WASP Software Engineering Assignment

Enno Breukelman, KTH Royal Institute of Technology
cebre@kth.se

August 2, 2024

1 Introduction / Abstract of my research area

My PhD project is supervised at the Divison of Decision and Control Systems, in the group working on Secure Network Control. In this section, I will give a brief introduction to Control Engineering and then go into details of my PhD project.

A control system, in general, is comprised of two main components: A plant, whose states are to be manipulated, as well as a controller, designed to influence the behavior of the plant. Based on sensor measurements, the controller computes control inputs to the plant. Many modern industrial control systems, especially for physically distributed plants, feature digital controllers and communication between the controller and plant over a network.

The resulting combination of physical entities (i.e. the plant) with digital control algorithms are often referred to as cyber-physical systems. Their ubiquitous implementation makes them vulnerable to cyber-attacks, where adversaries target the network communication between the controller and plant. If an attacker has access to the measurements or control signals, it can infer private information about the plant or even manipulate the signals sent by the controller.

Traditionally, cyber-attacks are classified by three dimensions. Firstly, the concealment of information: *confidentiality*; secondly, the trustworthiness of data: *integrity*; and thirdly, the ability to use information or resources: *availability*. More recently, specifically for cyber-physical systems, an attack space has been introduced, which is spanned by *model knowledge*, *disclosure*, and *disruption resources*. These dimensions aim to classify attacks on cyber-physical systems in a more appropriate way.

In this PhD project, we target the development of quantitative risk analysis tools for such networked control systems. The continuous rise of machine learning algorithms and AI tools in the engineering world has also impacted the design of control systems. The controller can incorporate machine learning elements into its control algorithm, just like attackers can be equipped with learning capabilities. This can pose new challenges on formal claims regarding the analysis of the severity of attacks. Therefore, we include machine learning components as part of the control systems and the attacker in our analysis.

In general, I am interested in formal guarantees and quantitative claims from a theoretical perspective. This means that the work mainly evolves around analyzing algorithms theoretically and features less coding / implementation of complex AI tools or machine learning models. Furthermore, since I am still in the first year of the PhD project, my research has not yet included machine learning elements in the analysis of attacks on cyber-physical systems.

2 Two principles from Robert's lectures and how they relate to my research

In this section, two principles from Robert's lectures that relate to my research will be discussed. The principles are version control and validation vs. verification.

2.1 Principle 1: Version Control

The use of version control when implementing new control algorithms (mainly in Python) is a crucial part in the development process. As hinted at in the introduction, coding does not take up a large part of my research. In addition, the project I implement are of a smaller size and (so far) contain no machine learning components. The main aim is to see if the formally derived algorithms yield the expected results in numerical simulations. Nonetheless, version control is a key element in the development of the code. The main arguments are: **reversibility**, **concurrency**, and **annotation**.

Using git allows me to mark states of the development of the code, which I can go back to, if I decide to implement a new function, which ends up not working as expected. Being able to reverse back to an old state, without creating multiple files, is an essential tool. In addition, concurrency features in the shape of multiple branches allows me to implement multiple ideas from the same starting point, again without having to create multiple files. Annotating the work by creating atomic commits, significantly improves the tracking of development. Especially, if I find a bug and I need to revert back to a previous state, the commit messages (or tags) help orienting through previous developments. Lastly, using version control in conjunction with online services such as GitHub, allows me to share the codebase with coworkers, that continue working on a similar project and want to re-use and build on previously written code.

2.2 Principle 2: Validation vs. Verification

For this section, I will interpret the concepts of validation and verification a little bit more freely, to apply it to my personal research. As previously mentioned, my code so far serves the purpose to create a proof-of-concept for an offline derived algorithm, ideally with formal guarantees. The principles of **verification** and **validation** can then be interpreted as follows:

1. **Verification**: Did I write the correct code to implement the method i formally derived?

2. **Validation**: Is the derived method correct? I.e., does the coded implementation validate the derived method?

In this interpretation, verification refers to the process of making sure, that the code I implement actually describes the method that I have derived using pen and paper. This is irrespective of whether the method I derived does what I intended it to do - it is only about whether the code reflects the method. Validation on the other hand can be seen as the process of using the results from simulations with the written code, to be able to assess whether the derived method is fulfilling its purpose. Questions like: do the simulations adhere to the formally derived formal guarantees? Is the derivation correct? Are aspects such as noise / rounding errors an issue?

3 Two principles from SAAB's guest lecture and how they relate to my research

In this section, I will talk about two concepts from SAAB's guest lecture. Unfortunately, this has very little overlap with my personal research. The two concepts are Behavioral software engineering and Autonomous decision making.

3.1 Behavioral software engineering

Behavioral software engineering was the overarching topic of this guest lecture. It is concerned with understanding and integrating human and behavioral aspects into software development processes. The only distant connection this has to my research at this point are the aspects discussed in the previous section about version control. When writing code, I aim to comment and annotate thoroughly, such that colleagues working on similar topics can use the code. Or create the code in such a way, that my supervisor can re-use it for further development. Besides that, software development does not play a large role in my research.

3.2 Autonomous decision making

Another large part of SAAB's guest lecture was autonomous decision making. In the context of flight control systems, having autonomous decision making processes can help reducing the cost be reducing manual labour. The decisions certainly need to be safe, such that no accidents happen. The concept of autonomous decision making cannot directly be transferred to my personal research. At least so far, we do not consider software, that makes autonomous decisions. Instead, we work towards getting formal claims on possible damages to a cyber-physical system through adversarial attacks. While the adversary may make autonomous decisions in a sense, we usually consider worst-case scenarios.

4 Two full papers published in one of the CAIN conferences

From the CAIN conferences, this section will discuss the following publications: *Developer Experiences with a Contextualized AI Coding Assistant: Usability, Expectations, and Outcomes* [1] and *Is Your Anomaly Detector Ready for Change? Adapting AIOps Solutions to the Real World* [2].

4.1 Paper 1: AI Coding Assistants

AI Coding assistants can be a helpful support and are used among colleagues, aiming to speed up the implementation process, in order to focus on the algorithm development rather than the coding itself.

4.1.1 Core idea(s) of the paper and why it/they are important

The core claims / ideas of the paper are that using an AI assistant while writing code can improve the workflow by making it faster and letting the software engineer focus on more import important tasks. There is a strong emphasis on a specific type of AI coding assistants, which feature contextualized capabilities. The context here refers to the codebase in which the software engineering is working. In contrast to general-purpose AI assistants that can only give unspecific answers to coding questions, a contextualized coding assistant knows the structure and the names of classes / objects / functions from the codebase. According to the authors, which conduct a study in the shape of a workshop and an interview, there is a large potential in the use of contextualized AI coding assistants *if used properly.*

4.1.2 How do they relate to my own research?

The main relation to my research is, that colleagues of mine are coding using AI assistants and recommend its usage to me. However, there are only a few colleagues who do, and the majority does not using coding assistants. The most common use case is looking up certain functionality using e.g. ChatGPT in the browser to then copy and modify the code into the local IDE. I have personally only briefly tried using GitHub Copilot for a few days, but found it not to be very useful. THe introduction of contextualized coding assistants seems very promising to me and spiked my interest while looking for papers for this homework.

4.1.3 How do my research and its results fit into a larger AI-intensive software project where one of the core ideas from the paper would benefit the project if applied? Describe both how the paper could help improve the project and how your WASP research would fit into the project.

Since I do not have any larger AI-intensive software projects in my PhD project, I can only comment on the use of a contextualized AI coding assistant on the smaller Python projects that I work on. Using a contextualized AI coding assistant can increase the speed at which I can implement any algorithms. The size of these projects is usually not too big. However, I can see that contextualized knowledge can improve the quality of the coding assistants' replies to questions. Though, according to the authors, the correctness of such contextualized coding assistants is certainly not at 100%, it can be a good guideline to begin with. Additionally, since the contextualized coding assistant from the paper also features a chat functionality, one can ask it directly questions about how to implement something given the variable names used in the code. It will be significantly easier to directly copy the answer from the assistant than to adopt a generic answer from a general-purpose AI coding assistant.

4.1.4 How could my research be potentially adapted/changed to make AI engineering in the project based on the idea of the paper even better/easier?

Another functionality that can be helpful, which is not necessarily related to a larger project, is the functionality to explain a certain code snippet. Suppose one continues the work of someone else's code, which is poorly commented. The contextualized coding assistant can then infer what a particular function does based on its codebase knowledge. Additionally, since the assistant can be used directly from within an IDE, efficiency is further increased since one does not need to switch between the browser and IDE. Lastly, the authors emphasize that the productivity boost can only be utilized if the proper use of the software is trained, i.e., one needs to familiarize oneself with the AI coding assistant until it becomes as helpful as possible.

4.2 Paper 2: Anomaly Detector

The second publication spiked my interest due to the fact that anomaly detectors are a common research topic in my field, and I was curious to see how they are used in the AIOps field.

4.2.1 Core idea(s) of the paper and why it/they are important

The main message of this publication lies in the changeability of operational data and the importance of retraining anomaly detection machine learning models in the use of monitoring systems for IT systems and operations. The authors analyze two different anomaly detection model maintenance techniques, which differ in how frequently the model is updated; a blind model retraining is simply a periodic retraining and an informed model retraining describes the retraining procedure after a change in the operational data is recognized. Additionally, the authors consider two different techniques in terms of the used data for retraining: A full-history vs. sliding window approach. The results show that informed retraining performs as well as the blind retraining approach as long as a sliding window is used. In the other case, the blind (periodic) retraining performs better than the other (informed) retraining approach.

4.2.2 How do they relate to my own research?

While my research has not evolved around anomaly detectors (yet), it does not directly influence my work. However, they are widely discussed and they embed very well into the larger scope of the risk analysis tools for adversarial attacks. The results can be useful when creating a machine learning anomaly detection in a larger attack prevention scheme. The authors compare multiple approaches of state-of-the-art anomaly detection models and analyze the impact of different window sizes. Unfortunately, the technical descriptions of the anomaly detection models are very general and not very insightful. Nonetheless, the presentation of results regarding the impact of retraining methodologies and window length is elaborate. Additionally, there are references to implementation repositories for the anomaly detection models.

4.2.3 How do my research and its results fit into a larger AI-intensive software project where one of the core ideas from the paper would benefit the project if applied? Describe both how the paper could help improve the project and how your WASP research would fit into the project.

This question does, unfortunately, only very remotely apply to this paper / any large AI-intensive software project. The paper describes machine learning tools to detect anomalies in operational data, such as CPU usage or similar. This can hardly be applied to a *large AI-intensive software project.* Nonetheless, the presented research can help my WASP research, as described in the previous subsection: anomaly detection models are closely related to the field of secure network control for cyber-physical systems.

4.2.4 How could my research be potentially adapted/changed to make AI engineering in the project based on the idea of the paper even better/easier?

Again, this paper did not focus directly on AI engineering. The inspiration it does give, however, suggests the use of machine learning anomaly detection models and recommends using periodic model retraining. This is a result, I can potentially use in a project in my future research and thus make anomaly detection in a larger attack prevention scheme even better.

References

[1] G. Pinto, C. De Souza, T. Rocha, I. Steinmacher, A. Souza, and E. Monteiro, “Developer Experiences with a Contextualized AI Coding Assistant: Usability, Expectations, and

Outcomes," in *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI.* Lisbon Portugal: ACM, Apr. 2024, pp. 81–91. [Online]. Available: https://dl.acm.org/doi/10.1145/3644815.3644949

[2] L. Poenaru-Olaru, N. Karpova, L. Cruz, J. S. Rellermeyer, and A. Van Deursen, "Is Your Anomaly Detector Ready for Change? Adapting AIOps Solutions to the Real World," in *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering - Software Engineering for AI.* Lisbon Portugal: ACM, Apr. 2024, pp. 222–233. [Online]. Available: https://dl.acm.org/doi/10.1145/3644815.3644961