

Multimethod Approach to Modelling of Complex Sociotechnical Systems

Dr. Rudolph Oosthuizen
CSIR and University of Pretoria
roosthuizen@csir.co.za

Prof. Leon Pretorius
University of Pretoria
Leon.Pretorius@up.ac.za

Copyright © 2018 by Rudolph Oosthuizen. Published and used by INCOSE with permission.

Abstract. Systems Engineering has to solve increasingly complex problems through the implementation of solution systems. Solving contemporary complex problems requires Systems Thinking to address the emergent and evolutionary behavior. System Dynamics, Cognitive Work Analysis and Systems Engineering all have their roots in the systems approach and Systems Thinking, albeit with a different focus. Integrating System Dynamics and Cognitive Work Analysis through Model Based Systems Engineering process provides an approach to model complex sociotechnical systems. This paper describes and demonstrates an integrated multimethod modelling approach with these methodologies. The modelling approach should assist in addressing Systems Engineering's three evils; complexity, understanding and communication.

Introduction

The systems approach aims to support understanding the part of a system in context of the whole, while it is interacting with and adapting to the environment. This leads to the concept of Systems Thinking, which has evolved as the modelling of systems behavior (Hitchins 2008). Checkland (1990) defines Systems Thinking as investigating the emergence, hierarchy, communication and control as characteristics of systems.

According to Stensson (2010) Systems Engineering has its roots in Systems Thinking to solve problems by bringing systems into being. It applies interdisciplinary activities in support of design and development of a useful system to solve a perceived problem. Top-down and bottom-up analysis should be implemented to view the system as a whole, progressively breaking it down and to integrate synthesized components for the desired system properties (Meadows 2008, McLucas and Ryan 2005).

Holt & Perry (2008) warns that Systems Engineering is becoming difficult as it increasingly suffers the three evils of complexity, communication, and understanding. The complex problems to be solved are ill-defined, with a wider impact that may result in serious side effects. Engineering of these complex systems has to address the emergent and evolutionary behavior in systems and the environment. This requires a renewed focus on Systems Thinking (Walden et al. 2015, Sheard & Mostashari 2009).

Systems Thinking focusses on an outward analysis of the problem in support of the inwards focus on developing the solution system. System elements should be understood and conceptualized as a part of the larger whole. However, modelling and quantification of structure and behavior at this high level is hard (Hitchins 2008, Kasser and Mackley 2008).

Sterman (2000) asserts that System Dynamics provides a qualitative and quantitative modelling technique to support visualization and simulation of dynamic system behavior at high levels of abstraction. System Dynamics modelling is able to capture the softer qualitative as well as harder quantitative system characteristics of a problem situation. It provides an approach that is useful to solve non-linear, complex, time delay and feedback problems. However, it is not easy to integrate System Dynamics modelling with Systems Engineering approaches (Chang and Tu 2005, Kasperek 2014).

However, Brezinski (1999) noted the similarities between System Dynamics, and Systems Engineering, despite being applicable to different problem spaces. However, a multimethod modelling approach, which integrates of these concepts, is required to assist in solving complex problems (McLucas and Ryan 2005).

The authors have previously published a modelling methodology for Sociotechnical Systems (STS) that integrates Cognitive Work Analysis (CWA) and System Dynamics using a Design Science Research approach (Oosthuizen and Pretorius 2014). This methodology was based on an ad hoc modelling methodology without a proper supporting tool. This paper will update the modelling methodology using Model-based Systems Engineering (MBSE) based on the Systems Modelling Language (SysML) as the “glue” to model the CWA artefacts and assist in developing the System Dynamics models.

MBSE is a modern Systems Engineering approach to address the increasingly complex problems experienced in the industry. The purpose of modelling in Systems Engineering is to gain insight into complex systems and ideas to other stakeholders. It focuses on application of models, instead of a document-based text, for specifying, designing, integrating, validating, and operating a system (Walden et al. 2015, Hitchins 2008).

The research in this paper is an inductive literature analysis of MBSE and System Dynamics modelling to identify commonalities in modelling views. The aim is to define SysML constructs that support System Dynamics modelling. The paper will first describe Systems Engineering, with a focus on implementing MBSE. This is followed by the methodology based on Design Science Research which combines the two approaches to model complex STS. This is followed by a brief overview of CWA and System Dynamics. The paper concludes by demonstrating how SysML constructs can be applied to implement the modelling methodology.

Systems Engineering

Systems Engineering Process

Hitchins (2008) defines Systems Engineering as “... the art and science of creating whole solutions to complex problems ...”. The objective of Systems Engineering is to solve problems by bringing systems into being through the application of Systems Thinking. It consists of interdisciplinary activities required to support the design and development of a useful system that creatively exploit energy, materials and information within organized systems of humans, machines and the environment (Stensson 2010).

The Systems Engineering process has to ensure that the stakeholders’ needs are met in a cost-effective and timely manner. The process elicits and distils the needs of stakeholders along with the characteristics of the environment to define requirements and develop concepts. The functionality of the system is derived from interactions of the system with its operating environment and users (Walden et al. 2015).

Holt & Perry (2008) list the three evils of Systems Engineering as complexity, communication, and understanding. System complexity depends on the number of system elements and their interaction. Problems faced today can be ill-defined, with a wider impact, which may result in serious socioeconomic complexities and other environmental effects. Such problems may include climate change, natural hazards, healthcare, international drug trafficking, nuclear

weapons, nuclear energy, waste and social injustice important. An improper understanding of the problem and user needs leads to inaccurate requirements and the improper application of Systems Engineering. Communication problems between engineers and the stakeholders lead to inaccurate interpretations of requirements.

Sheard and Mostashari (2009) warn that the Systems Engineering approach requires a focus wider than only the technical aspects when addressing complex systems. This includes the context in which the system is to be engineered, developed, acquired and operated. Emergent system behavior cannot be understood by analyzing behavior of its elements in isolation. The elements that cause complexity should be identified in order to be managed and utilized (Walden et al. 2015).

Articulating and capturing a complex problem with a model or framework remains a challenge. Cloutier (2015) maintains that Systems Thinking is key to successful Systems Engineering through a multi-scale analysis with the support of modelling tools and methods to comprehend the “big picture” and appreciate the multiple perspectives and stakeholder objectives. A balance needs to be established between process (being systematic) and the application of Systems Thinking (being systemic).

Model-Based Systems Engineering

Wells (2011) agrees that modelling helps to manage the three evils of Systems Engineering and simplify understanding of complex systems. Ramos et al. (2012) defines a model as an explicit and incomplete representation or idealized abstraction of reality to aid its description and understanding. Models provide a visual representation of complex structures and processes for analyses and interpretation. Humans also use models to understand the implications of different solutions and communicate ideas to other stakeholders (Hitchins 2008).

MBSE is a modern approach to design and development of systems through application of models, instead of a document-based text, for specifying, designing, integrating, validating, and operating a system. The MBSE employs software modelling tools and formal languages to develop models. Various forms of models and constructs can be used to capture and represent the information and knowledge on problems (Walden et al. 2015).

In Systems Engineering, conceptual models describe selected aspects of the structure, behavior, operation and operational environment characteristics. Models are used to abstract complex systems to develop an architecture and generate data for analysis (Ramos et al. 2010, Walden et al. 2015). Modelling requires multiple views of the system for understanding, calculations and predictions concerning the system (Davis 2004).

MBSE utilizes standard modelling languages such as SysML or Unified Modelling Language (UML). SysML is a methodology and tool-independent language for modelling complex systems through diagrams on behavior, structure, requirements, relationships, and capabilities in various consistent viewpoints. The different views of the system may be organized in an architecture framework, to structure the model and analyze the system at different levels. The language utilizes a graphical notation, enhanced with relevant parameters, attributes and qualifying information (Wells 2011, Friedenthal et al. 2012).

Structural diagrams represent the physical parts of a situation with their logical relationships. Behavioral diagrams represent the parts of a situation and their causal interactions. The requirements views specify desired and undesired structural and behavioral properties with traceability to the system models. System requirements are captured in use case diagrams, requirements diagrams, and requirements tables. Parametric views provide the critical engineering parameters of the system for evaluating performance, reliability and physical characteristics required for trade-off analyses (Hause 2014).

Simulations address system functions and structure in defined scenarios to better understand the system through hypotheses testing. It supports learning about a real system through new or

unknown situations even though limited information is available. The model's performance under known conditions is compared with the performance of the real system for validation (Walden et al. 2015). However, modelling and simulation does have limits. Modelling of a complex and sociotechnical systems is difficult as it has to present the structure and behavior of human work in the system. Behavior is caused by dynamic interaction between the humans, operators, system elements and the environment. It is impossible to capture every reality with models, as the models' utility will diminish. Abstraction also requires making assumptions, which may cause incorrect solutions (Oosthuizen and Pretorius 2014).

Modelling Methodology for Complex Sociotechnical Systems

Modelling Methodology

Oosthuizen and Pretorius (2014) developed and demonstrated a modelling methodology for complex STS based on the Design Science Research (DSR) framework that integrated CWA and System Dynamics, as seen in Figure 1. The two basic activities in DSR are designing a novel and useful artefact for a specific purpose as well as evaluating its utility (March & Smith, 1995). CWA and System Dynamics are two fundamentally different methodologies, which are integrated to support modelling and learning as part of the design phase in the Systems Engineering process.

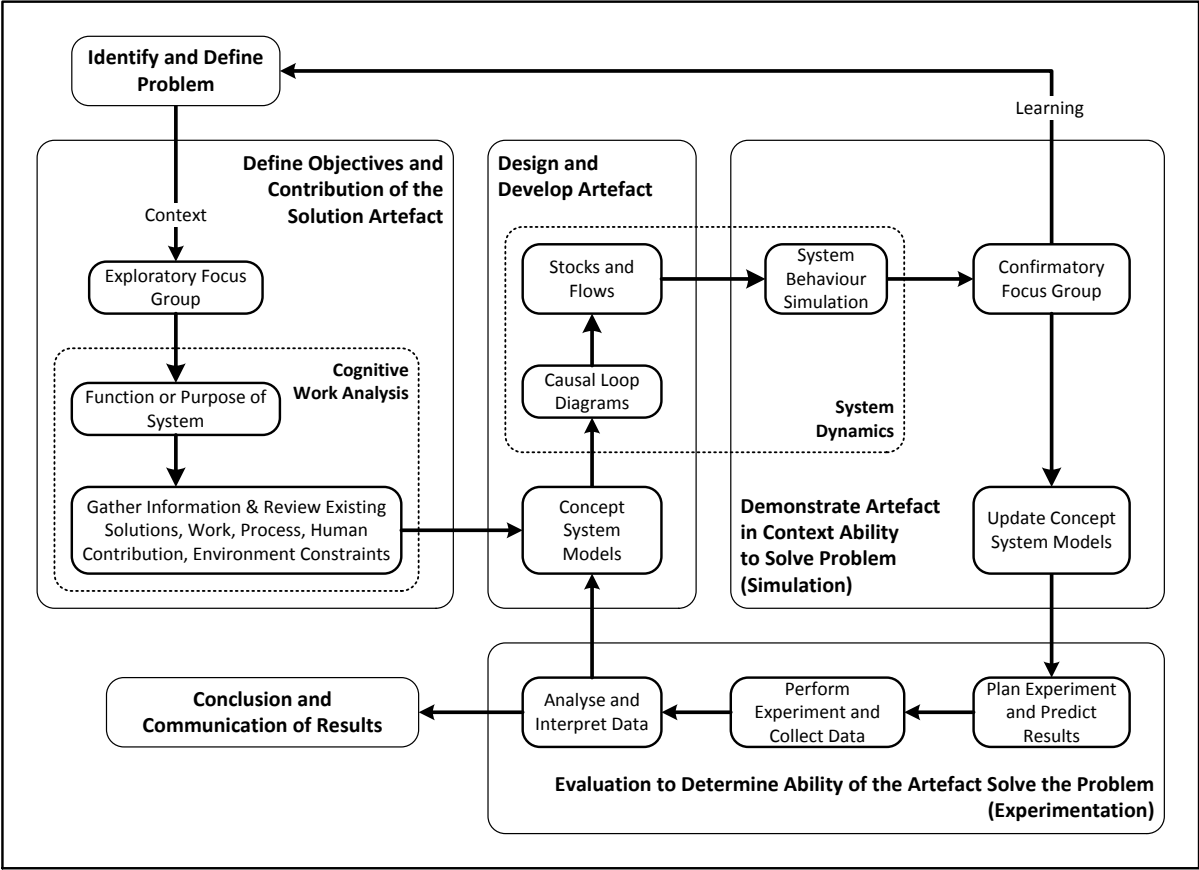


Figure 1: Modelling Methodology for Complex Sociotechnical Systems

The theory on STS, as developed by Trist (1981), provides a framework for modelling and analysis of complex systems. The concept of a “Sociotechnical” system addresses the interaction between “social” humans and “technical” systems. In a STS, people perform

purposeful work with technological artefacts within organizations to achieve an objective. Technological artefacts consist of the tools, devices, and techniques to transform inputs into outputs for economic gain. The social subsystem addresses structure of an organization with authority structures, reward systems, knowledge, skills, attitudes, values, and needs (Bostrom & Heinen 1977). The social and technical interaction can be non-linear as a result of unexpected, uncontrolled, and unpredictable complex relationships, which include being an open system operating in a complex environment. A complex system has elements with non-linear interactions (including feedback loops with delays) that cause non-deterministic, emergent, and unexpected behavior.

The first step in the modelling methodology is to identify and define the problem to be solved. This includes stating the dynamic hypothesis of the cause and effect of the perceived problem situation to provide focus and direction for the modelling effort. The next step assembles the available information from stakeholders. CWA helps to identify the components, interactions with cognitive and social requirements for the system from an ecological perspective (Dhukaram and Baber 2016, Oosthuizen & Pretorius, 2014).

Naikar et al. (2006) point out that a top-down analysis identifies the goals and purposes of the STS, which are integrated with a bottom-up view of available physical resources to achieve the purposes of the system. The output of the second step is a set of constructs to support the development of modelling artefacts in the third step. The artefacts consist of a model describing the structure and behavior of the problem and solution space through integrated views.

The next step is to demonstrate the models through System Dynamics simulations. Sterman (2000) noted that dynamics observed over a long time leads to dynamic patterns of system behavior that support learning about the underlying structure and other latent behaviors. The System Dynamics simulation outputs provide a visual representation of expected and counterintuitive behavior of the system and the environment to the stakeholders. This may lead to revisit the definition of the problem or an update to the models.

Oosthuizen and Pretorius (2014) initially developed this methodology without a recognized modelling approach for the conceptual system models. The only view used was a loosely defined “functional flow diagram”. The advent of MBSE provides a new opportunity to reinvestigate this, especially as part of developing models through integration of tools used in the Systems Engineering processes (Chavy-Macdonald et al. 2018).

The contribution of his paper is to propose the implementation of SysML to transform the CWA information into System Dynamics models. This can be seen as part of MBSE implemented with a data centric tool. SysML has many benefits over ad hoc modelling tools (Microsoft’s Visio, PowerPoint, pen and paper etc.). However, current MBSE models do not support analysis of the dynamic impact of the solution on the problem situation or environment.

Cognitive Work Analysis

Rasmussen et al. (1994) declared that CWA provides a formative and constraint-based framework that analyzes complex STS to uncover requirements, constraints, and implied affordances in the work environment. The theoretical roots of CWA are in Systems Thinking, Adaptive Control Systems and Ecological Psychology. The CWA approach has been applied in systems analysis, modelling, design, and evaluation of complex STS, which include command and control, aviation, health care and road transport (Jenkins et al 2009, Lintern 2012).

Lintern (2012) defined “work” as an activity aimed at accomplishing something useful with a purpose, values and success criteria. CWA enables mapping of the system’s affordances; what the system is composed of and what the system can do. This leads to defining the system users’ goal structures and knowledge (Naikar et al 2006). Although CWA include many phases, only the Work Domain Analysis (WDA) is applied in this modelling approach.

The WDA elicits and presents information on the system to understand its purpose, functional structure and the work environment. All possible behaviors that the system affords are mapped to the available physical elements. The goals and purposes of the system are defined in a top-down approach using means-end relationship. This is integrated with a bottom-up view of available physical resources for the human operators to achieve the purposes of the system (Naikar et al 2006, Manganeli 2013).

Vicente (1999) highlighted that the WDA is useful where technical systems, the environment and people interact dynamically, resulting in many possible instantiations. This analysis uses an Abstraction Decomposition Space (ADS) to model the work domain, by identifying system elements in the following level (Wells 2011):

System Purpose. This provides the reason why this specific system is being developed.

Values and Priorities. The reasoning process requires performance measures, principles, standards, or qualities, to be maintained while executing the process.

General Functions. This level provides the domain or general functions required to execute the work in satisfaction of the system purpose. These functions must be performed independently of the physical elements utilized.

Physical Functions. The physical functions are implemented through activating or using the physical objects.

Source Objects. These are the physical elements present in the work domain available to perform the work.

The top three levels of the hierarchy address the functional domain and are independent of the technology used in the system. The bottom two levels consist of the physical objects and the functions they perform in the system. Each level independently provides a complete description of the work.

System Dynamics

System Dynamics is a visual and mathematical modelling technique to study the dynamic behavior of systems due to feedback and delays using simulation at high levels of abstraction. This iterative and interdisciplinary approach views problems holistically to identify the counterintuitive behavior of the system due to policy based decisions. The System Dynamics process develops qualitative influence models to support quantitative simulation models (Kasperek 2014, Meadows 2008).

Sterman (2000) asserted that mental models of problems and systems tend to be dynamically deficient as they omit feedbacks, time delays, accumulations and non-linearities. Simulation is a practical way of testing mental models. Simulations require dynamics models built upon algebraic relationships. The soft variables ultimately become a hard (quantitative) representation of a particular problem expressed in precise mathematical way. Behavior observed over a long time leads to dynamic patterns of system behavior that support learning about the underlying structure and other latent behaviors (Meadows 2008).

A Causal Loop Diagram (CLD) represents the feedback structure of the dynamic system through capturing a hypothesis from stakeholder mental models on its dynamics and causes. CLDs consist of variables (selected nouns representing system properties) being connected by arrows to show the causal influences and relationships. CLDs represent interdependencies and feedback processes to capture the mental models of stakeholders (Chang and Tu 2005).

CLDs are influence diagrams that provide a simple way of showing how the parts of a system interrelate. Feedback systems have a closed loop structure containing balancing (or negative) and reinforcing (or positive) feedback loops. Sterman (2000) reminds us that it is important to identify the delays in each loop, as they are critical in creating inertia and dynamics. Feedback

in a system is one of the main causes of the complexity due to interactions between the agents or components over time.

System structure is the source of behavior, and consists of interlocking stocks, flows, and feedback loops. The Stock and Flow Diagram (SFD) shows the system structure with a richer visual language than CLD to distinguish between the parts of the system and the variables that causes them to change. Stock and flow variables are defined by a set of differential equations that can be solved to obtain the complex behavior of a system over time (Sterman 2000).

The boundary of the system under consideration is represented by clouds, which can be sources and sinks. Stocks indicate the state of the system as accumulations of resources through integrals of the inflow and outflow. They are things of importance in the system that can be measured. Stocks change over time through flows of elements through the system, and serve as the inertia and memory of the system (Sterman 2000).

Flows are the rates or derivatives of the net changes in stock. Changes in stock may also affect the flows through feedback loops. Feedback is the result of a causal connection between the stock and the flow. Feedback provides the effect of information on the system through applied decision rules, which are dependent on the stock levels to influence the flows (Brezinski 1999). The behavior of a system is defined over time in terms of growth, decline, oscillation, randomness and evolution. Stabilizing or balancing loops allows for the maintenance of an acceptable level (equilibrium) of stocks, by opposing the direction of change in the system. However, feedback can fail due to delays in information or incomplete and hard-to-interpret information (Meadows 2008, Sterman 2000).

Role of System Dynamics in Systems Engineering

Systems Engineering and System Dynamics has the same origins. However, Systems Engineering focuses on engineering problems and technical systems with hard variables while System Dynamics is more applicable to STS involving soft variables. System Dynamics models support framing, discussing and understanding complex issues and problems in a wide range of social and technical systems. Schwaninger and Ríos (2008) also noted that System Dynamics needs to be complemented with other methodologies. Synergy between System Dynamics and Systems Engineering can be identified as the following (Oosthuizen and Pretorius 2018, McLucas and Ryan 2005):

- System Dynamics provides a different view than Systems Engineering to assist in dealing with complex problems.
- System Dynamics is based on Systems Thinking with a holistic view of the problem and solution space that support modelling of the system at different system abstraction.
- The System Dynamics process is capable of capturing underlying assumptions and non-linear relationships in quantifiable “hard” and qualitative “soft” data.
- System Dynamics simulations with mathematical models of the systems’ structure enable visualization of complex dynamic patterns of behavior and emergent properties over time.
- System Dynamics explores paths into the future for design and operation decisions, including the robustness and vulnerabilities of policies and strategies.
- The System Dynamics models may also be used to interact with stakeholders to ensure the problem and its implications are correctly understood.

Improved Modelling Methodology for Complex Sociotechnical Systems

Model Based Systems Engineering for Complex Sociotechnical Systems

Hitchins (2008) stressed that a systemic approach must be followed to construct system models to guide questions during the design process. It encapsulates Systems Thinking in terms of boundaries, flows, relationships, feedback loops, and patterns between a system and its environment. This approach provides a way for designers to synthesize new emergent wholes instead of deconstructing those.

Implementing the modelling approach discussed above (Figure 1) through a formalized MBSE tool and process will improve modelling and analysis of Complex STS. Wells (2011) noted that data centric based software SysML tools enable reusing, data forwarding, relationships and traceability between diagrams in different views. This implies that changes in one diagram may be fed through to linked diagrams and subsequent phases.

SysML modelling tools and diagrams focus on how to model the components, relationships, and interactions in systems. But, currently SysML diagrams do not fully support analysis of the system affordances, user goal structures, and use strategies for the system of interest. Manganelli (2013) proposed applying SysML in the first three phases of the CWA framework. CWA support conceptualization and design of architecture models which can be mapped onto SysML. Also, these integrated techniques check and validate each other as well as bridge design and implementation (Dhukaram and Baber 2016).

Chang and Tu (2005) defined similarities in application of System Dynamics modelling and UML. Since then, SysML was developed. However, transforming SysML diagrams into System Dynamics models is not easy at this time. For example, SFDs are synchronous and sequencing diagrams are asynchronous, making it is difficult to decide which attributes or operations should be first in SFD sequencing. Sequencing diagrams do not have time delays, which are important to SFDs (Chang and Tu 2005).

The remainder of this section will describe how to implement MBSE through SysML to provide the “glue” between CWA and System Dynamics to enable the modelling approach for complex STS. The characteristics of SysML elements and diagrams are compared to the CWA and System Dynamics methodologies to establish the mapping.

Mapping System Modelling Language to Cognitive Work Analysis

The modelling methodology is demonstrated through an STS system example of a Command and Control system for Anti-Poaching Operations as previously published by the authors (Oosthuizen & Pretorius 2015). The WDA phase with the ADS decomposes the system from general functions to its physical form. The ADS structurally and logically map to SysML's structural diagrams. Structure analysis consists of system decomposition and behavior allocation. As seen in Figure 2, the SysML elements applicable in this phase include blocks, activities, use cases, properties (parts) and constraints. These elements may also be depicted in standard MBSE views, but the ADS helps in identification of the relationships between the views.

SysML stereotypes can be applied the different layers of abstraction in the ADS. “Constraint” blocks provide the purpose of the system. Values and Priority Measures can be modelled using “Property” or “Part” stereotypes. These can be related to the Measures or Effectiveness and other constraints for trade-off analysis. The General functions are the Use Cases for the whole system; what the system will be used for by the actors. Physical Functions are depicted by Activities and the Physical elements by “Blocks”.

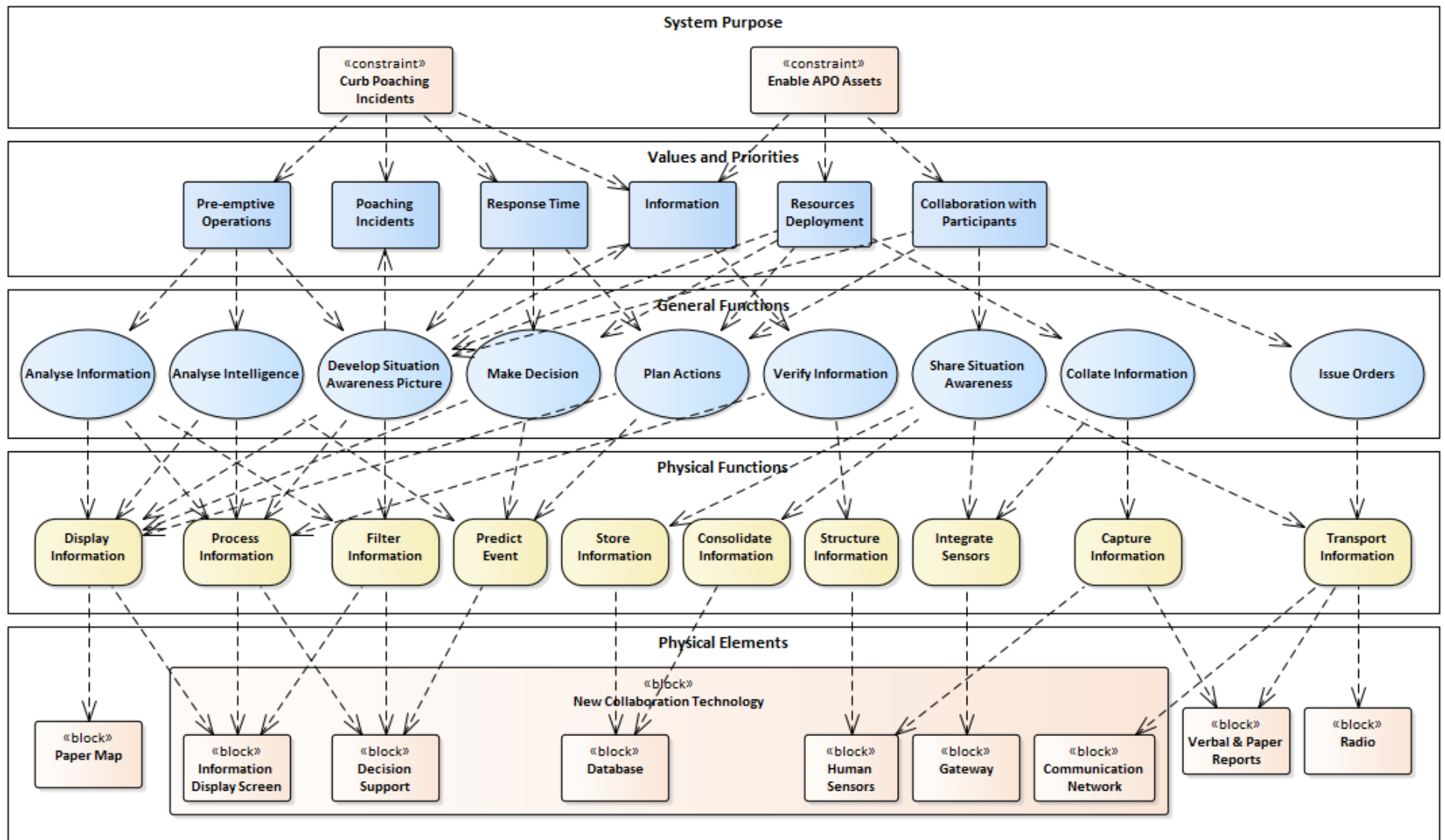


Figure 2: Work Domain analysis Using SysML

The SysML elements used in the ADS will form the foundation for modelling other views as part of the remaining CWA phases if required. The output of this phase consists of conceptual (logical) structure and behavior models of the STS, as seen in Figure 3 and Figure 4. These are different views of the same information captured in the model. The elements identified in the ADS can support development of Use Case Diagrams to identify functional requirements. The use case relationships in Figure 3 can be derived from overlaps between the ADS levels. The ADS is also helpful in placing model elements at the correct level of abstraction.

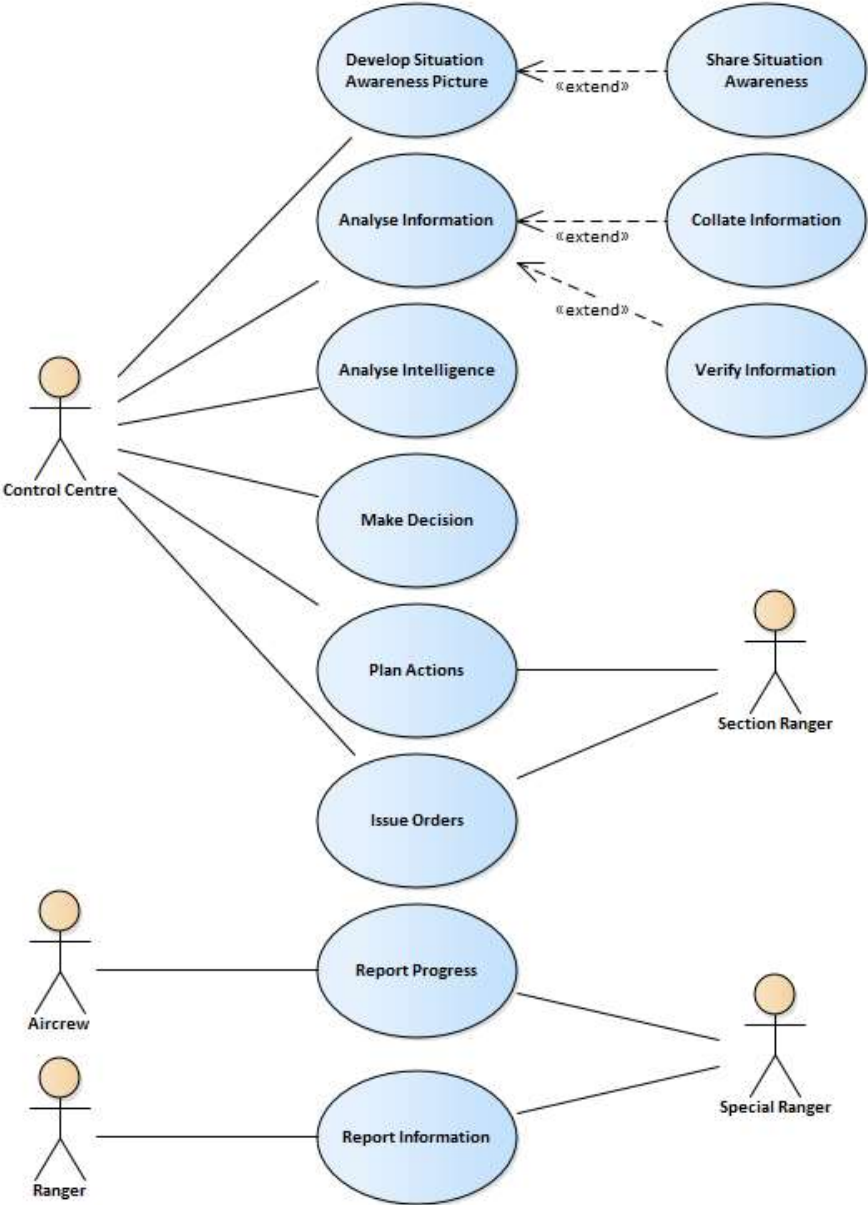


Figure 3: Use Case Diagram

The Activity Diagram in Figure 4 provides the control flow between the system element functions. This will help to identify the required interfaces between the system elements. These two diagrams will assist in development of the System Dynamics models for simulation. The SysML diagrams should be captured in a suitable modelling tool that manages the elements with relationships as well as traceability between views.

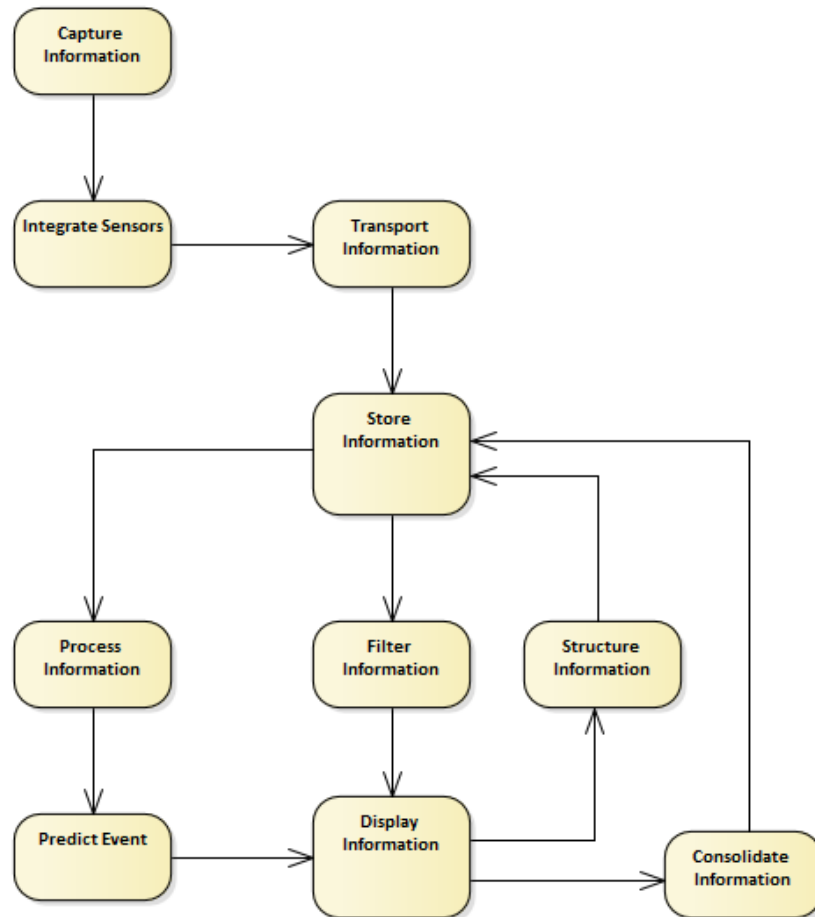


Figure 4: Activity Diagram

Mapping System Dynamics to the Systems Modelling Language

The next step in the STS modelling methodology is to generate System Dynamics models to simulate the dynamic behavior of the system and the environment. At this stage of the modelling approach, a number of views in SysML are derived from the CWA on the problem situation and the system of interest. Akbas (2015) observed that Use Case Diagrams support defining dynamic hypothesis, environment boundaries and system behavior through internal and external interactions for the System Dynamics modelling process (Chang and Tu 2005).

Causal Loop Diagrams. The Properties and Use Case from the ADS provide the variables in the CLD while the Activity Diagrams support identifying the links and loops between them through dependency relationships. As seen in Figure 5 a diagram, similar to a typical System Dynamics CLD, with multiple loops can be constructed, despite the dependency relationships being in the opposite direction. Note that dependency relationships are used, not item or information flows.

Stock and Flow Diagrams. Stock (level) characterizes the state of the system and provides the basis for actions which is similar to an Attribute in a SysML block, according to Chang and Tu (2005). The rate is the action to change the state of the level with time, which is similar to Operation of a SysML block as it presents the action which changes the state of system (Tignor 2003). As seen in Figure 6 a diagram, similar to a typical System Dynamics SFD, with multiple loops can be constructed, despite the dependency relationships being in the opposite direction. Note the different relationships between the elements. Between the blocks (stocks) item flows are used and between the Properties (variables) dependencies are used. These affect the flows between the stocks.

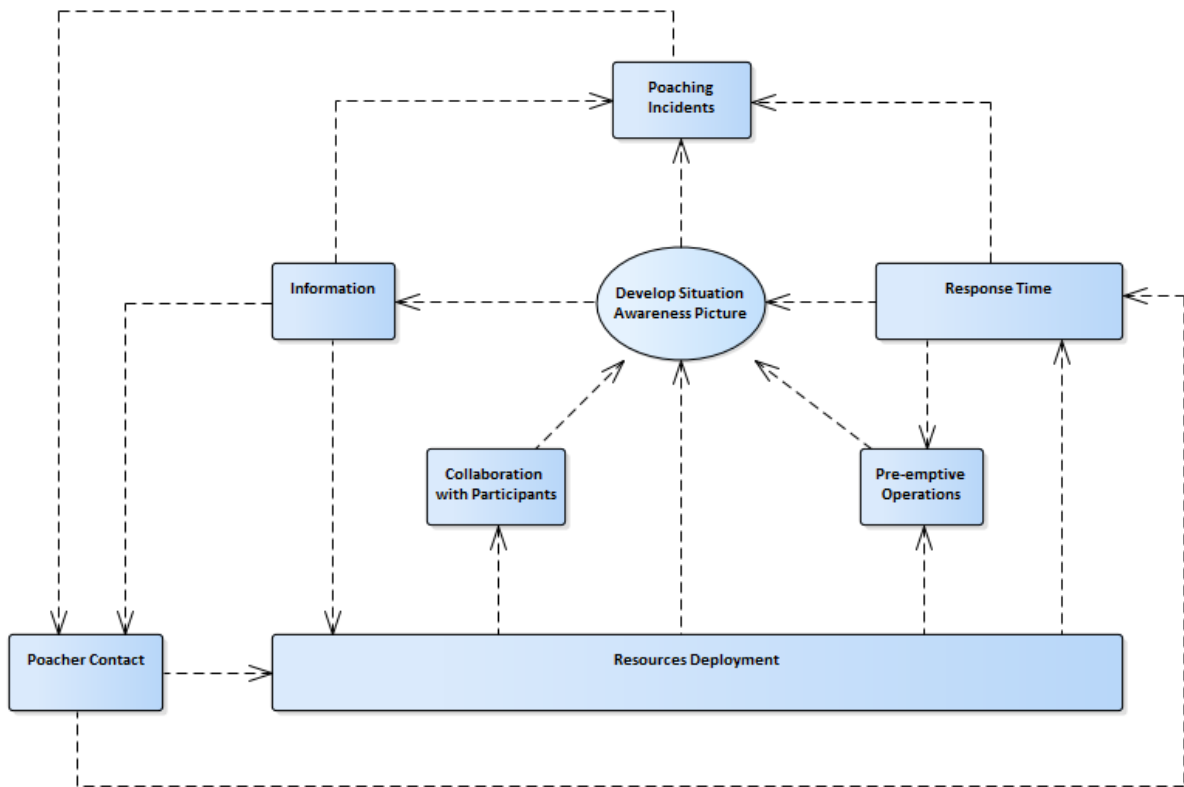


Figure 5: Causal Loop Diagram

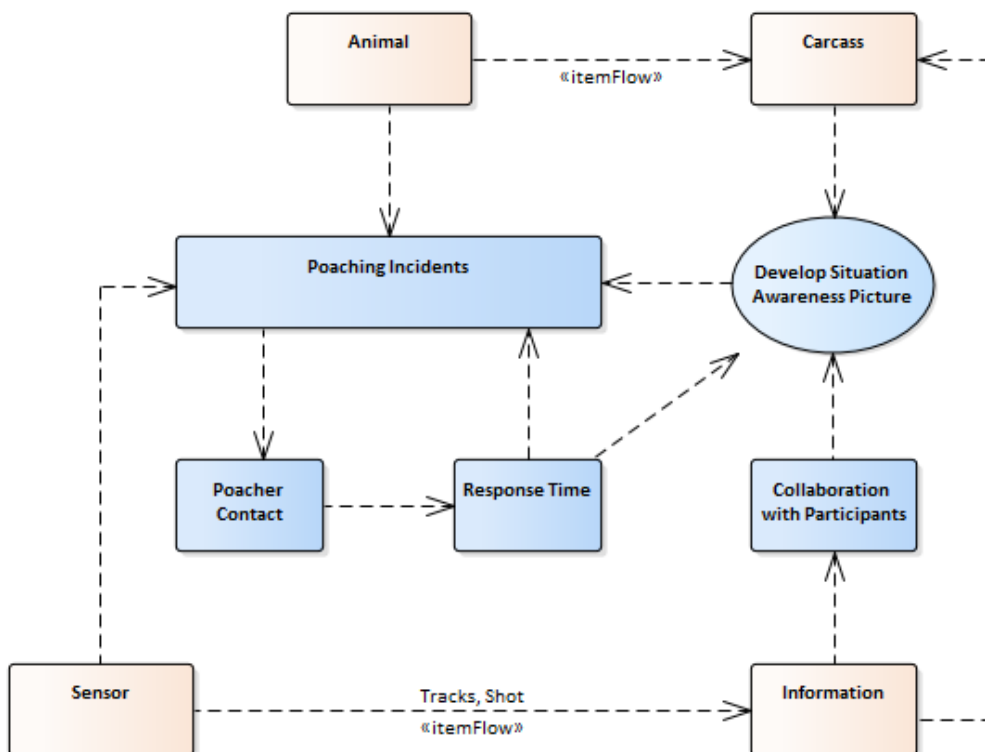


Figure 6: Stock and Flow Diagram

These diagrams provide a transition to convert from SysML through MBSE tools into a suitable System Dynamics tool for simulation. Although CLDs and SFDs in this paper are not representative as those elements used in standard Systems Thinking and System Dynamics

literature, these provide traceability to the views in MBSE. These views will assist in integrating System Dynamics into the Systems Engineering process.

Conclusion

Solving modern day complex problems requires Systems Thinking, which aims to understand the part of a system in the context of the whole, while interacting with and adapting to the environment to investigate emergence and other characteristics of systems. The behavior of a system cannot be determined by consideration of the parts in isolation.

This paper describes a multimethod approach for MBSE to integrate CWA and System Dynamics to model complex STS during the design phase of the Systems Engineering process. Engineering of these complex systems has to address the emergent and evolutionary behavior in systems and the environment through Systems Thinking. The initial complex STS modelling approach from the authors provided the integration between CWA and System Dynamics, but lacked the formality of an integrative modelling language.

The contribution of the WDA, as part of the CWA, to differentiate between the different levels of abstraction and decomposition of a multi layered system is quite clear. However, linking the SysML based models to System Dynamics still needs some work and verification. However, the modelling approach should assist in addressing the three evils of Systems Engineering. CWA and Systems engineering are based on Systems Thinking to help addressing the complexity of hard and soft variables in problems to be solved.

The ability of System Dynamics to simulate systemic behavior based on structure helps stakeholders to better understand the problem situation and impact of the solution system. Models are also better to describe multilevel complex requirements and solutions than text based artefacts. Integrating the diverse approaches using a standard MBSE notation, such as SysML, assist in communicating concepts and solution architectures with the stakeholders and the solution implementers.

Future Work

Further research on transferring SysML models to System Dynamics is still required. The ideal would be an automated exchange of .XML files between the tools. Another aspect to be investigated is the feedback loop to implement lessons learnt from the System Dynamics simulations as part of the initial stages of the modelling methodology.

References

- Akbas, A., 2015. Agent-Based and System Dynamics Hybrid Modeling and Simulation Approach Using Systems Modeling Language.
- Borshchev, A., & Filippov, A. 2004. From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools. In Proceedings of the 22nd International Conference of the System Dynamics Society, July 25 - 29, 2004, Oxford, England. no. 22.
- Bostrom, R. P., & Heinen, J. S. (1977). MIS Problems and Failures: A Socio-Technical Perspective. Part I: The Causes. MIS Quarterly 1(3), 17-32.
- Brezinski, L.J., 1999, June. 2 Systems Engineering, Systems Thinking, System Dynamics: Complementary Viewpoints. In INCOSE International Symposium (Vol. 9, No. 1, pp. 1183-1187).
- Camelia, F. and Ferris, T.L., 2016, July. Systems thinking in systems engineering. In INCOSE International Symposium (Vol. 26, No. 1, pp. 1657-1674).

- Chang, L.C. and Tu, Y.M., 2005, July. Attempt to Integrate System Dynamics and UML in Business Process Modeling. In Proceedings The 23rd International Conference of the System Dynamics Society July (pp. 17-21).
- Chavy-Macdonald, M.A., Oizumi, K. and Aoyama, K., 2018. A Model Framework for Determining Dynamic Architecture Goals in a Systems-of-Systems. *Disciplinary Convergence in Systems Engineering Research*, p.231.
- Checkland, P., & Scholes, J. 1990. *Soft Systems Methodology in Action*. Wiley, Chichester, UK.
- Cloutier, R., Sauser, B., Bone, M. and Taylor, A., 2015. Transitioning systems thinking to model-based systems engineering: systemigrams to SysML models. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(4), pp.662-674.
- Davis, P.K. 2004. *Space Modeling and Simulation: Roles and Applications throughout the System Life Cycle*. L. B. Rainey (Ed.). AIAA.
- Dhukaram, A.V. and Baber, C., 2016. A Systematic Approach for Developing Decision Aids: From Cognitive Work Analysis to Prototype Design and Development. *Systems Engineering*, 19(2), pp.79-100.
- Hause, M. C., 2014. SOS for SoS: A new paradigm for system of systems modeling. In *Aerospace Conference, 2014 IEEE* (pp. 1-12). IEEE.
- Hitchins, D. K. 2008. *Systems Engineering: A 21st Century Systems Methodology*. Wiley.com.
- Holt, J. and Perry, S., 2008. *SysML for systems engineering (Vol. 7)*. IET.
- Rasmussen, J., Pejtersen, A.M. and Goodstein, L.P., 1994. *Cognitive systems engineering*. Wiley & Sons, Inc.
- Jenkins, D.P., Stanton, N.A., Walker, G.H., & Salmon, P.M. 2009. *Cognitive Work Analysis: Coping with Complexity*. Ashgate Publishing, UK.
- Kasperek, D., Maisenbacher, S. and Maurer, M., 2014, March. Structure-based analysis of dynamic engineering process behavior. In *Systems Conference (SysCon), 2014 8th Annual IEEE* (pp. 83-88). IEEE.
- Kasperek, D., Schenk, D., Kreimeyer, M., Maurer, M. and Lindemann, U., 2016. Structure - Based System Dynamics Analysis of Engineering Design Processes. *Systems Engineering*, 19(3), pp.278-298.
- Kasser, J. and Mackley, T., 2008, June. Applying systems thinking and aligning it to systems engineering. In *INCOSE International Symposium (Vol. 18, No. 1, pp. 1389-1405)*.
- Lintern, G. 2012. *Work-Focused Analysis and Design*. *Cognitive Technical Work*, vol. 14, pp. 71–81. Springer-Verlag London Limited
- Manganelli, J.C., 2013. *Designing complex, interactive, architectural systems with CIAS-DM: A model-based, human-centered, design & analysis methodology* (Doctoral dissertation, Clemson University).
- McLucas, A.C. and Ryan, M.J., 2005. Meeting Critical Real-World Challenges In Modelling Complexity: What System Dynamics Modelling Might Learn From Systems Engineering. In *The 23rd International Conference of the System Dynamics Society*, Boston, USA.
- Meadows, D., 2008. *Thinking in Systems: A Primer*. Chelsea Green Publishing.
- Naikar, N., Moylan, A., & Pearce, B. (2006). Analysing activity in complex systems with cognitive work analysis: Concepts, guidelines and case study for control task analysis. *Theoretical Issues in Ergonomics Science*, 7(4), 371–394.
- Oosthuizen, R. and Pretorius, L., 2014, October. Modelling methodology for engineering of complex sociotechnical systems. In *INCOSE International Symposium (Vol. 24, No. s1, pp. 268-281)*.

- Oosthuizen, R. and Pretorius, L., 2018, The Role of System Dynamics in Systems Engineering. The 36th International Conference of the System Dynamics Society. In publication.
- Oosthuizen, R. and Venter, C., 2016. Implementing UPDM to develop command and control systems.
- Oosthuizen, R. and Pretorius, L., 2015. System Dynamics Analysis of Factors Impacting on Command and Control Technology Acceptance for Anti-Poaching Operations. American Society for Engineering Management 2015 International Annual Conference.
- Ramos, A.L., Ferreira, J.V., & Barceló, J. 2010. Revisiting the Similar Process to Engineer the Contemporary Systems. *Journal of Systems Science and Systems Engineering*, vol. 19, no. 3, pp. 321-350.
- Ramos, A.L., Ferreira, J.V., & Barceló, J. 2012. Model-Based Systems Engineering: An Emerging Approach for Modern Systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 1, pp. 101-111.
- Ryan, J., Sarkani, S., & Mazzuchi, T. (2014). Leveraging variability modeling techniques for architecture trade studies and analysis. *Systems Engineering*, 17(1), 10-25.
- Friedenthal, S., Moore, A. and Steiner, R., 2014. A practical guide to SysML: the systems modeling language. Morgan Kaufmann.
- Schwaninger, M. and Ríos, J.P., 2008. System dynamics and cybernetics: a synergetic pair. *System Dynamics Review*, 24(2), pp.145-174.
- Schwaninger, M., 2006. System dynamics and the evolution of the systems movement. *Systems Research and Behavioral Science*, 23(5), pp.583-594.
- Sheard, S.A., & Mostashari, A. 2009. Principles of Complex Systems for Systems Engineering. *Systems Engineering*. vol. 12, no. 4, pp. 295 – 311.
- Stensson, P. 2010. Thoughts about the Consequences of Inappropriate Application of Systems Engineering. In *Proceedings of the 7th European Systems Engineering Conference, Stockholm, INCOSE*.
- Sterman, J. D. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York: Irwin/McGraw-Hill.
- Sterman, J.D, 1994. Learning in and about Complex Systems. *System Dynamics Review*, vol. 10, no. 2-3. pp. 291-330.
- Sterman, J.D., 2002. All Models are Wrong: Reflections on Becoming a Systems Scientist. *System Dynamics Review*, vol. 18, no. 4, pp. 501–531.
- Tako, A.A. and Robinson, S., 2009, December. Comparing model development in discrete event simulation and system dynamics. In *Winter Simulation Conference* (pp. 979-991). Winter Simulation Conference.
- Tignor, W. 2003. Stock and Flow, and Unified Modeling Language Relationships. *Proceedings of 21th International Conference of the System Dynamics Society*.
- Trist, E.L. (1981). *The Evolution of Socio-Technical Systems: A Conceptual Framework and an Action Research Program*. Ontario Quality of Working Life Center, Occasional Paper no. 2.
- Vicente, K.J., 1999. *Cognitive work analysis: Toward safe, productive, and healthy computer-based work*. CRC Press.
- Walden, D.D., Roedler, G.J. and Forsberg, K., 2015, October. INCOSE Systems Engineering Handbook Version 4: Updating the Reference for Practitioners. In *INCOSE International Symposium* (Vol. 25, No. 1, pp. 678-686).
- Wells, W.H., 2011. Development of a cognitive work analysis framework tutorial using systems modeling language. *Electronic Theses and Dissertations*. 1727.