SHORT TERM SCIENTIFIC MISSION (STSM) SCIENTIFIC REPORT

This report is submitted for approval by the STSM applicant to the STSM coordinator

Action number: IC1404
STSM title: Automated Reasoning Support for Contract-Based Concurrent Engineering of Cyber-Physical Systems
STSM start and end date: 2018-10-23 to 2018-10-29
Grantee name: Ferhat Erata

PURPOSE OF THE STSM:

The problem of collaboration (concurrent, multi-view engineering) between engineers in different disciplines (e.g., electrical, control, software) is addressed by MPM4CPS at different levels. Once technical models are available, with precise semantics given in the form of simulate-able units (FMUs), co-simulation can be used (though scientific challenges remain when the FMUs are very different, such as when discrete-event and continuous models are mixed). In earlier stages of system development, collaboration is much harder, and Contract-Based Design can be used to avoid inconsistencies (as opposed to detecting and subsequently fixing them). As engineers in different disciplines may not understand each others' domains, (upper) ontologies may be used to describe links between concepts in the different disciplines. These links can subsequently be used to reason about inconsistencies between models in the different disciplines. The above (Contract Based Design and ontologies) were contributions in the PhD of Ken Vanherpen [7], built on the work of Benveniste et.al. [1]. The reasoning techniques in Ken's work are based on ontological reasoning, augmented with symbolic reasoning over algebraic equations (constraints). The extensive experience of young researcher Ferhat Erata in a wide scala of reasoning techniques will be used to investigate how Ken's techniques can be extended and made more efficient/scale-able. The concrete deliverable is a report on the fundamental problems and the different techniques that can be used to support automated reasoning for Contract Based Design in concurrent (multi-view) engineering in cross-disciplinary (CPS) projects. The report will mostly explore alternatives, though it is possible that some small prototypes will be built too. This is a contribution to both WG1 and WG2.

DESCRIPTION OF WORK CARRIED OUT DURING THE STSMS

Different engineering disciplines are involved in the design of a Cyber-Physical System (CPS) due to the combination of computational, networking and physical artifacts. This multidisciplinary approach leads to inconsistencies between shared properties, causing unexpected behaviors during the integration of the different design artifacts. To preserve consistency between those different views, Contract-Based Design (CBD) [1] is increasingly being used by system engineers to formalize an agreement between stakeholders from two or more engineering domains. Originating from contracts used in software engineering, such an agreement consists of a set of assumptions and guarantees, respectively corresponding to Require (pre-) and Ensure (post-) clauses (conditions) introduced by contract-based software development. These assumptions and guarantees describe the conditions under which a system promises to operate while satisfying desired properties.

Although the current state-of-the-art describes the abstraction/refinement, composition and multi-view analysis and verification principles of CBD, it lacks methods and techniques to identify the shared properties in

Funded by the Horizon 2020 Framework Programme
of the European Union

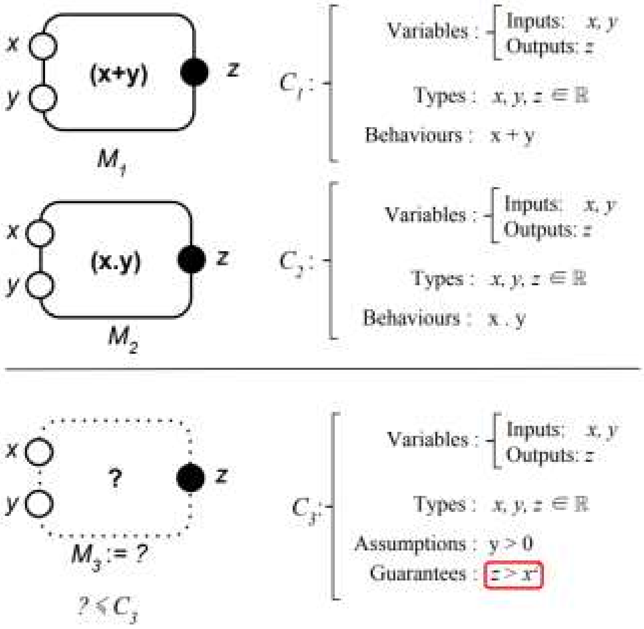
concurrent engineering (co-design) processes. To this end, The host institution introduced a framework which enables Contract-Based Co-Design (CBCD) by combining the current state-of-the-art of CBD with the principles of ontological reasoning [2, 3]. On the other hand, the applicant developed two frameworks, namely Tarski [4, 5] and AlloyInEcore [6], both of which provide automated reasoning support for the design of engineering artefacts. On the one hand, Tarski introduces a novel approach to dynamically configure the key semantic relationships among engineering products in first-order relational logic in order to provide automated reasoning support for traceability such as detecting inconsistencies and inferring missing traces (https://modelwriter.github.io/Tarski/). On the other hand, AlloyInEcore is a tool for specifying metamodels with their static semantics to detects inconsistent model parts, and completes partial models using finite model finders and Satisfiability-Modulo Theories (SMT) solvers (https://modelwriter.github.io/AlloyInEcore/). The tool automatically detects inconsistent model fragments with respect to formal semantics specified in contracts by the user using many-sorted first-order logic with transitive closure.

To this end, during the short-term scientific mission we investigated several ways in which the host institution's framework could be improved by exploiting finite model finders or other SAT/SMT-based analysis approaches for detecting inconsistencies and exploring design space to find suitable candidate solutions in Contract-Based Co-Design of Cyber-Physical Systems. During the STSM, the following research questions were examined in depth.

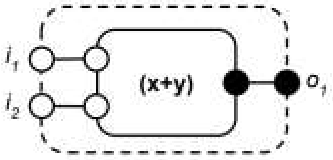
- in what ways the reasoning capabilities such as consistency checking and design space exploration of AlloyInEcore and Tarski tools can contribute to the implementation of theories of Contract-Based Design [6, 7] on block diagrams.
- how finite model finding and SMT-based reasoning techniques used in AlloyInEcore and Tarski can be employed to ontological reasoning [2] proposed by the host institution in the design of Cyber-Physical Systems.

DESCRIPTION OF THE MAIN RESULTS OBTAINED

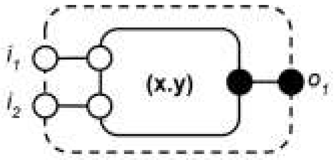
- A framework to provide automated reasoning support for finding design alternatives based on the partial information given in the form of assumption/guarantees and behavior of components/blocks considering the semantics of algrebraic causal block diagrams has been initially formulated.
- In the following, a simple setting is illustrated showing meaningful orderings of two algebraic operations in order to satisfy the contract of the abstract block, *M₃* defined by the user. The ultimate goal is to design and build an efficient solver that is capable of finding meaningful design alternatives as well as considering one or more quality attributes.


x
y
(x+y)
z
M_1
C_1 :
Variables : Inputs: x, y
Outputs: z
Types : $x, y, z \in \mathbb{R}$
Behaviours : x + y
x
y
(x.y)
z
M_2
C_2 :
Variables : Inputs: x, y
Outputs: z
Types : $x, y, z \in \mathbb{R}$
Behaviours : x . y
x
y
?
z
M_3 := ?
$? \preccurlyeq C_3$
C_3 :
Variables : Inputs: x, y
Outputs: z
Types : $x, y, z \in \mathbb{R}$
Assumptions : y > 0
Guarantees : $\boxed{z > x^2}$

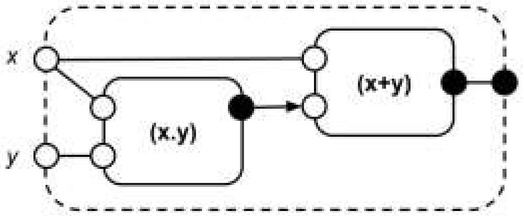
If we assume that user fixes the input and output pins, in the search space there must be a fix number of design choices reachable; however, only some of which are able to satisfy the guarantee contract specified by the designer, the output must be strictly greater than the square of the one of the inputs, *x*.



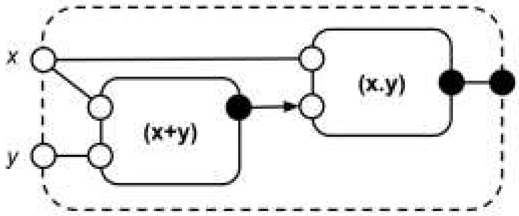
M_1



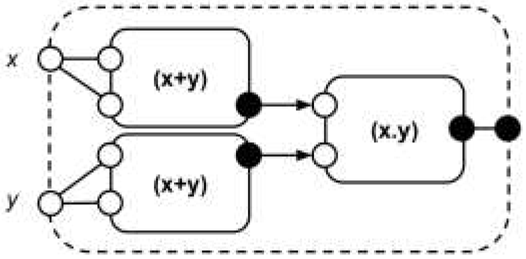
M_2



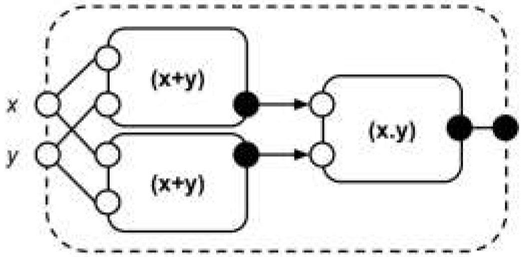
$(x + x.y) = x + xy$



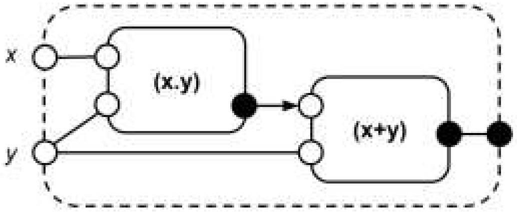
$x. (x + y) = x^2 + xy$



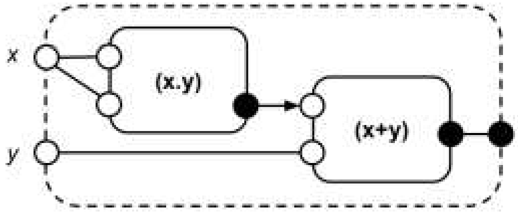
$(x + x).(y + y) = 4xy$



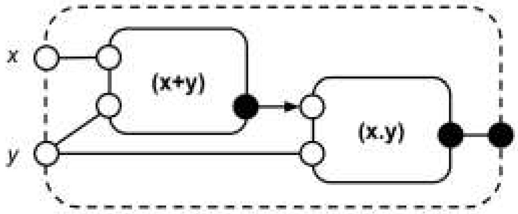
$(x + y).(x + y) = x^2 + 2xy + y^2$



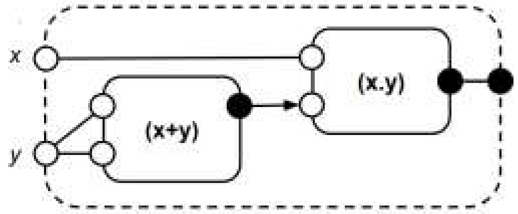
$(x.y + y) = xy + y$



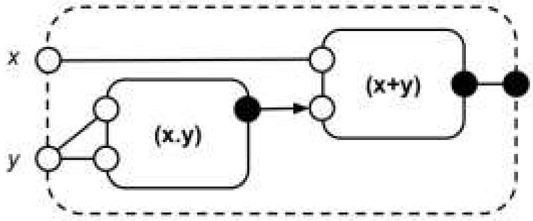
$(x.x + y) = x^2 + y$



$(x + y).y = xy + y^2$

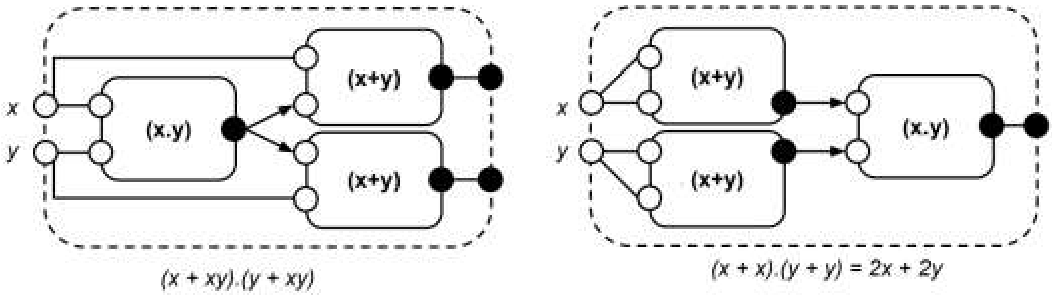


$(y + y).x = 2xy$



$x.(y.y) = x + y^2$

There are also other alternatives if the component and connecter language would have a slightly different semantics or the user does not limit the maximum number of usage of each block for each design alternative. In such a setting that does not limit the search space the reasoner might end up with an undecidable problem.



Nevertheless, a cost function as a quality attribute can be set for the design alterhatives. A non-functional property could be defined for the abstract block as part of its contracts. An example property may be the execution time of a block. And the reasoner can find a solution that has minimum execution time or below a certain value set by the user as a contract on the abstract block.

The solution space will definitely exponentially increase if more blocks with different behaviours are added to the scope or algebraic loops are also taken into consideration; however, we believe that this will bring new research challenges for us to investigate new clever techniques exploiting the domain specific knowledge as well as adopting existing techniques the automated reasoning community already invented in dealing with efficient computation for the combination of decision procedures.

FUTURE COLLABORATIONS (if applicable)

- The institution of the applicant will initiate a project to implement a framework in research collaboration with the host instituition, namely, “Design Space Exploration in Contract-Based Design of Causal Block Diagram (DSEInCBD)”. The first step is to introduce a minamal textual design language that supports
    - abstraction/refinement,
    - composition/decomposition,
    - and possibly view merging/view decomposition.

  Additionally, the language will allow users to partially define:
    - assumptions/guarantees
    - behaviours
    - on algebraic and/or discrete-time Causal Block Diagrams.
- The tool will provide Consistency Checking and Desing Space Exploration, automatically analyzing partial information collected from assumption and guarantee contracts of hierachical block definitions, any behavior definitions given, the axioms of the theory of refinement, composition, and conjunction operators and the overall semantics of the block diagram.
- Once the tool becomes stable, connectors for AADL, SysML or Simulink are also considered to be implemented.

REFERENCES

[1] Albert Benveniste, Benoît Caillaud, Dejan Nickovic, Roberto Passerone, Jean-Baptiste Raclet, et al.. "Contracts for Systems Design: Theory". Research Report RR-8759, Inria Rennes Bretagne Atlantique; INRIA. 2015, pp.86. https://hal.inria.fr/hal-01178467

[2] Vanherpen, Ken, Joachim Denil, Paul De Meulenaere, and Hans Vangheluwe. "Ontological reasoning as an enabler of contract-based co-design." In International Workshop on Design, Modeling, and Evaluation of Cyber Physical Systems (CyPhy), pp. 101-115. Springer, Cham, 2016.

[3] Vanherpen, Ken, Joachim Denil, István Dávid, Paul De Meulenaere, Pieter J. Mosterman, Martin Torngren, Ahsan Qamar, and Hans Vangheluwe. "Ontological reasoning for consistency in the design of cyber-physical systems". In 1st International Workshop on Cyber-Physical Production Systems (CPPS), pp. 1-8. IEEE, 2016.

[4] Ferhat Erata, Claire Gardent, Bikash Gyawali, Anastasia Shimorina, Yvan Lussaud, Bedir Tekinerdogan, Geylani Kardas and Anne Monceaux. 2017. Modelwriter: Text and model-synchronized document engineering platform. In Proceedings of 32th IEEE/ACM International Conference on Automated Software Engineering, (ASE '17). Urbana-Champaign, IL, USA, 2017, 928–933. DOI: https://doi.org/10.1109/ASE.2017.8115703

[5] Ferhat Erata, Arda Goknil, Bedir Tekinerdogan, and Geylani Kardas. 2017. A tool for automated reasoning about traces based on configurable formal semantics. In Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering (ESEC/FSE 2017). ACM, New York, NY, USA, 959-963. DOI: https://doi.org/10.1145/3106237.3122825

[6] Ferhat Erata, Arda Goknil, Ivan Kurtev, and Bedir Tekinerdogan. 2018. AlloyInEcore: Embedding of First-Order Relational Logic into Meta-Object Facility for Automated Model Reasoning. In Proceedings of the 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE ’18), November 4–9, 2018, Lake Buena Vista, FL, USA. ACM, New York, NY, USA. DOI: https://doi.org/10.1145/3236024.3264588

[7] Vanherpen, Ken. A contract-based approach for multi-viewpoint consistency in the concurrent design of cyber-physical systems. Diss. University of Antwerp, 2018.