram aplicados em dois protótipos de carros autônomos com *hardwares* idênticos — com mesmos sensores, atuadores e controladores

— controlados por agentes ARGO situados em um SMA embarcado em uma Raspberry PI posicionada nos protótipos. Cada um dos protótipos possuem em seus respectivos SMA agentes dedicados ao controle do *hardware*, pois é mais eficiente dividir as tarefas e os controles de *hardware* entre diferentes agentes [Pantoja et al. 2016a]. Na Figura 2, podem ser vistos os protótipos dos carros.

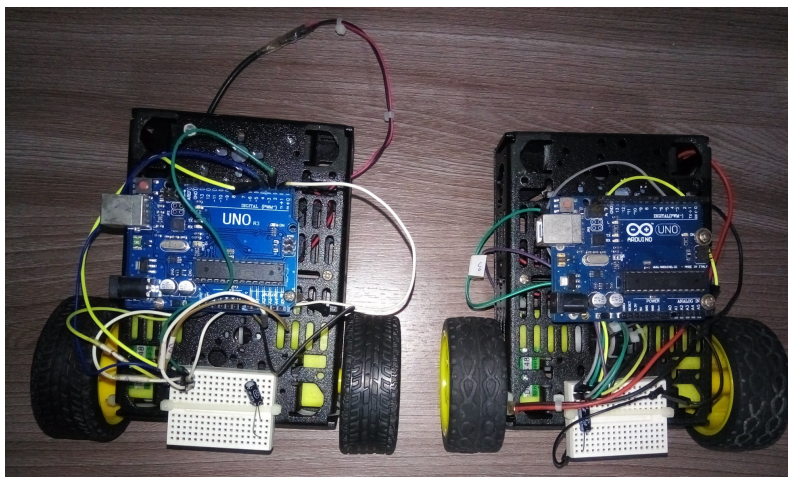


Figure 2. Prototipo dos carros autônomos.

O teste é constituído de dois veículos autônomos com SMA distintos, que são encarregados de uma missão hipotética, onde o SMA líder captura informações sigilosas para o sucesso da missão, todavia, este é atingido por um adversário de forma que seu *hardware* está irrecuperável. Para proteger a integridade dos seus conhecimentos e concluir a missão, o líder atingido aciona o protocolo de transferência de agentes utilizando a relação de predatismo, por conta da situação de risco iminente de parada total de funcionamento do seu *hardware* e, assim, transferir os seus conhecimentos, dominando o *hardware* de um outro SMA participante da missão.

Os protótipos dos carros são idênticos para que não haja problemas de compatibilidades, por exemplo, um SMA aplicado em um carro tem conhecimento para controlar um carro, agora se este SMA se transferir para um avião, não conseguirá controlar o seu *hardware* e, conseqüentemente, cairá. Neste caso, o protocolo de transferência de agentes com a relação de inquilinismo poderia ser aplicado, já que o SMA do carro pode se mover para o avião e fazer parte deste novo SMA até aprender a controlá-lo até que possa dominar ou auxiliar no controle do avião.

Ao se conectar ao ContextNet, todo SMA terá um código de identificação, representado pela identificação do seu agente Comunicador na rede IoT, para que o SMA de origem possa enviar agentes para o SMA de destino. Porém, como o teste proposto possui somente 2 dispositivos e sendo feito com *hardwares* idênticos, o SMA de origem não precisa atentar-se para questões de rastreabilidade e compatibilidade.

A construção dos SMA dos protótipos dos carros é similar, um agente ARGO para fazer a comunicação com o *hardware*, já, o controle das funcionalidades será feita por agentes Jason tradicionais. Desta maneira, o agente ARGO somente recebe as informações de leitura dos sensores e aguarda para atuar ou não no meio através dos seus atuadores. Porém, são os agentes Jason que através das informações recebidas do agente

ARGO e baseado em seus conhecimentos (crenças, desejos e intenções) tomam decisões para atuar ou não no meio físico enviando mensagens para o agente ARGO acionar ou não os atuadores.

5. Resultados e discussões

Nesta seção, serão abordados os resultados e discussões visando os detalhes da implementação dos protocolos de transferência de agentes, limitações e resultados.

Todos os agentes em Jason têm um arquivo onde fica armazenado o seu código-fonte, por exemplo, se houver um agente chamado bob no SMA, no projeto do Jason terá um arquivo bob.asl. Todavia, este arquivo armazena somente o código-fonte de criação do agente, com as crenças, desejos e intenções iniciais do agente, assim, os conhecimentos adquiridos em tempo de execução não estão presentes neste arquivo. Então, como o protocolo de transferência de agentes envia o arquivo asl de agentes, antes do envio, o protocolo adiciona todos os conhecimentos adquiridos em tempo de execução pelos agentes aos seus respectivos arquivos asl.

O momento da chegada ou a saída de um SMA ou agente também deve ser considerado, pois, quando um SMA é iniciado pelo *framework* Jason, este inicia o processo de criação de todos os agentes do sistema através dos arquivos asl de um projeto Jason. Todavia, o protocolo de transferência de agentes é executado pelo SMA já em tempo de execução, então é preciso integrar os agentes enviados aos SMA de destino e destruir as suas cópias no SMA de origem manualmente, além de apagar os arquivos asl no SMA de origem.

Atualmente, a biblioteca do Jason utilizada neste projeto conta com três tipos de arquitetura de agentes, a arquitetura tradicional do Jason, a ARGO e a *Communicator*, protocolo de transferência de agentes utiliza a *Communicator* para a transferência de agentes, porém, há uma discussão aberta para decidir se é preciso ou não criar uma nova arquitetura.

Além disso, o protocolo de transferência de agentes possui limitações na tomada de decisão para a escolha do SMA de destino, pois, no atual estágio de desenvolvimento, para utilizar o protocolo, o programador precisa conhecer os códigos de identificação dos SMA no ContextNet e definir o SMA de destino.

Por fim, os resultados dos testes com os protótipos dos carros foram satisfatórios e a transferência dos agentes e a execução do protocolo com a relação de predatismo funcionou corretamente. Vale ressaltar também que a velocidade de transmissão dos dados depende da rede que o ContextNet estará utilizando. Porém, como os arquivos-fontes dos agentes são de extensão asl (texto) e, portanto, pequenos, não necessitam de uma rede robusta.

6. Trabalhos Relacionados

Nesta seção serão apresentados trabalhos teóricos sobre SMA abertos e agentes móveis, que não possuem implementação e tem foco na segurança e no comportamento de um SMA ao receber um agente externo. Estes trabalhos não utilizam o transporte de agentes propriamente dito e nenhuma relação biológica; por sua vez, este trabalho propõe relações biológicas baseados na movimentação de agentes de um SMA para outro SMA em situações onde o *hardware* esteja danificado ou em perigo eminente.

O trabalho [Ossowski et al. 2007] propõe uma arquitetura abstrata chamada THOMAS, onde esta fornece a solução de problemas e limitações na programação de agentes, com a possibilidade de programar SMA abertos. Porém, não aborda a transferência de agentes para outros SMA, somente a relação entre agentes em um ambiente aberto e o tratamento de negociação entre eles. Em [Sanchez-Anguix et al. 2012], não é levado em consideração à origem dos agentes que entram no ambiente aberto proposto, e nem situações como um agente se movendo para outro SMA. Em FIRE, [Dong-Huynha et al. 2004], são demonstrados, matematicamente, modelos de confiança que são projetados para garantir que apenas agentes bem intencionados serão usados dentro do contexto. Em [Calvo 2012], foi aplicado o sistema bio-inspirado na colônia de formigas como estratégia de agrupamento de sistemas robóticos que exploram e vigiam ambientes desconhecidos. Já os protocolos propostos neste artigo visam o transporte de SMAs ou agentes para outros SMA, abordando a recepção dos mesmos dentro dos sistemas de destino e os seus objetivos finais (conhecer, ensinar, sobreviver ou explorar).

7. Considerações Finais

Este trabalho apresentou o desenvolvimento dos protocolos de transferências de agentes, a aplicação de conceitos provenientes da biologia aplicados na relação entre SMA de origem e SMA de destino, e os testes realizados em protótipos para verificar a aplicabilidade dos protocolos no meio físico. Além disso, o foco foi no protocolo com a relação de predatismo, uma vez que o atual estágio de desenvolvimento está concentrado no mesmo.

Para trabalhos futuros pretende-se realizar alguns testes em meio físico com os demais protocolos, desenvolver uma extensão para o *framework* Jason para permitir a transferência de SMA inteiros ou agentes possibilitando o controle de acionamento dos protocolos de transferência. Além disso, existe um projeto de um laboratório inteligente feito em escala real no CEFET Maria da Graça pelos autores que será aplicado os protocolos de transferência de agentes.

Referências

- Augusto Wrede, J., Nakashima, H., and Aghajan, H. (2010). Ambient intelligence and smart environments: A state of the art. pages 3–31.
- Bellifemine, F. L., Caire, G., and Greenwood, D. (2007). *Developing multi-agent systems with JADE*, volume 7. John Wiley & Sons.
- Bordini, R. H., Hübner, J. F., and Wooldridge, M. (2007). *Programming Multi-Agent Systems in AgentSpeak using Jason*. John Wiley & Sons Ltd.
- Bratman, M. E. (1987). *Intention, Plans and Practical Reasoning*. Cambridge Press.
- Calvo, R. (2012). *Sistemas bio-inspirados para coordenação de múltiplos robôs móveis*. PhD thesis, Universidade de São Paulo.
- Chebout, M. S., Mokhati, F., Badri, M., and Babahenini, M. C. (2016). Towards preventive control for open mas - an aspect-based approach. In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, pages 269–274. INSTICC, SciTePress.
- David, L., Vasconcelos, R., Alves, L., André, R., Baptista, G., and Endler, M. (2012). A communication middleware for scalable real-time mobile collaboration. In *Enabling*

- Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2012 IEEE 21st International Workshop on*, pages 54–59. IEEE.
- Dong-Huynha, T., Jennings, N., and Shadbolt, N. (2004). Fire: An integrated trust and reputation model for open multi-agent systems. In *ECAI 2004: 16th European Conference on Artificial Intelligence, August 22-27, 2004, Valencia, Spain: including Prestigious Applicants [sic] of Intelligent Systems (PAIS 2004): proceedings*, volume 110, page 18.
- Endler, M., Baptista, G., Silva, L., Vasconcelos, R., Malcher, M., Pantoja, V., Pinheiro, V., and Viterbo, J. (2011). Contextnet: context reasoning and sharing middleware for large-scale pervasive collaboration and social networking. In *Proceedings of the Workshop on Posters and Demos Track*, page 2. ACM.
- Hübner, J. F., Bordini, R. H., and Vieira, R. (2004). Introdução ao desenvolvimento de sistemas multiagentes com jason. *XII Escola de Informática da SBC*, 2:51–89.
- Lazarin, N. M. and Pantoja, C. E. (2015). A robotic-agent platform for embedding software agents using raspberry pi and arduino boards. In *9th Software Agents, Environments and Applications School*.
- Matarić, M. J. (2007). *The robotics primer*. Mit Press.
- Ossowski, S., Julián, V., Bajo, J., Billhardt, H., Botti, V., and Corchado, J. (2007). Open mas for real world applications: An abstract architecture proposal. In *Proc. XII Conference of the Spanish Association for Artificial Intelligence (CAEPIA)*, volume 2, pages 151–160.
- Pantoja, C. E., de Jesus, V. S., and Filho, J. V. (2016a). Aplicando sistemas multi-agentes ubíquos em um modelo de smart home usando o framework jason. In *II Workshop de Pesquisa e Desenvolvimento em Inteligência Artificial, Inteligência Coletiva e Ciência de Dados (Workpedia)*. Universidade Federal Fluminense.
- Pantoja, C. E., Stabile Jr, M. F., Lazarin, N. M., and Sichman, J. S. (2016b). Argo: A customized jason architecture for programming embedded robotic agents. *Fourth International Workshop on Engineering Multi-Agent Systems (EMAS 2016)*.
- Sanchez-Anguix, V., Aydogan, R., Julian, V., and Jonker, C. M. (2012). Analysis of intra-team strategies for teams negotiating against competitor, matchers, and conceders. In *The 5th International Workshop on Agent-based Complex Automated Negotiations (ACAN 2012)*, pages 1–8.
- Stabile Jr., M. F. and Sichman, J. S. (2015). Evaluating perception filters in BDI Jason agents. In *4th Brazilian Conference on Intelligent Systems (BRACIS)*.
- Tissot-Squalli, M. (2009). *Interações ecológicas & biodiversidade*. Unijuí.
- Vasconcelos, I., Vasconcelos, R., Baptista, G., Seguin, C., and Endler, M. (2013). Desenvolvendo aplicações de rastreamento e comunicação móvel usando o middleware sddl. In *Salão de Ferramentas, Brazilian Symposium on Computer Networks and Distributed Systems (SBRC 2013)*.
- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. Wiley.
- Zhang, D., Ning, H., Xu, K. S., Lin, F., and Yang, L. T. (2012). Internet of things. *J. UCS*, 18:1069–1071.