Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 28 (2014) 449 – 456
Procedia Computer Science

Conference on Systems Engineering Research (CSER 2014)

Eds.: Azad M. Madni, University of Southern California; Barry Boehm, University of Southern California; Michael Sievers, Jet Propulsion Laboratory; Marilee Wheaton, The Aerospace Corporation
Redondo Beach, CA, March 21-22, 2014

Dynamic Planning of System of Systems Architecture Evolution

Zhemei Fang*, Daniel DeLaurentis

Purdue Unicersity, 701 W. Stadium Avenue.,West Lafayette, IN 47907, USA

Abstract

The dynamic planning and development of a large collection of systems or a 'System of Systems' (SoS) pose significant programmatic challenges due to the complex interactions that exist between its constituent systems. Decisions to add, remove, or reconstitute connections between systems can result in repercussive failures across operational and developmental dimensions of an SoS. The work conducted in this research is part of a larger body of work funded by the DoD Systems Engineering Research Center (SERC) towards the development of an Analytic Workbench. This paper in particular develops a tool that adopts an operations research-based perspective to SoS level planning based on metrics of cost, performance, schedule and risk. Specifically, our work employs an Approximate Dynamic Programming approach that is well suited to address issues of computational tractability of the resulting dynamic planning optimization problem. This approach allows for identification of near-optimal multi-stage decisions in evolving SoS architectures. A Naval Warfare Scenario SoS example problem illustrates application of the method.

Keywords : System of Systems Architecture Evolution; Approximate Dynamic Programming; Planning; Uncertainty

1. Introduction

US Department of Defense (DoD) has recognized the prevalence of a 'System of Systems' view to the acquisition and development of military assets[1]; this means that SoS capabilities being sought are a direct consequence of the

* Corresponding author. Tel.: +1-765-398-0739;
E-mail address: fang59@purdue.edu

doi:10.1016/j.procs.2014.03.055

interactive effects of their constituent systems. These constituent systems possess operational and managerial independence, yet interact on various levels to give rise to an overarching SoS level capability. Decisions to support the development of these monolithic entities have required acquisitions using systems engineering based policies that can better account for the complexities associated with SoS architectures. To this end, Defense Acquisition Guidebook (DAG)[2] has been developed to aid the understanding and implementation of DoD acquisition practices including evolutionary acquisition strategies that are the norm for SoS capability evolution. Consistent with DAG, Systems Engineering Guide for Systems of Systems (SoS SE)[1] examines the SoS challenges and provides a 'Trapeze' model to give a good conceptual view of the SoS SE core elements, their interrelationships and SoS decision-making artifacts. Dahmann[3] unwinds the trapeze model to a more familiar and intuitive time-sequenced "Wave" model and identifies information critical to decision making in SoS evolution. Fig.1 illustrates the wave model and original SoS SE core elements.

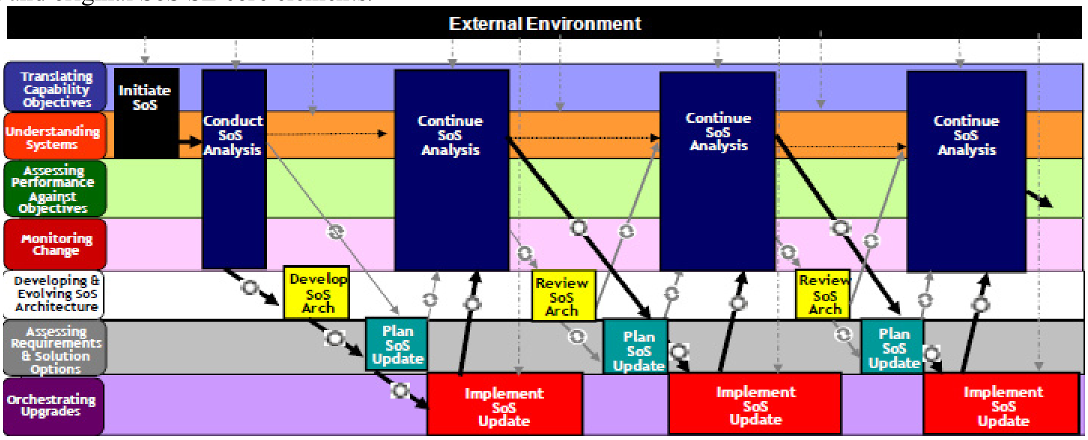

Fig.1. Wave Model and Related SoS SE Core Elements[3]

The systematic procedures as shown in Fig.1 provide guidance to SoS practitioners to make decisions properly; in addition to the logical procedures, an analytic solution framework to objectively quantify the state and outcome of consequent actions to evolve an SoS architecture is required by SoS practitioners. Guided by the wave model, 'Plan SoS Update' and 'Implement SoS Update' are core parts to drive an SoS forward, where key decision points for SoS architecture evolution are mainly located. Accordingly, actions may involve a sequence of decisions that include adding new systems, retiring old systems, upgrading system, etc for 'Plan SoS Update' that could provide policy makers decision sets for achieving optimal or near-optimal SoS capability over a time period. Operational decisions for 'Implement SoS Update' might be integrated meanwhile to provide prompt feedbacks to developers. Unlike traditional production, investment or supply chain planning problem at system level, the dynamic planning in an SoS exhibits a multitude of distinguishing features that need to be carefully addressed. Typical questions could include: how to deal with the interactions between decisions from multiple independent organizations, how to deal with the diverse time scales occurring in an SoS (such as investment decisions every five year versus operational deployment every few months), how to deal with the complexity resulting from the sheer number of uncertain variables involved, and so forth.

The sequential development process and the objective of maximizing the overall capability for a time period makes dynamic programming a natural choice to address the problem. However, the characteristics of an SoS such as the large number of systems that may be involved, multi scale decisions and significant uncertainties lead to state, decision and sample explosion respectively, which challenges the use of dynamic programming. *Approximate Dynamic Programming (ADP)* is an umbrella covering various methods and techniques, aiming to solve the three curses of dimensionality primarily by approximating the future value functions. Thus this paper employs ADP such that the SoS architecture evolution process can be formulated into a dynamic planning problem and meanwhile complexity from multi time scale decisions and uncertainties can be addressed.