Report on the outcomes of a Short-Term Scientific Mission[1]

Action number: CA20111

Grantee name: Elvis Gerardin KONJOH SELABI

Reference: E-COST-GRANT-CA20111-c5bcd172

Details of the STSM

Title: Model-Checking Smart Contracts

Start and end date: : 14/04/2025 to 25/04/2025

Description of the work carried out during the STSM

Description of the activities carried out during the STSM. Any deviations from the initial working plan shall also be described in this section.

(max. 500 words)

During this Short-Term Scientific Mission (STSM), we carried out a thorough investigation of the current state of the art in smart contract verification, with a specific focus on the applicability and limitations of model checking (MC) techniques in the context of data-aware behavioral models. The central question addressed during the STSM was whether existing model-checking tools and frameworks (especially those based on data-aware Petri nets (DPNs)) are capable of reasoning about smart contracts modeled as Data-Aware Finite State Machines (DAFSMs), which incorporate rich data structures and control-flow semantics.

The first stage of the work involved a comprehensive literature review and an understanding of existing tools. Tools such as VeriSolid, FSolidM, Mythril, Certora, and DAppVerify were analyzed in relation to the expressive capabilities required by our DAFSM-based coordination model. We identified significant limitations in these tools, especially regarding the representation and verification of contracts that involve dynamic data manipulation, parametric roles, and modular interaction patterns (core aspects of our model based on behavioral types).

A series of technical meetings and brainstorming sessions were held with researchers at DTU to discuss the formal underpinnings of our model and to assess whether DPN-based model-checking approaches could feasibly support verification tasks for DAFSMs. One major insight from these meetings was the recognition that while DPNs offer powerful abstractions for representing both control

[1] This report is submitted by the grantee to the Action MC for approval and for claiming payment of the awarded grant. The Grant Awarding Coordinator coordinates the evaluation of this report on behalf of the Action MC and instructs the GH for payment of the Grant.

flow and data, mapping the full semantics of our behavioral model onto DPNs presents challenges in terms of tractability and semantic preservation.

We conducted manual modeling experiments using small but representative DAFSM examples and encoded them into DPNs using verification tools. These experiments allowed us to explore the extent to which key properties; such as deadlock freedom, reachability of safe states, and role-specific access control; could be checked. *From this, we derived an initial set of benchmarks and criteria for evaluating expressiveness, scalability, and tool interoperability.*

Although we adhered closely to the original working plan, some deviations proved necessary. In particular, the translation from Data-Aware Finite State Machines (DAFSMs) to Data-aware Petri Nets (DPNs) turned out to be significantly more complex than initially anticipated. This complexity arose due to semantic gaps and the absence of direct encoding support for certain constructs inherent to our model. For example, our "Simple Marketplace" case study could not be straightforwardly modeled in a way that would allow us to fully leverage the model-checking capabilities of existing DPN tools. This limitation stems from the fact that the marketplace model involves operations on numeric data variables subject to arithmetic constraints, a class of behaviors for which model checking is known to be undecidable. Moreover, the dynamic nature of certain datatypes, such as the set of participants, introduces additional challenges. These participants can be created and modified at runtime, leading to an unbounded state space and further complicating the application of traditional model-checking techniques. As a result, verifying properties over such dynamically evolving data structures proved to be beyond the direct capabilities of the existing DPN-based tools without introducing substantial restrictions or abstractions.

Description of the STSM main achievements and planned follow-up activities

Description and assessment of whether the STSM achieved its planned goals and expected outcomes, including specific contribution to Action objective and deliverables, or publications resulting from the STSM. Agreed plans for future follow-up collaborations shall also be described in this section.

(max. 500 words)

The STSM successfully achieved its core objectives, namely to assess the feasibility and expressiveness of applying model-checking (MC) techniques, particularly those based on Data-aware Petri Nets (DPNs), to smart contract models defined as Data-Aware Finite State Machines (DAFSMs). Through this collaboration, we analyzed the theoretical foundations and tool-level implications of integrating formal verification into a model-driven development approach for smart contracts, rooted in behavioral types.

One of the key achievements of the STSM was the identification of two promising strategic directions for enabling model checking on our DAFSM-based smart contract coordination model:

1. Code-Based Verification through Tool Integration:
   Given that our framework already supports the automatic generation of smart contract code from DAFSM models, one practical approach is to leverage the model-checking capabilities of existing code-based verification tools. These tools(such as VeriSolid or FSolidM or SolVent) can analyze the generated code directly. However, to ensure the correctness and faithfulness of the verification results with respect to the original model, we will need to formally prove a bisimulation between the DAFSM model and the generated code. This proof of semantic equivalence is critical to establish trust in the indirect verification pipeline.
2. Bounded Model Checking (BMC) of DAFSMs:
   The second approach involves adapting bounded model checking techniques to operate directly on the DAFSMs. By constraining the domain of the data variables and bounding the execution depth, classical BMC can be applied to check temporal and safety properties under certain

assumptions. This would allow us to bypass the translation to code and instead verify properties directly at the model level. Such an approach would require defining a BMC semantics tailored to our data-aware models and adapting the existing tools accordingly.

The broader goal remains the development of a lightweight, developer-friendly verification toolchain, enabling smart contract developers to reason about the correctness of their contracts directly at the model level rather than relying solely on analysis of the generated code. This abstraction is essential to enhance productivity and trust in the development process while hiding unnecessary low-level complexity from developers.

The STSM has led to meaningful contributions to the objectives of EuroProofNet WG3, particularly in advancing formal verification techniques and methodologies applicable to blockchain technologies. It has paved the way for integrating model-driven engineering and formal methods in a unified framework for secure and verifiable smart contracts.

In the short term, the follow-up work will include the formalization of the model-code bisimulation proof and the implementation of encoding strategies to interface with existing MC tools. We also plan to define a set of representative benchmark models for empirical testing. In the long term, we envision the development of a dedicated verification framework supporting statistical model checking and quantitative analysis of behavioral smart contracts, as well as the dissemination of results through joint publications and tool development collaborations between our groups.

In short, this STSM successfully laid the groundwork for integrating verification into the smart contract development pipeline supported by behavioral types and DAFSMs. The work identified both promising directions and key bottlenecks, providing a foundation for continued collaboration and tool integration in the near future.

Host: Prof. Alberto Lluch Lafuente (DTU)

signed on 29/04/2025