

The Role of Systems Thinking in Systems Engineering, Design and Management

Chan, W.T.¹

Abstract: Systems thinking is a widely recognized and subscribed-to concept. Many benefits are ascribed to systems thinking and its result - the holistic solution. Yet, there is a wide range of opinion as to what systems thinking really is, and how its benefits can be realized in engineering practice. In fact, the concept of what constitutes a 'system' is wide and variable. The purpose of the paper is to draw together diverse perspectives of systems thinking useful in engineering, and to present a set of core concepts that are useful in the successful design and operation of engineered systems. These concepts will be illustrated with examples drawn from the author's experience in teaching and research on engineered systems.

Keywords: Engineered systems; holistic solution; systems thinking.

Introduction

This paper is based on the author's personal experience about systems thinking—learning systems thinking, and using it in consulting and research projects over a span of a quarter of a century. It reflects the experiences gained in teaching systems thinking in modules at the National University of Singapore (NUS) in two engineering departments – the Department of Civil and Environmental Engineering, and Division of Engineering and Technology Management. The modules take the macro perspective of systems, especially the planning, design, and strategy aspects, rather than the analysis and details of particular engineering components or specialized technical disciplines. The author's research focusses on modelling, simulating, and optimizing various kinds of systems in engineering and management, particularly civil engineering systems.

The author serves as the Program Manager for the M.Sc. program in Systems Design and Management, a cross-disciplinary program based on a rational unified process for developing engineering systems. In addition, the author is the co-Director of the NUS-JTC i3c, referred to simply as the 'I-cubed centre' – the 'Innovative Industrial Infrastructure' centre. This is a joint initiative of faculty specialized in master planning, architectural, and engineering design, who collaborate to innovate new solutions and design thinking methods for the industrial city of the future. I3c is where my research and practice come together in real life projects in design studios with engineering and urban planning students.

This paper covers - What is systems thinking; Foundational concepts; Systems engineering; Soft Systems Methodology; Systems management; and Adaptive systems.

What is Systems Thinking

Systems thinking is understood to mean different things by different people. A system could mean a set of physical parts that are part of a bigger whole, e.g. the structural system of a building, or the traction control system of an automobile. We could speak of natural or man-made physical systems. A system could also refer to a set of activities, processes or procedures, e.g. a safety management system, or the human blood circulatory system. It could refer to a group of people, firms or organizations, or more abstract concepts like political, religious or social beliefs, rules and norms held by the people in these groupings. Systems are therefore very pervasive, and the word is used to refer to a set of things or concepts that are related to each other, and which convey a meaning, effect or outcome that could not be achieved by any single part on its own. Systems created by humans are put together to achieve a purpose, whilst the purpose imputed to natural systems serves man's view of the world and his relationship with nature. One aspect of systems thinking is therefore about identifying the parts of a whole, or the factors that are important to an outcome. In this perspective, a systems view aims to be comprehensive, and exhaustive in coverage.

However, systems thinking is also associated with seeing the 'big picture' or macro view of things, as opposed to a more detailed microscopic view. In this sense, systems thinking allows one to comprehend how all the pieces fit together to explain a phenomenon, or how all the parts act to produce the intended effect. It is said that with the ability to 'see the forest from the trees', one is able to solve a problem in a balanced, and holistic way, rather than narrowly focusing only on one aspect of the problem. Holistic solutions, that address several issues simultaneously in an effective way, are said to be preferred to locally optimized fixes.

¹ Department of Civil & Environmental Engineering, National University of Singapore, SINGAPORE. Email: ceecwt@nus.edu.sg

Engineering and allied technical disciplines tend to use the word systems in the first sense – to refer to a collection of parts that have been systematically put together into a purposeful whole. In doing so, the second meaning of systems – as in systemic – that emphasizes the whole rather than the parts, seems to have been forgotten.

Foundational Concepts

The following are some of the key concepts central to systems thinking. Some will be instantly recognizable, yet others will not be so familiar. The implication of adopting these concepts will influence the way systems engineers approach problems, as well as design and implement technical solutions that are required to work in complex environments.

Boundary: The system boundary is a notional line that separates things that are considered part of the system, and everything else which forms its environment. Systems can be open or closed depending on whether there is exchange of material, energy or information across the boundary.

Part-whole structure: The parts of a system can be organized into sub-systems, and these in turn, can be organized into larger sub-systems in a bottom-up fashion until the top-level system is reached. It implies that systems can be decomposed into increasingly smaller and specific parts.

Function and behavior: Man-made, engineered systems are designed for a purpose, which is achieved by incorporating parts and components with the required functions. Functions transform their inputs into outputs, and creates change. The changes that result when functions work together result in the behaviour observed in the system.

Figure 1 shows a ‘white box’ view of an idealized system operating in its environment. The figure emphasizes the boundary separation between the system and its environment, the hierarchical arrangement of the parts of the system, and interactions between the different parts of the system, as well as between the system and its environment. If one obscures all the internal details of the system, and focuses on the interactions between the system and other actors in its environment, we would get the ‘black box’ view of the system.

Non-functional properties: The functional behaviour of a system arises the parts of a system working together. However, the interaction between parts also leads to non-functional system characteristics like safety, reliability, and other so-called system ‘ilities’ which cannot be attributed to the function of any part or component, but arise from the interaction between parts of a system.

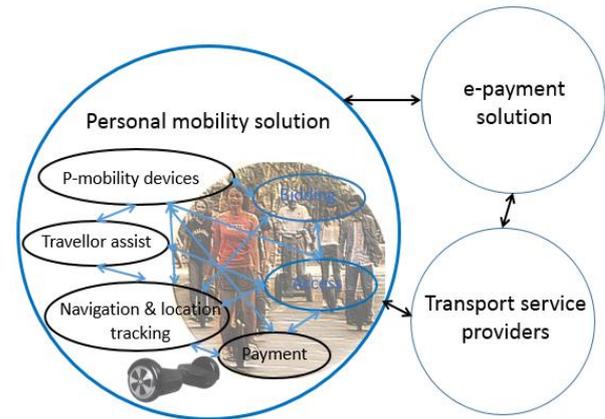


Figure 1. Systems Hierarchy

Determinism: This relates to the relation between cause-and-effect - it says that the cause is sufficient to determine the effect. However, this may be too simplistic except in the case of carefully designed mechanisms since the level of the effect, or its nature, may also depend on other factors, i.e. the cause is necessary, but not sufficient to explain the effect observer. The implication of non-determinism for systems thinking is that, the parts of a system influence each other’s behaviour, and a system’s behaviour is influenced by its environment.

Feedback and system dynamics: Another aspect of system behaviour is the directionality of cause-and-effect. Cause can determine effect, but effect can, in turn, influence the cause subsequently if there is feedback in the system. The existence of positive and negative feedback loops, along with delays in these loops, makes the repertoire of systems more extensive, and more complex than what is available from a collection of simple mechanisms that produce one-way cause-and-effect changes.

Analysis vs synthesis: In systems analysis, the system is broken down into increasing levels of detail, whether it is by function or physical component, and an attempt is made to understand the system by knowing details of the parts. However, the opposite occurs with synthesis – the behaviour of the whole is determined by relating a part to a larger whole, and understanding behaviour in its wider context. Analysis employs reductionist thinking, where complexity is reduced to simple cause-and-effect, whereas synthesis employs expansionist thinking. Analysis accumulates knowledge, but it is synthesis which confers understanding.

Conceptual models: Since everything can be considered a system to be analysed further, or part of a larger system, it can be said that systems thinking is about building conceptual models that explain the complexity of the real world in terms of structure, and behaviour.

Adaptation & learning: Systems thinking wouldn't be complete if it did not provide the means of modelling how people adapt to a changing world, and learn to improve their performance by acquiring and using knowledge from their past actions and experiences. The most important thing about learning is to have a model, whether explicit or implicit, of how change occurs in the world, and to be willing to reconsider our basic assumptions of change, cause-and-effect, in our model.

Systems Thinking in Systems Engineering

Systems engineering: The International Council on Systems Engineering (INCOSE) defines systems engineering as 'an interdisciplinary approach and means to enable the realization of successful systems' [1]. It is a systematic process of realizing technical systems from needs, to requirements, concept, design, and the eventual realized product. This definition envisions the creation of technology-based systems as solutions that fit within larger assemblies of technical systems to accomplish goals and deliver services that could not be done by a single system alone. Society requires systems that are safe, reliable, efficient and cost effective to operate, and which can work as part of larger systems in a trustworthy manner. Technical complexity arises either from the way that system is intended to work in its environment, or the way in which is developed through interdisciplinary collaboration between teams of many different technical specialists.

Systems engineers have a unique role among the many technical disciplines involved in the creation of such complex engineered systems. It is said that the systems engineer is the only role/person in the project team that: (1) places importance on seeing the problem and its solution from a wide variety of useful perspectives; (2) focusses on delivering a well-balanced cost-effective technical solution, rather than delivering technology for its own sake; and (3) bridges the gap between different traditional technical disciplines and specialist disciplines in safety, reliability, constructability etc.

Vee model of systems development: In order to collaborate together in an interdisciplinary way, engineers and specialists working on a system must share a vision of how the system will be developed and coordinated, as well as a means of communicating, concepts, requirements, issues, and design solutions. The Vee model, shown in Figure 2, is one such commonly used model of depicting the steps in systems development; the process is depicted in an idealized way to emphasize particular points like: (a)

orderly hierarchical problem definition and development of design detail; (b) progressive integration of partial solutions into larger wholes, and ultimately the complete system itself; (c) existence of technical reviews to serve as gates between different stages; (d) the intimate relationship between design and testing, verification and validation of the design.

In reality, the systems development effort is more complex, and involves iteration between different stages, as well as recursive execution of more fundamental problem solving steps at different levels of system abstraction and definition.

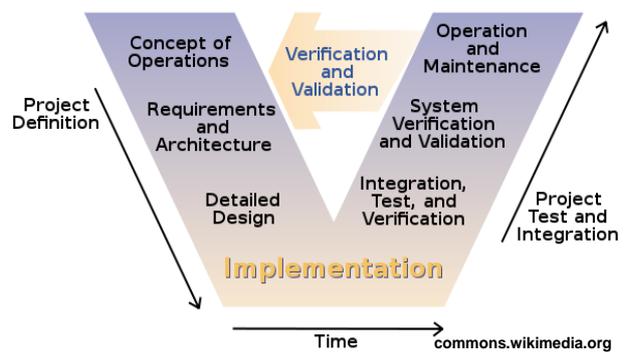


Figure 2. Vee Model of Systems Development

Model-based Systems Engineering (MBSE) is an approach to doing systems engineering using a set of systems concepts for modelling systems. These models enable engineers to communicate and address the different systems-level concerns during the entire development life-cycle. Instead of using information that may not be coordinated across the different project documents, MBSE uses a system model that is described through a set of interrelated diagrams. Changes to the system, facilitated through the view of the system in a particular diagram, can be reliably propagated to related parts of the system, and other diagrams. Systems Modelling Language (SysML) [2], is the specialized language used to do MBSE. With SysML, it will become increasingly possible to support and automate many aspects of the systems engineering effort with some form of intelligent computer assistant.

Soft Systems Methodology

Technology is pervasive in human society, and is incorporated into our human created systems to solve problems and deliver services. Even so, we must not forget that the root of our problems and needs arise from human activities in society—'human activity systems'. Focusing on technology alone, and not getting the purpose, goal and requirements for systems right often leads to short term, ineffective and costly solutions that need to be

reworked. Peter Checkland and co-workers devised Soft Systems Methodology (SSM) [3] as a way to 'inquire about a problematic situation', and to model the way people interact and use technical systems within systems of human activity to accomplish their goals. The model of a socio-technical system includes the key stakeholders, the assumptions and world-view that make the human activity meaningful, and the means by which humans accomplish their goals. This is all captured in the acronym CATWOE, which stands for Customers, Actors, Transformation, World view, Owner, and Environment. Soft systems models emphasize the human dimension of systems, especially human activity, rather than the technological means used in these activities. Unlike many 'hard' systems models seen in Operations Research and Industrial Systems Engineering, soft systems models do not assume deterministic outcomes, nor make any commitment to a particular view of causality in the situation. The emphasis is on building up understanding of increasing parts of the whole from observation, rather than accumulating detailed knowledge about an ever decreasing part of the whole. In effect, SSM focusses on the 'system-in-use' and the hidden assumptions behind it, rather than the system as espoused. System engineers also use SSM to work with stakeholders on the concept of use for technical systems, as well as the key system requirements for a successful solution.

Systems Management

Systems thinking can be employed in the development of management systems, in particular the Safety and Health Management System (SHMS) popular in the construction industry. A SHMS is a collection of processes and activities, instituted by a project organization, for the purpose of assuring the safety and health of the people working on the worksite. Its scope encompasses risk identification; work and methods planning; provision and management of the adequate resources for safe work; training; preparation for handling emergencies; and accident investigation among other things.

Due to the comprehensive nature and broad scope of the activities undertaken by such a system, it is not mandatory for all projects nor can smaller firms implement a system in its entirety for every project. As with many management systems, there are international standards defining the requirements for the documentation and implementation of the processes in the system. Even so, there is no consensus on how to audit the implementation of a safety management system or assess its performance. In Singapore, worksites with a contract sum above S\$30 million must perform regular audits of the SHMS, using a standardized checklist and

scoring scheme called the Construction Safety Audit Scoring System (ConSASS) [4]. The standardized questions and rubrics used in ConSASS improve the reproducibility of audit scores, and help to remove some elements of subjectivity in the assessment. However, real progress in construction safety can only be achieved if the capability of the construction firms, particularly the smaller and medium sized ones, to manage safe worksites is improved in a systematic and systemic way.

Zhang *et al.* [5] takes a process view of a SHMS and proposes a model called ConSASS-2D that is better suited for process and system capability development. The different activities and processes defined in safety standards are organized into key process areas that address different goals and concerns. The processes in each of these areas show increasing levels of capability, depending on the quality of their outcomes they achieve, as well as the features and attributes they possess. The process capability levels are patterned after the Capability Maturity Model (CMM) [6], first developed by the Software Engineering Institute (SEI) at Carnegie-Mellon University for software development but now widely used for other service and process oriented systems as well.

Where ConSASS-2D departs from conventional capability frameworks is to define a second dimension for the development of management systems. The maturity dimension defines how well processes work together to determine the quality of outcomes produced by the system as a whole. ConSASS-2D defines 4 maturity levels for SHMS (defined as performed, managed and quantitatively managed, and optimizing levels) so that firms can progressively develop their systems by adding new processes or increasing the capability of existing processes. In this way, small firms can start out with the core processes defined in the most basic level of system maturity, and develop plans to develop the safety management system to increasing levels of maturity (much like the case in human development). Figure 3 shows the two dimensional framework of ConSASS-2D comprising capability and maturity level definitions.

Adaptive Systems

Intelligent systems learn to adapt their behaviour and improve their performance through experience, which comes from interaction and feedback with the environment. Systems thinking can also be employed in creating intelligent and adaptive systems. One approach is based on evolution in natural systems and is called genetic algorithms (GA) [7]. A population of string-like structures receives feedback from

its environment about the fitness of each individual in the population. At the same time, these individuals undergo selection, and transformations through crossover and mutation. Crossover is the mechanism whereby the good traits of individuals are propagated throughout the population. Selection ensures that fitter individuals have more chance at reproducing themselves, and propagating their traits within the evolving population.

S/No.	Process Areas	CL		
		1	2	3
1.1	OSH Policy			
1.2	Risk management			
1.3	Legal and other requirements Management			
1.4	Responsibility and authority management	ML1		
1.5	Monitoring and control			
1.6	Emergency preparedness and response			
1.7	Accident and incident management			
2.1	Objectives and program(s) management			
2.2	Training		ML2	
2.3	Communication management			
2.4	Management review			
3.1	Audit			
3.2	Performance measurement			ML3
3.3	Document control			
4.1	Performance analysis			
4.2	Causal analysis and resolution	ML4		

Figure 3. Capability-Maturity Dimensions of ConSASS-2D

In systems terms, individuals are composed of genes which express different traits. These traits influence fitness as determined by the environment in which the individuals are placed. Interaction between the environment and the individuals leads to selection pressure, which in turn determines how the string structures evolve to produce fitter individuals. The system therefore comprises a population of string-like structures, the environment that determines fitness, and transformation operations that operate over the population. This generic system can be adapted and mapped to many kinds of planning, scheduling and design problems, thus leading to its popularity as a way to optimize plans, schedules and designs. In Chan *et al.* (1996) [8], each individual in the population pool represents a possible way to schedule the construction activities in a project. The different ordering possibilities determines the resource usage profile as well as the total project duration. Better schedules result in fewer instances of resource usage exceeding resource availability, and shorter project durations.

The GA was again employed to optimize production schedules, but this time in the context of the production of precast members in a factory. The basic model was formulated in Chan *et al.* (2001) [9], and extended to handle more production constraints from the factory floor [10], as well as site coordi-

nation constraints imposed by the project schedule [11,12].

The problem of optimizing a maintenance programme for a road network under budget constraints was considered in Chan *et al.* (1994) [13]. The individuals in the population pool represent different possible maintenance programmes, and each is evaluated for its effect on the performance of the network over a study period. This basic optimization model was extended in different ways. Chan *et al.* (2001) [14] developed a more computational efficient model to handle constraints on desired pavement performance, as well as constraints on equipment availability and budget. Chan *et al.* (2003) [15] formulated a model for optimizing the maintenance budget allocations between different road districts for a pavement network. The issue was the best allocation of limited budget resources given different maintenance priorities among the districts. A search algorithm based on genetic algorithms was used to determine the best allocation. However, this work also marked the beginning of experimentation with using an agent-based approach in the design of optimal plans and schedules.

Intelligent artificial agents are software abstractions of intelligent entities capable of sensing, messaging, reasoning and taking actions in the pursuit of goals. In agent-based models (ABM) [16], the actions and interactions between autonomous agents result in the emergence of system behaviour and properties. Planners and designers are interested in representing problems as agent-based models because of the realism it affords, and the ability to incorporate phenomenon like learning, adaptation, competition and strategy. Policy issues and strategy can be studied by varying the design of the agents, their interactions, or the environment in which they operate. Being computational in nature, ABMs are very adaptable, thus making it possible to address a wide range of problems that would be difficult to formulate and solve using mathematical modelling.

In the Chan *et al.* (2003) article cited above [15], agents were used to represent the different districts, each responsible for a portion of the pavement network. The districts interacted with a central authority on budget proposals as well as overall network performance levels. However, the districts also negotiated between themselves over the use of specialized equipment.

Agents are again used to devise good strategies for a problem involving the determination of competitive pricing by freight forwarders in the logistics domain. The problem is first described and formulated as a multi-level game involving freight forwarders, cargo

owners and shipping carriers [17]. Freight forwarders compete among themselves for the business of cargo owners, and must fulfil their obligations by contracting for cargo space from shipping carriers. The system is modelled from the perspective of the freight forwarder as the ‘man-in-the-middle’ – taking prices from cargo owners, and costs from shipping carriers. Solving the model using game theory gave key insights into how prices were determined by the level of competition among freight forwarders, and the price sensitivity of cargo owners.

However, the game theory formulation imposes restrictions on the number of players in the model, as well as the assumption of full information availability to all players. The effect of learning on prices could not be determined except under final equilibrium conditions. These restrictions were removed when an ABM was created for the same problem situation, and different adaptation strategies were incorporated into the freight forwarder agents based on reinforcement learning [18]. Previously, only the final equilibrium prices could be computed. With the computation ABM model, time trajectories of all the system variables can be recorded and studied. Research in the use of ABM in engineering is still in its early stages but shows a lot of potential for the study of policy, strategic and operational decisions in the area of transportation, safety, and construction management.

Conclusion

Systems thinking is conceptualizing real world phenomenon as models. A model is a set of interrelated concepts, expressed in some language that captures some aspects of reality of interest. In systems thinking, models are characterized by the assumptions made about: (1) purpose; (2) the nature of cause-and-effect; (3) certainty and determinism; (4) inclusion of human and social concerns; and (5) presence of feedback, adaptation and learning.

Systems thinking is now more important than ever in the development of innovative solutions for socio-technical systems. Soft systems methodologies have proven popular to identify stakeholder’s, their roles, capabilities and concerns, elicit requirements, formulate use cases and innovate solutions.

Increasingly, the focus of systems engineering (in the large sense of the word) is on delivering capability for people and society to accomplish its purpose and goals, and not just on delivering technology.

Systems engineers have a variety of models to choose from, and a good systems thinker (systems engineer) will be familiar with several of these. The examples

of systems thinking presented have been those I have used in teaching and research. Knowing which one to use depends on knowledge and experience, and mastering systems thinking is a lifetime’s quest. I wish you all great success in your systems thinking journey!

References

1. INCOSE, *What is Systems Engineering?*, [Online]. Available: <http://www.incose.org/AboutSE/WhatIsSE>. [Accessed September 2015].
2. Wikipedia, *Systems Modeling Language*, [Online]. Available: https://en.wikipedia.org/wiki/Systems_Modeling_Language. [Accessed September 2015].
3. Wikipedia, *Soft Systems Methodology*, [Online]. Available: https://en.wikipedia.org/wiki/Soft_systems_methodology. [Accessed September 2015].
4. Ministry of Manpower Singapore, *Submit a ConSASS Audit*, [Online]. Available: <http://www.mom.gov.sg/workplace-safety-and-health/safety-and-health-management-systems/submit-a-consass-audit>. [Accessed September 2015].
5. Zhang, J. and Chan, W.T., Developing a Construction Safety Management System, in *Modeling Risk Management in Sustainable Construction*, D. D. Wu, Ed., Springer Berlin Heidelberg, 2011, pp. 139-144.
6. Wikipedia, *Capability Maturity Model*, [Online]. Available: https://en.wikipedia.org/wiki/Capability_Maturity_Model. [Accessed September 2015].
7. Wikipedia, *Genetic Algorithm*, [Online]. Available: https://en.wikipedia.org/wiki/Genetic_algorithm. [Accessed September 2015].
8. Chan, W.T., Chua, D., and Kannan, G., Construction Resource Scheduling with Genetic Algorithms, *Journal Construction Engineering Management*, 122(2), 1996, pp. 125–132.
9. Chan, W.T. and Hu, H., An Application of Genetic Algorithms to Precast Production Scheduling, *Computers & Structures*, 79(17), 2001, pp. 1605-1616.
10. Chan, W.T. and Hu, H., Constraint Programming Approach to Precast Production Scheduling, *Journal of Construction Engineering and Management*, 128(6), 2002, pp. 513-521.
11. Chan, W.T. and Zeng, Z., Coordinated Production Scheduling of Prefabricated Building Components, in *Construction Research Congress: Wind of Change: Integration and Innovation*, Honolulu, Hawaii, US, 2003.
12. Chan, W.T. and Zeng, Z., Rescheduling Precast Production with Multiobjective Optimization, in *Computing in Civil Engineering*, Cancun, Mexico, 2005.

13. Chan, W.T., Fwa, T.F. and Tan, C.Y., Road-maintenance Planning using Genetic Algorithms. I: Formulation, *Journal of Transportation Engineering*, 120(5), 1994, pp. 693-709,
14. Chan, W.T., Fwa, T.F., and Hoque, K.Z., Constraint Handling Methods in Pavement Maintenance Programming, *Transportation Research Part C: Emerging Technologies*, 9(3), 2001, pp. 175-190.
15. Chan, W.T., Fwa, T.F. and Tan, J.Y., Optimal Fund-allocation Analysis for Multidistrict Highway Agencies, *Journal of Infrastructure Systems*, 9(4), 2003, pp. 167-175.
16. Wikipedia, *Agent-based Model*, [Online]. Available: https://en.wikipedia.org/wiki/Agent-based_model. [Accessed September 2015].
17. Qin, H. and Chan, W.T., A Game-Theoretic Approach for Freight Forwarders' Pricing Strategy Design, in *Proceedings of the Transportation Research Board 94th Annual Meeting*, Washington DC, United States, 2015.
18. Qin, H. and Chan, W.T., "Freight Forwarder's Pricing Strategy Incorporating Learning from Repeated Transactions," in *Proceedings of the Transportation Research Board 94th Annual Meeting*, Washington DC, United States, 2015.