Meeting on applying for a cost action

June 19 from 14.00 to 18.00

Venue:

```
Lille at École National Supérieure des Arts et Métiers (ENSAM)
Address: 8, Boulevard Louis XIV, 59046 Lille, France
Metro stop: Grand Palais
Room: 028.B (to be confirmed)
```

Zoom:

https://universityofmalta.zoom.us/my/adrianfrancalanza

Programme

```
- 14.30 - 14.45: welcome and intro about the action
- 14.45 - 16.00: individual presentations 1
- 16.00 - 16.30: coffee break
- 16.30 - 17.00: individual presentations 2
- 17.00 - 18.30: brainstorming towards shaping the proposal
```

Executive Summary

Recently, national and EU legislation and regulations on computational devices, AI, and software started requiring:

- interoperability,
- explainability,
- correctness.

Legislation and regulations are technical, formal but ambiguous, written in natural language; algorithms and applications are formal and precise - bridges between law and code needs to have several bricks (intermediate languages/models, going from natural language to logical descriptions and code), and back.

Many areas of our societies have bodies controlling the activity of public/private entities with respect to regulations (data privacy and protection bodies, financial markets control agencies, etc), but the tools to check how algorithms and code enforcing the regulations abide by the law are scarce and limited, often based on checking events occurred in the past and not verifying implementations to prevent violations.

In short, presently there is a huge discrepancy between:

A. the **regulations** governing the so-called digital society, on the one hand; and

B. the **digital systems** supposed to adhere to these regulations, on the other.

The full picture is more involved. Particularly, we identify two intermediate entities that can help to bridge this gap:

C. the **formal models** that capture, as close as possible, the regulations (e.g., logical rules, in the case of laws; workflows, in the case of administrative protocols);

D. the **verification methods** that are better suited to the validation of the formalised legislation, either with itself (for inconsistencies), and against the executable code (adherence).

Centred around these 4 entities (labelled **A**, **B**, **C** and **D** respectively, we identify the following concrete challenges:

From regulations to code (A $\rightarrow$ C) The extraction of formal description of regulations from legal texts. This can be automated via NLP and Generative AI techniques but it needs to be validated, being human intervention probably unavoidable.

From code to rigorous models (B $\rightarrow$ C) Methods for extracting formal model from code (which is formal itself) using techniques such as:

- abstract interpretation;
- (soft) Type/model inference;
- software verification;
  - execution logs as a *partial* extraction of the model;
  - instrumentation exposing parts of the system runtime state.

Evaluation of models (C $\leftrightarrow$ C) There are many levels here:

- internal inconsistencies within the codified formalised regulations;
- inconsistencies between the formalised regulation and well-formedness constraints or cross-checked against certain administrative protocols. We could apply/adapt logics over any graph-based models (LTSs/Kripke structures) that allow us to formally specify properties on these models.

From verification to descriptive models (D $\leftrightarrow$ C) Again, there are the usual concerns here:

- checking that verification methods are sound with respect to the models (correct);
- check that they are complete (expressive enough to handle most/all of the models and properties).

Evaluation of verification methods (D $\leftrightarrow$ D) We can determine properties about these verification methods such as:

- decidability issues;
- resource requirements and complexity analysis.

Generation of correct code (D $\leftrightarrow$ B) The purpose might be to automatically:

- check that the code implements correctly the algorithms defined by the verification techniques;
- overcome engineering challenges to implement the algorithm in host language of the code (e.g. what libraries to use, how to maximise modularity, code reuse, maintainability).

To this end, we are proposing the creation of a **COST Action** to initiate an interdisciplinary research network that brings researchers and innovators from the fields of Formal Methods, Natural Language and Automated Learning technologies, Fintech and European Law. The ultimate aim of the COST action to produce digital solutions that streamline compliance processes with EU legislation and automate compliance checking.