Some lessons drawn from Systems Engineering (SE)

Freddy Kamdem Simo (see also)

September 6, 2021

SWISSED21

Objective

Show that challenges encountered with SE can also be encountered with Model-based SE

- As MBSE is simply an approach to the implementation of SE, **the fundamental goal/problem** - to satisfice* user needs under a multi-objective (cost, time, resources, etc.) and multi-domain (science, engineering, society, art, etc.) context - of SE should not be forgotten

Review these challenges and adverse-side effects and correlate them with some SE challenges

Overview of related works

Introduce MODEF – a computational approach to cope with modelling activities understood as a *technical-programmatic (or* product-and-project) *system (TPS)*

- **Show how it can contribute to some SE challenges**

Question about the development of SE

* A term coined by Simon Herbert

Outline

Modelling activity (overview and obstacles)

Some Systems Engineering challenges

Some results and lessons drawn from the literature

MODEF – contribution to systems engineering
life cycle state space exploration

Summary and perspectives and questioning

Modelling Activity (overview and obstacles)

MBSE and Model-driven

- yes, **can** foster better communications, representation, understanding, knowledge reuse and preservation, analysis, verification, etc.
- but, the same problems (i.e. operation, analysis, traceability, reuse, etc.) as with documents might be encountered, i.e. risk of model proliferation

Brown 2011

When and why do obstacles arise with modelling activity?

- If operated by autonomous stakeholders, within parallel and distributed tasks
- If involving specific and various business domains, methods, tools, formalisms, etc.
- If having different evolutions and long lifecycles (from weeks to years)
- Are creative and iterative

IT infrastructure and tools try to mitigate these obstacles, but the fundamental problem remains

Some Systems Engineering Challenges (1)

Difficulty in engineering of systems, limits of classical sciences

Organisation and *skills* of the (socio-technical) system that addresses the fundamental problem

Integration, validation, collaboration, communication and traceability

Leveraging Digital Engineering (IT)

- has impacts on SE ecosystem (people, data, operations, methods, tools etc.)

(Systems) methodologies/field

- lack firm scientific integrity and systemic nature
  (systematically following processes is not enough)
- have no or weak theoretical foundations
  (when does one work or fail? Why, how far does one work?)
- cannot assess its own completeness and articulate its potential future value in a principled way

Rousseau *et al* 2016

Some Systems Engineering Challenges (2)

1 Mission complexity is growing faster than our ability to manage it . . . increasing mission risk from inadequate specifications and incomplete verification.

2 System design emerges from pieces, rather than from architecture . . . resulting in systems that are brittle, difficult to test, and complex and expensive to operate.

3 Knowledge and investment are lost at project life cycle phase boundaries . . . increasing development cost and risk of late discovery of design problems

4 Knowledge and investment are lost between projects . . . increasing cost and risk: dampening the potential for true product lines.

5 Technical and programmatic sides of projects are poorly coupled . . . hampering effective project risk-based decision making.

Focus

6 Most major disasters such as Challenger and Columbia have resulted from failure to recognize and deal with risks. The Columbia Accident Investigation Board determined that the preferred approach is an "independent technical authority".

INCOSE SE VISION 2025

Some results and lessons drawn from SE evolution

Probability, queuing, game, information Theories   Operations Research   Systems thinking

Cybernetics   General Systems Theory   Dynamical Systems   Design theory

Many other applied or methodological works on systems engineering, complexity, systems theories, systemic, engineering design, etc.

Handbooks (INCOSE, NASA, SEBok, etc.) and standards (Mil-STD-499, etc.)

SE ROI, some findings*   Honour PhD thesis

- ROI cost (% SE cost on Program cost)
  - No SE (0%) => 7:1
  - Median level (7.2%) => 3.5:1
  - (20%) => **-2.8:1**
- **Technical leadership/management** is unique in providing optimun program success simultaneously in cost, schedule and stakeholder acceptance
- **No correlation founded** between SE and system technical quality

*8 SE Activities are considered (Mission definition, Requirement Engineering, System Architecting, System Integration, V&V, Technical Analysis, Scope management, Technical Management) for particular kinds of programs

Lessons drawn from product-and-project system (TPS) design

Few works formally address the coupling of both sides (programmatic and technical)

Sharon et *al* 2008, 2011
Vareilles *et al* 2015
Wynn and Clarkson 2017

Almost all works assume a design methodology for the technical side

Few works explicitly deal with the state of artefacts (e.g., models/data) related to the System to be made

Few evidence of formal requirements involved in model analysis

i.e. what is formally expected from the system models of technical-programmatic system (TPS)?

Tooling is both limited and a challenge (new tools and languages)

Kamdem Simo
2017 PhD thesis

MODEF – contribution to SE life cycle state space exploration (1)

Kamdem Simo 2017 PhD thesis

Challenges

Foundational challenge: how to formally deal with the technical-programmatic system (TPS)?

- Q1 How to abstract it?
- Q2 What are its possible models and for which goals?
- Q3 How to reason on these models to achieve the goals?
- Q4 How to take advantage in practice of the answers of Q1, Q2 and Q3?

Practical challenge

- Q5 How to better understand TPS and models/data of the system to be made it produces
- Q6 How to analyse and identify the impact of TPS' changes?
- Q7 How to help (using Q5 and Q6) in mastering TPS' evolution?

MODEF – contribution to SE life cycle state space exploration (2)

Principles

Abstraction of TPS

- Understanding TPS as a system (and federation of systems) at the level of its architecture
- Expressing the expectations (or desired requirements) from this understanding

Iteratively Modelling an architecture of TPS

- Representing state, process, and conceptual views related to TPS
- Projecting requirements on these views

Reasoning on this architecture to improve TPS operation

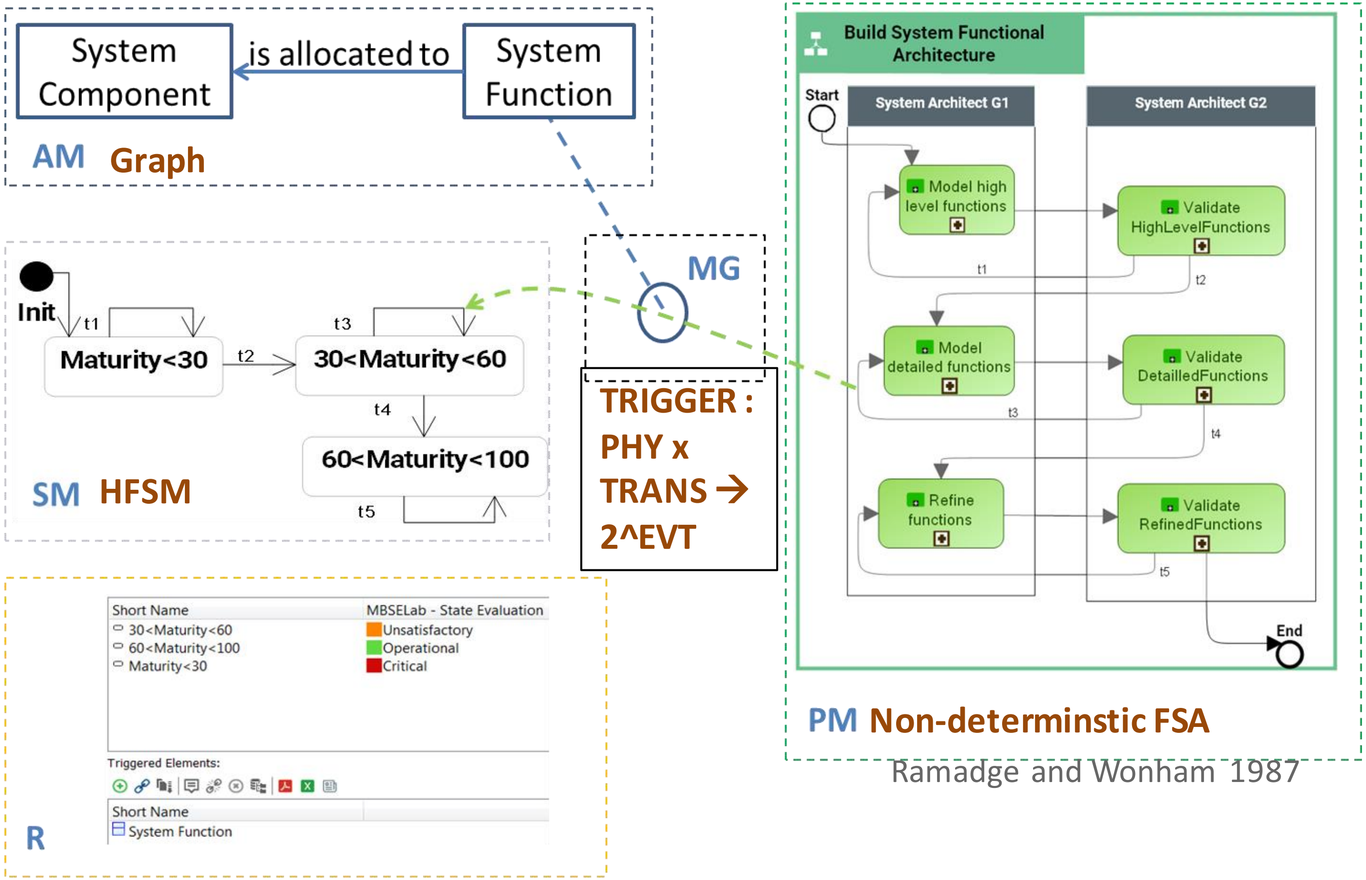
- Integrating state-of-the-art search/OR and model analysis algorithms

Implementation and beyond (open and modular enablers based on formal foundations)

Kamdem Simo 2017 PhD thesis

MODEF – contribution to SE life cycle state space exploration (3)

TPS Sample and instantiation of principles (formalism)

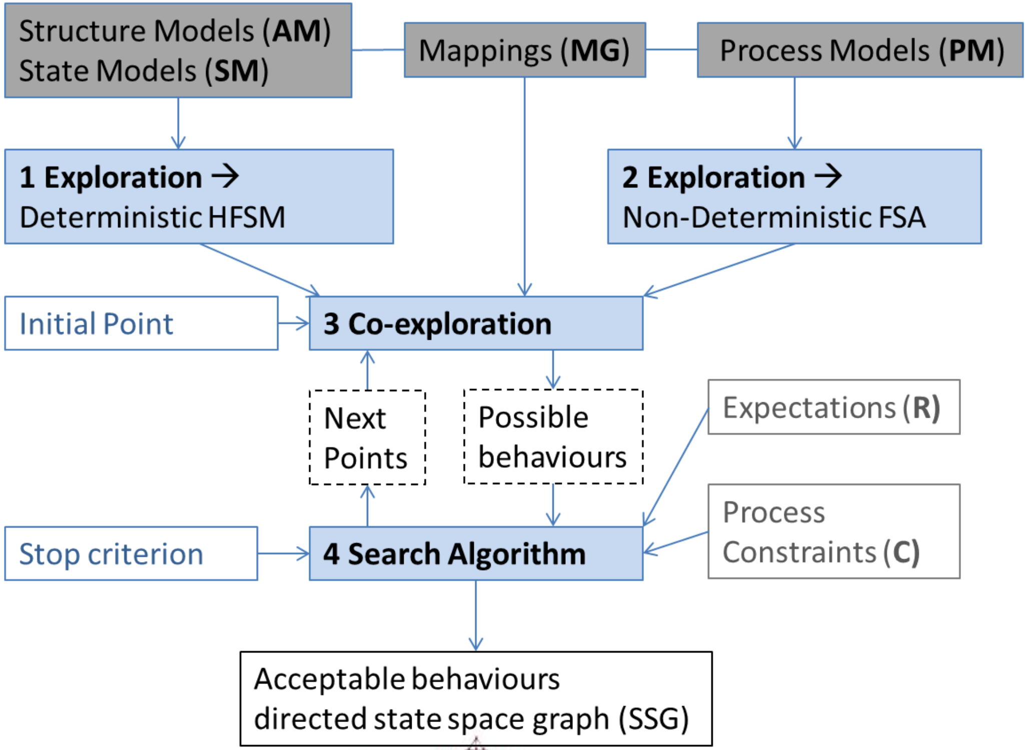


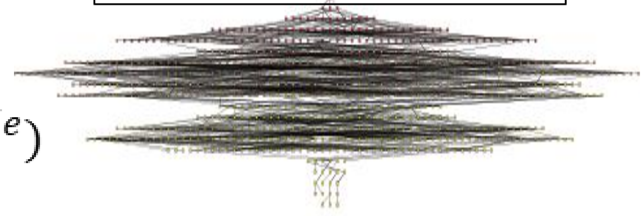
Id	A	P
1	m_1	$m_2 \preccurlyeq m_3, m_2 \preccurlyeq m_4$

From A/G contracts Benveniste *et al* 2012
to Assume/Preference formalism

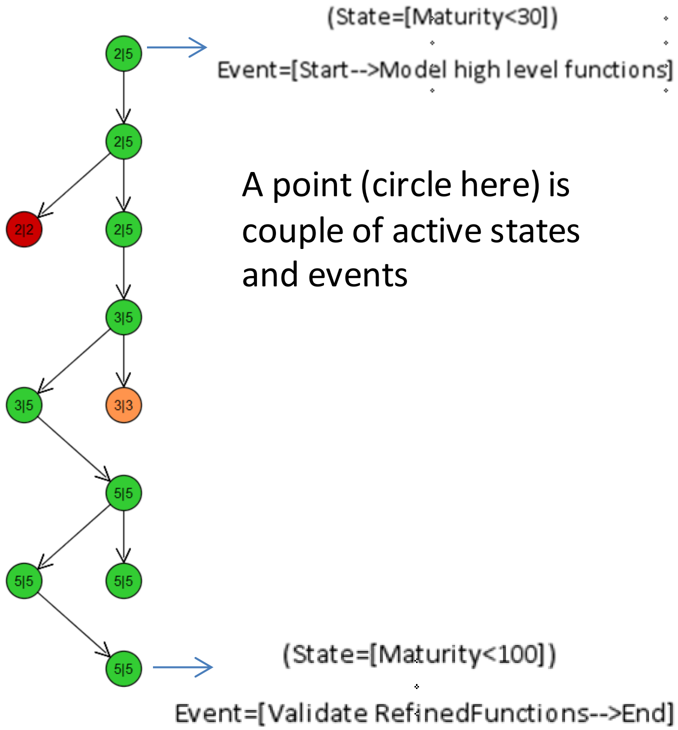
MODEF – contribution to SE life cycle state space exploration (4)

TPS Sample and instantiation of principles (reasoning and informing)





$O(b^{1+C/e})$



A point (circle here) is couple of active states and events

Information

- OK everything is ok
- KO something is not ok
- “ok” is problem and objective dependent

Summary and perspectives

MBSE may give rise to the same problems as those encountered with non-MBSE approaches

MBSE does not alter the fundamental goal or problem addressed by SE

Modelling or SE activities can become difficult to master

Systems methodologies lack firm (when, why and how far can one work?) foundations

The coupling of technical and programmatic aspects of SE is vital to achieve the fundamental goal of SE, but poses many problems

A computational approach can help navigating in the foreseeable states (or situations) of a technical-programmatic system for in fine supporting decision making, risk identification, audit, validation and even certification

Questioning on the development of SE

Can we experiment in SE, different from observational studies encountered in economics?

- engineering projects are unique, (almost) non-reproducible
- SE is still much driven (necessary for the art leg of SE) by best practices, experts -> challenges of reusing (prescriptive) knowledge
- Short term vs long term

What is the position of SE in relation to a science of artificial as presented by H A Simon ?

Are predictions useful, desirable and/or possible in SE?

- Not necessarily in a physical sense where a *predetermined harmony* can be assumed

In the problem(s) that systems engineering seeks to solve, what is, or is there, a more important or critical aspect?

- Entirety vs neatness

References

Brown 2011. Model-based systems engineering: Revolution or evolution. Thought Leadership White Paper. IBM Rational
Rousseau *et al* 2016. The synergy between general systems theory and the general systems worldview." Systema: connecting matter, life, culture and technology 4.1 (2016): 61-75
INCOSE SE VISION 2025. https://www.incose.org/products-and-publications/se-vision-2025
Honour PhD thesis. https://www.hcode.com/seroi/documents/SE-ROI%20Thesis-distrib.pdf
Sharon et *al* 2008. A project-product lifecycle management approach for improved systems engineering practices. INCOSE International Symposium, 18. 200
Sharon et *al* 2011. Project management vs. systems engineering management: A practitioners' view on integrating the project and product domains. Systems Engineering, 2011, vol. 14
Vareilles *et al* 2015. System design and project planning: Model and rules to manage their interactions. Integrated Computer-Aided Engineering, 22(4)
Wynn and Clarkson 2017. Process models in design and development. Research in Engineering Design, 2018, vol. 29
Kamdem Simo 2017 PhD thesis. https://tel.archives-ouvertes.fr/tel-01948889/document
Kamdem Simo *et al* 2021. Principles for coping with the modelling activity of engineered systems. Research in Engineering Design, 2021, vol. 32
Ramadge and Wonham 1987. Supervisory Control of a Class of Discrete Event Processes. SIAM Journal on Control and Optimization 25(1):206-230
Benveniste *et al* 2012. Contracts for System Design https://hal.inria.fr/hal-00757488/document
Kasser 2010. Seven systems engineering myths and the corresponding realities. Proceedings of the Systems Engineering Test and Evaluation Conference, Adelaide, Australia, 2010
Simon 1996. The sciences of the artificial. 3rd Edition

Contact and more on systems enginering

frks@protonmail.ch

linkedin.com/in/ydderfs

timeinx.com a proposal for an operational global vision of systems engineering