

Aplicação de Runtime Verification ao Projeto de um VANT

Orientação: Prof. Leandro Buss Becker

Os sistemas ciberfísicos (do inglês Cyber-Physical Systems - CPS) têm sido cada vez mais aplicados a ambientes onde altos níveis de confiabilidade e segurança são exigidos, dentre os quais temos o controle de processos automotivos ou no ambiente aeroespacial [1]. Os Veículos Aéreos Não Tripulados (VANTs) são exemplo de sistemas ciberfísicos, onde o cumprimento de requisitos acerca de confiabilidade são fundamentais.

Entretanto, a validação das propriedades no projeto dos VANTs não é uma tarefa simples e exige normalmente a utilização de diferentes técnicas. Dentre estas técnicas é possível citar a realização de testes e simulações e também o uso de técnicas de verificação formal [2,3]. Sobre verificação formal, destaca-se a utilização do *Model Checking* (MC) [4], técnicas de verificação estática que busca a validação de propriedades por meio da construção de autômatos temporizados. No entanto, devido a complexidade destes sistemas, ou mesmo pela expressividade da linguagem utilizada, algumas das propriedades acabam por não ser satisfeitas, ou mesmo são satisfeitas parcialmente, o que exige a utilização de técnicas adicionais para garantir a satisfação dos requisitos das aplicações.

Neste contexto, a técnica de verificação descrita como *Runtime Monitoring* (RM) [5] tem sido utilizada de forma a complementar o processo de verificação MC aplicado no projeto dos VANTs. A mesma é caracterizada pela extração de informação do sistema em execução, sendo essas informações utilizadas para detectar e reagir de acordo com os comportamentos observados avaliando a satisfação ou violação das propriedades desejadas [6].

No RM a partir da especificação dos requisitos formais, monitores são implementados e essas estruturas integradas ao sistema desenvolvido por meio de instrumentação. Neste sentido, o RM pode ser utilizado com diferentes propósitos que incluem, monitoramento ou política de segurança, depuração, teste, verificação, validação, proteção a falhas, modificação de comportamento entre outras [7].

Considerando a necessidade de validação de propriedades das aplicações em tempo de execução a utilização de técnicas de simulação se fazem necessárias, garantindo um ambiente seguro e controlado para validação dos sistemas. Visando integrar componentes de hardware real das aplicações com elementos de software, de forma a proporcionar um ambiente mais próximo da aplicação final a técnica de simulação em *Hardware-In-the-Loop* (HIL) tem sido utilizado como proposta para validação das propriedades em um ambiente mais próximo do real [8,9].

Neste contexto a presente proposta tem por objetivo a integração de técnicas de verificação de RM ao processo de desenvolvimento de CPS, em especial ao desenvolvimento do VANTs. Para aplicação desta técnica se busca utilizar simulações com HIL para permitir análises em um ambiente mais próximo da aplicação real.

Esta proposta está inserida no escopo do projeto ProVant¹, projeto criado em 2012 no Departamento de Automação e Sistemas (DAS-UFSC), em parceria com a Universidade Federal de Minas Gerais (UFMG), tendo como objetivo o desenvolvimento completo de

1 <http://provant.paginas.ufsc.br>

aeronaves autônomas aplicadas aos mais diferentes contextos [10,11]. A proposta também se encontra no contexto de uma colaboração entre o DAS e o centro de pesquisa CISTER, no Instituto Superior de Engenharia do Porto (ISEP).

Bibliografia Relacionada

- [1] LEE, E. A.; SESHIA, S. A. Introduction to Embedded Systems - A Cyber-Physical Systems Approach. 2. ed. [S.l.]: Lee and Seshia, 2015. <http://leeseshia.org/>. ISBN 978-1-312-42740-2.
- [2] P. Derler, E. A. Lee, and A. S. Vincentelli, "Modeling cyber-physical systems," Proceedings of the IEEE, no. 1, pp. 13–28, Jan 2012.
- [3] M. Huth and M. Ryan, Logic in Computer Science, 2004, vol. 2142.
- [4] C. Baier and J.-P. Katoen, Principles Of Model Checking, 2008, vol. 950.
- [5] Delgado, N., Gates, A.Q., Roach, S.: A taxonomy and catalog of runtime software-fault monitoring tools. IEEE Trans. Softw. Eng. 30(12), 859–872 (Dec 2004).
- [6] Leucker, M., Schallhart, C.: A brief account of runtime verification. The Journal of Logic and Algebraic Programming 78(5), 293 – 303 (2009), the 1st Workshop on Formal Languages and Analysis of Contract-Oriented Software (FLACOS07).
- [7] Nelissen, G., Pereira, D., & Pinho, L. M.. A Novel Run-Time Monitoring Architecture for Safe and Efficient Inline Monitoring. In A. J. de la Puente & T. Vardanega (Eds.), Reliable Software Technologies -- Ada-Europe 2015: 20th Ada-Europe International Conference on Reliable Software Technologies, Madrid Spain, June 22-26, 2015, Proceedings (pp. 66–82). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-19584-1_5.
- [8] SHIXIANJUN; JIAKUN, S.; HONGXING, L. Hardware-in-the-loop simulation framework design for a uav embedded control system. In: Control Conference, 2006. CCC 2006. Chinese. [S.l.: s.n.], 2006. p. 1890–1894.
- [9] JUNG, D.; TSIOTRAS, P. Modeling and hardware-in-the-loop simulation for a small unmanned aerial vehicle. In: Aerial Vehicle," AIAA Infotech at Aerospace, Rohnert Park, CA, May 2007, AIAA. [S.l.: s.n.], 2007. p. 07–2763.
- [10] F. S. Goncalves, J. P. Bodanese, R. Donadel, G. V. Raffo, J. E. Normey-Rico, and L. B. Becker, "Small scale uav with birotor configuration," in Unmanned Aircraft Systems (ICUAS), 2013 International Conference on, 2013, pp. 761–768.
- [11] Fernando Silvano Gonçalves. Projeto da arquitetura de software embarcado de um veículo aéreo não tripulado. Programa de pós-graduação em engenharia de automação e sistemas, UFSC, 2014.