Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2015) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

2015 Conference on Systems Engineering Research

Multi-Stakeholder Dynamic Planning of System of Systems Development and Evolution

Zhemei Fang*, Daniel DeLaurentis

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

Abstract

This paper focuses on the master development planning problem in the context of SoS acquisition with resource constraints, uncertainty, and conflicting stakeholder interests. Because most of the master plans are developed by an authority with significant control, which might not be true in a SoS environment. Different stakeholders might have conflicting master plans that require communication, negotiation and coordination. We propose a decentralized multi-stakeholder decision making framework where local stakeholders conduct acquisition development planning for individual benefits while the SoS-level stakeholder designs a coordination mechanism to facilitate the communication between stakeholders and further achieve a harmonious outcome. Specifically, we employ approximate dynamic programming and a transfer contract coordination mechanism to address the problem. We demonstrate the applicability of the proposed approach through a simple illustrative example.

Keywords: System of Systems Development; Decentralized Stakeholders; Approximate Dynamic Programmming; Transfer Contract

1. Introduction

A *System of systems (SoS)*, a set of systems with managerial and operational independence integrating together to pursue unique capabilities[1], brings many challenges to the current acquisition practices. Among the guidelines and procedures to enhance acquisition success[2,3], the “Wave” model[4] developed by Dahmann provides a time-sequenced guideline for practitioners to support SoS architecture development through primary steps of “conduct SoS analysis”, "develop SoS architecture”, “plan SoS update”, and “implement SoS update”. The Wave model is an iterative

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

1877-0509 © 2015 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of Stevens Institute of Technology.

process requiring a series of modeling, simulation, and analysis activities to establish associated SoS *Systems Engineering (SE)* information artifacts in each step. The artifacts are categorized into four groups: SoS capability-related artifacts, SoS technical artifacts, system-related artifacts, and SoS management and planning artifacts. Among these information artifacts, a vital one is the SoS master plan which gives a top-level view across multiple SoS upgrade cycles and describes the long-term SoS acquisition and evolution strategy for SoS capabilities. Successful acquisition needs effective planning. Unfortunately, challenges emerge during the SoS planning process which include: 1) how to address the risks from the significant future uncertainties; 2) how to communicate and coordinate effectively between multiple stakeholders with different interests; 3) how to mitigate the computational complexity from an increasing number of systems and interactions. Some methods are available to tackle these challenges. For example, risk in acquisition is frequently addressed through a standard risk management process from risk identification to mitigation[3]. The acquisition community also employs methods like real options analysis[5] and portfolio analysis[6] to cope with risks. Methods for coordination between different stakeholders are commonly negotiation-based (e.g., Memorandum of Agreement[2]) or standardization-based (e.g., Department of Defense Architecture Framework [DoDAF][7]). However, these methods are not adequate to construct effective plans for SoS development. For instance, the real options analysis suffers from the computational intractability when uncertainties become more complex; the loss of communication efficiency from the back and forth negotiation leads to inappropriate plans. Hence we are motivated to propose a more effective framework for SoS planning.

We describe the master development planning problem in the context of SoS acquisition as, in technical language, a sequential decision making problem with resource constraints, uncertainty, and conflicting stakeholder interests. Fig. 1 shows the fundamental problem setup, illustrating a hierarchy of stakeholders where, at the lowest level, an individual stakeholder (e.g., 'Army' in Fig. 1) controls a set of systems and acquisition decisions for long-term planning. The higher-level stakeholder (e.g., 'DoD' in Fig. 1) exercises certain control over these lower-level counterparts to accomplish required capabilities. Specifically, the higher-level stakeholder provides funding to the lower-level stakeholders with the understanding that the lower-level stakeholders are bounded by funding interdependencies. Christensen[8] defines four types of interdependency between acquisition programs: funding, technological, support, and systems interaction requirements interdependency. We will focus on issues of funding interdependencies and technological interdependencies (upper right of Fig. 1), and particularly funding interdependency in this paper. In this setting, local stakeholders share limited funding resources governed by a SoS-level stakeholder and interactions between systems may or may not exist. Each individual stakeholder observes the shared funding and uses it to develop an optimal portfolio of systems sequentially under uncertainty for its own sake. Meanwhile the SoS-level stakeholder aims to optimize the aggregate capabilities. In other words, the research objective is to formulate and solve SoS level capability development planning problem (under uncertainty) where local stakeholders seek to optimize individual capability by a sequence of decisions.

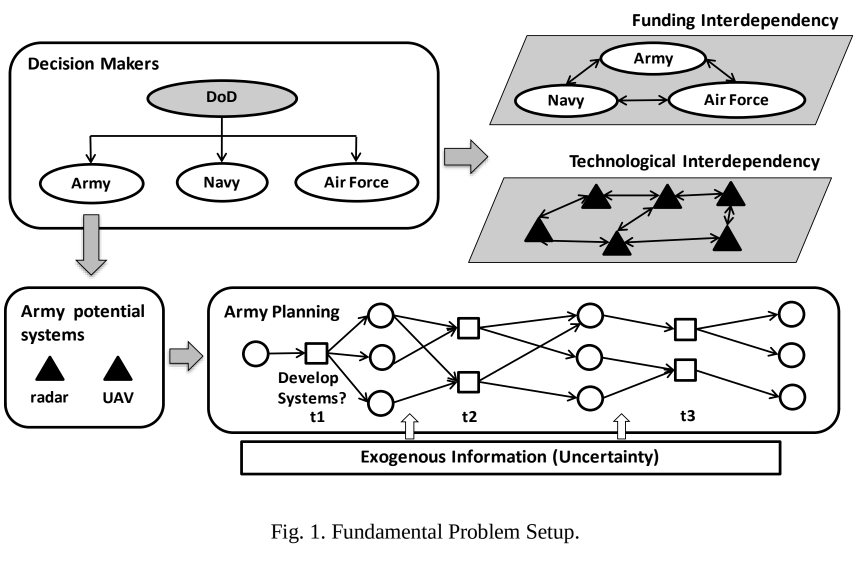

Fig. 1. Fundamental Problem Setup.

2. Related Work

The need to deal with questions of architecture and acquisitions from a SoS perspective has resulted in much research from multiple perspectives. The military acquisition community focused large efforts on developing conceptual and procedural guidance, such as the Defense Acquisition Guidebook[3], the Systems Engineering Guide for SoS[2], the Development Planning Guide[9], the "Wave" model[4] and so on. Other researchers have spent efforts developing analytical approaches to facilitate the conceptual guidance. Available methods include Epoch Era Analysis (EEA)[10,11,12], multi-stage robust portfolio analysis[6,13], Real Options Analysis (ROA)[5,14,15], and method combining agent-based modeling, genetic algorithm and fuzzy evaluation[16,17,18]. Each method distinguishes itself by several advantages; for example, the multi-stage robust portfolio analysis is effective at generating the tradespace between capability and risks that depends on users' choices, while ROA provides a clear flexibility evaluation by inserting different options under uncertainty. However, these methods all suffer more or less from the issues of computational complexity when the target problem becomes complex (i.e., increase of systems, interactions, and uncertainties). Moreover, the decentralized nature of the stakeholders in an SoS (i.e., different stakeholders possess independent management and authority) has not been adequately addressed yet. These two issues are the primary identified gaps of our research.

Many methods are suitable for the long-term sequential decision making. Typical methods are multi-stage stochastic programming and stochastic dynamic programming[19,20,21], which have been widely applied in the area of operations research, energy management, finance management and so forth. However, both of them are confounded by the curses of dimensionality, that is, the explosion of state space, decision space and sample space. To reduce the computational intractability, approximation methods are combined with them under the name of *Approximate Dynamic Programming (ADP)*[22,23]. Thus ADP is quite a flexible modeling and algorithmic framework with various techniques ranging from approximation strategies to stochastic search strategies. By using these techniques ADP excels in converting the intractable problems to tractable problems, which has been proved in real applications of energy dispatch[24], climate policy determination[25] and so on. Based on these advantages of ADP and the need of addressing the potential computational complexity in the SoS development plan, we propose to use ADP to support the long-term decision making.

Coordination mechanisms among decentralized stakeholders have been studied in a variety of fields involving control, operations research and revenue management, among others. Related methods include consensus control[26], distributed optimization[27], game-theory[28], mechanism design[29], and so forth. An important flow of work concerning decentralized stochastic dynamic systems has been researched upon in the airline alliance revenue management[30,31], where itineraries and profits are coordinated between different airlines through transfer price. Transfer contract (price)[32] originally came forth in economics and finance to coordinate between different business units while currently it is also extended to software incremental development[33]. Considering the fact that the acquisition scenario we described shares significant similarities with the airline alliance or multi-business unit firm, we adopt the transfer contract mechanism to deal with the coordination. The noticeable differences such as the decision variables, the uncertainties and benefit evaluation metrics distinguish our work from others.

3. Proposed Approach

To address the development planning problem in SoS acquisition concerning the conflicting stakeholders, uncertainties, and restricted resources, we propose a decentralized planning framework integrating approximate dynamic programming and the transfer contract approach. We introduce these two elements in this section and subsequently build the model formulation under the framework.

3.1. Approximate Dynamic Programming

The beauty of ADP lies in the ability of generating decisions based on an approximation of expected future capability (or reward) value and corresponding updating from resultant new information. ADP alleviates the pain of collecting complete information and building a perfect model for optimizing a complex system of systems. At its core, the power of ADP is derived from the Bellman equation which is also the key driver for dynamic programming in general. Classical dynamic programming recursively computes the Bellman equation below in a backward manner.

$$V_t(S_t) = \max_{x_t \in \mathcal{X}_t}(C_t(S_t, x_t) + \gamma E\{V_{t+1}(S_{t+1})|S_t\}) \quad (1)$$

Where S_t represents state variables, x_t represents decision variables, $C_t(S_t, x_t)$ means contribution earned by taking action x_t while in state S_t at time t, γ is the discount factor and $V_{t+1}(S_{t+1})$ means expected value function of being in state S_{t+1}.

However, the exact future value is extremely difficult to obtain, thus approximated value $\bar{V}_{t+1}$ is usually used to replace V_{t+1}. A variety of approaches are available to approximate the value function, such as linear / piecewise linear approximation, Gaussian network, and so on. A generic structure for the value function approximation is given as follows[23]:

$$\bar{V}_t(S_t) = \sum_{f \in F} \theta_{tf} \varphi_f(S_t) \quad (2)$$

Where $\{\varphi_f(S_t): f \in \mathcal{F}\}$ are referred to as features that describe the important characteristics capturing the total expected capability contribution in the future. θ_{tf} represents adjusting parameters that allow us to obtain different value function approximations.

Another challenge of solving the Bellman equation is calculating the expectation. Powell[23] proposed an important concept of post-decision state variables to separate the effect of decisions and exogenous uncertain information. By using post-decision state variables, the Bellman equation can be written as:

$$V_{t-1}^x(S_{t-1}^x) = E\{\max_{x_t \in \mathcal{X}_t}(C_t(S_t, x_t) + \gamma \bar{V}_t^x(S_t^x)|S_{t-1}^x)\} \quad (3)$$

Where S_{t-1}^x represents post-decision state vector. In this equation, expectation can further be dropped by using a sample realization of the uncertainty $W_t(\omega)$; then the equation turns to the following form:

$$\tilde{V}_{t-1}^x(S_{t-1}^x) = \max_{x_t \in \mathcal{X}_t}(C_t(S_t, x_t) + \gamma \bar{V}_t^x(S_t)|S_{t-1}^x) \quad (4)$$

Given a particular realization of $W_t(\omega)$, the above equation becomes a deterministic optimization problem. Therefore, approximate dynamic programming manifests the advantage by avoiding the explosion of state space, sample space and decision space.

3.2. Transfer Contract Coordination Mechanism

Transfer Contract (Price) was developed as a tool for economics. Many corporations have divisional organizations, in which some or all of the separate divisions are virtually autonomous profit centers to achieve the benefits of decentralization in decision-making[32]. The transfer contract mechanism deals with the problem of pricing the products and services that are exchanged between such divisions within a firm and with the way these prices should be set in order to cause each division to act in such as way that firm profits are maximized as a whole. As illustrated in Fig. 2, transfer price essentially represents the price for the internal market when business unit B needs a product from business unit A. To incentivize the business units to achieve the firm-level best profit, the firm has to determine appropriate transfer price for the internal exchange. A wide variety of transfer pricing methods exist in practice under different contexts today[34].

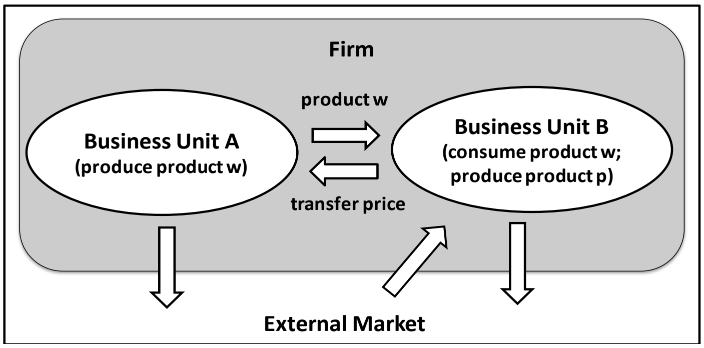

Fig. 2. Original Meaning of Transfer Contract (Price).

With respect to the acquisition field, Army, Navy, and Air Force all receive funding from DoD to develop systems under individual consideration. We define the transfer contract as the compensation that each stakeholder (e.g., Army) needs to pay to other stakeholders (e.g., Air Force) for consuming the shared resources. That is to say, the transfer contract represents the prices of using the shared resources for each stakeholder. And the problem becomes how to choose appropriate transfer contracts to induce the individual stakeholders to make development plans that could benefit the global stakeholder. The value associated with these transfer contracts is not always monetary as in economics. Instead, the transfer contracts in acquisition might represent the information value, partial capability units or a special type of technology.

3.3. Model Formulation

Under the framework of ADP and transfer contract, we build the model formulation in both a centralized and decentralized case, where the centralized case serves as a baseline for comparison. In centralized mode, the global stakeholder has the access to the full information and has the authority to make decisions with the objective of maximizing the overall SoS capability over time. The SoS capability at one time step is assumed as a linear combination of individual system capabilities. In a decentralized mode, each local stakeholder needs to solve an optimization problem. The SoS-level stakeholders do not make decisions directly but they guide the system of systems toward a SoS-level optimal capability by influencing local stakeholders' decisions. The transfer contract coordination mechanism is incorporated in the optimization problem of each individual stakeholder. We list the mathematical formulation in Table 1.

Table 1. Mathematical Formulation under Centralized and Decentralized Management

	Centralized (Baseline)	**Decentralized (stakeholder k)**
Objective Function	Maximize the expected sum of SoS capability over time	Maximize the expected sum of stakeholder k's capability over time with incorporation of transfer contracts
	max $\mathrm{E}\{\sum_{t=0}^{T} c\_l_t R_t\}$	max $\mathrm{E}\{\sum_{t=0}^{T}[c\_l_t^k R_t^k + T\_M_t^k(other\ stakeholders)]\}$
Budget Limits	$c\_s_t x_t \leq b_t$	$c\_s_t^k x_t^k + E\_M_t^k(other\ stakeholders) \leq b_t$
Integer Decisions	$x_t \geq \mathbf{0},\ x_t \in Z$	$x_t^k \geq \mathbf{0},\ x_t^k \in Z$
Transition Function	$R_{t+1} = R_t + x_t$	$R_{t+1}^k = R_t^k + x_t^k$
	$c\_l_{t+1} = c\_l_t + \widehat{c\_l}_{t+1}$	$c\_l_{t+1}^k = c\_l_t^k + \widehat{c\_l}_{t+1}^k$

The left column of Table 1 describes the centralized case and the right column describes the decentralized case. c_l_t denotes the vector of individual system capabilities, c_s_t denotes the system cost. R_t represents the state variable or vector of the number of different systems, x_t represents the decisions variables or vector of the number of systems that are added at each time step. We bound the objective function by budget constraint b_t at each time step. The transition function demonstrates the evolution of state variable R_t by acting the decisions, and the evolution of c_l_t by adding the uncertain change $\widehat{c\_l}_{t+1}$ from the environment. Note that in decentralized mode, each local stakeholder k needs to solve the optimization problem where $T\_M_t^k$ represents transfer contract mechanism and $E\_M_t^k$ delineates the estimated cost and decisions of other stakeholders. These equations are further translated to ADP structure to be solved.

4. Preliminary Results

In this section, we solve a synthetic example to explain the proposed approach and demonstrate the feasibility of finding good solutions. We assume that 1) a SoS-level stakeholder requires a type of capability (e.g. conduct terrestrial and maritime surface surveillance) to be accomplished over ten years; 2) each of the three individual stakeholders A, B, C (e.g., Army, Navy, Air Force) has two systems in the option pool. Under the assumptions, we recap our ultimate goal: individual stakeholder makes its own optimal development decisions (e.g., Army targets at maximizing the terrestrial surface surveillance) while these decisions can reach the aggregate optimal capability for the SoS-level stakeholder (e.g., maximizing the overall terrestrial and maritime surface surveillance) by choosing an appropriate set of transfer contracts (e.g., exchange of information value, converted monetary value of capabilities, etc). We intend to demonstrate the applicability of approximate dynamic programming and transfer contract coordination mechanism by using notional data.

4.1. Centralized Case – The Applicability of Approximate Dynamic Programming

We demonstrate the ability of ADP by comparing with other methods in a centralized mode where a virtual global stakeholder (without stakeholder A, B, C) possesses all the information. We start with the deterministic capability and cost of single systems, basing on which we solve the equations on the left column of Table 1. In this case, the SoS-level stakeholder optimizes the capability over ten years with decisions of whether developing systems {S1a, S2a, S1b, S2b, S1c, S2c} at each year. The individual system capability, cost and annual budget limit are given as notional numbers. We list the numbers in Table 2 and Table 3.

Table 2. Input of capability and cost of individual systems.

Stakeholder	Stakeholder A		Stakeholder B		Stakeholder C	
Systems	S1a	S2a	S1b	S2b	S1c	S2c
Capability	50	40	30	20	20	20
Cost	80	70	60	50	40	20

Table 3. Input of budget at each time step.

Time Step	t = 0	t = 1	t = 2	t = 3	t = 4	t = 5	t = 6	t = 7	t = 8	t = 9
Budget	220	250	180	200	230	170	200	280	120	240

This is a simple multi-stage problem that is solvable through a set of single-stage optimizations, by which we obtain the exact solution as a benchmark. Since we set the decision variables as binary variables (acquire or not acquire), we employ linear value function approximation to maintain simplicity while not losing much accuracy. We express our value function approximation as the linear combination of basis functions with the coefficients named adjusting parameters. The basis functions usually refer to the key features characterizing the future capabilities. Commonly, we choose the basis functions from the state variables. It is important to identify the appropriate basis functions, because they determine the accuracy of the results. The primary state variables in this case are the number of different systems at each stage, capability and cost of individual systems. We use all of them as our basis functions for a start. With the basis functions available, we employ a recursive least squares method to update the adjusting parameters in the value function approximation. The results are demonstrated in Fig. 3.

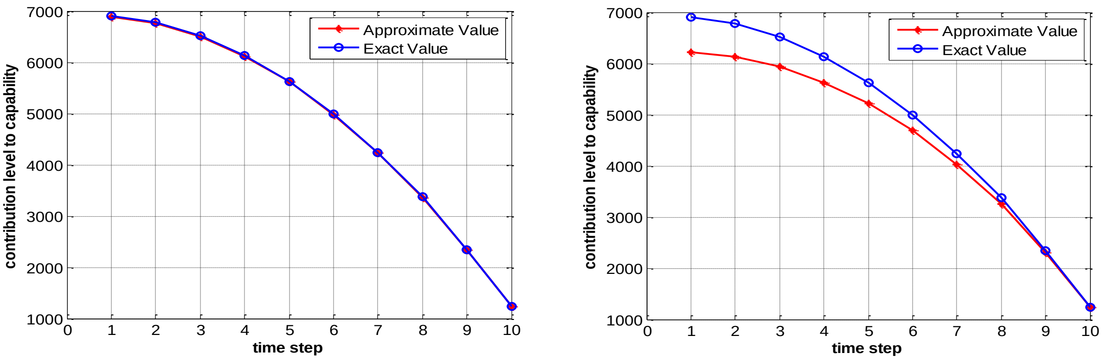

Fig.3. (a) comparison of exact and approximate capability; (b) comparison of exact and approximate capability with reduced basis function.

As shown in Fig. 3(a), the objective capability obtained from ADP is very close to the exact objective capability; through calculation the approximate value is only 0.28% lower than the exact objective value and converges within 20 iterations. Often times, to reduce the computational complexity, only key state variables are selected to represent the basis functions, though this may reduce the accuracy of the approximate result. For example, if we choose basis functions as the vector of number of systems and a constant value, the accuracy of the ADP objective becomes 10% lower than the exact optimal objective (at the first stage), as illustrated in Fig. 3(b). We can also observe that as time progresses, the approximate value converges to the exact value. This is because the basis functions are less adequate to capture the future capabilities over ten stages than over one stage.

Next, we add uncertainty into the problem to validate the ADP implementation in a stochastic environment. We begin with the uncertainty of individual system capability (as shown in Table 1). It is reasonable to hypothesize that the variations of individual system capability occur at each stage due to exogenous information fluctuation such as the technology development or national priority change. In order to make the problem solvable by other methods, we give a fixed set of the individual system capability and the variation appears at each stage independently. We assume the variation as a uniform distribution with bounds [-2, 2] according to subjective judgment.

We still use the basis functions including all the state variables to implement the ADP algorithm. Through running 200 iterations we obtain an acceptable value function approximation, which we use to calculate the capability and decisions at each stage given a certain sample realization. We employ a posterior bound solution as a benchmark to compare with the results from ADP. Given a specific sample realization of the system capability variations, the posterior bound solution "cheats" by being able to use information only known in the future and solves the problem with the same techniques in the deterministic case. Since the uncertainty is quite simple, we also use classical backward recursive dynamic programming to conduct the analysis, which solves the problem through backward induction and expectation calculation. We observe some limitations of backward dynamic programming when doing the computation. For example, regardless of the large number of loops over the state variables to compute the future value expectation, it is almost impossible to calculate the expectation of the decision-dependent stochastic variables.

The results when given a specific sample realization are shown in Fig. 4. Fig. 4(a) shows a comparison between the capability values from ADP, dynamic programming, and posterior bound solutions at each stage. It is apparent that the ADP solutions are quite acceptable. Specifically, Fig. 4(b) points out the iteration process of the objective value from ADP (value at first stage). We see that the curve does fluctuate but is stable, since fluctuation range is within 10% of the posterior optimum.

Returning to the real world acquisition decision making process, data from the uncertainties tends to be exposed gradually, thus the actual ADP implementation for decision makers is 1) estimating the probability distribution of future uncertainties or building simulation models, 2) constructing the approximation of future value function based on characteristics of state variables, 3) using the approximate value function to compute the current decisions, 4) implementing the decisions in simulation model or real world with new data coming out to update the uncertainty and value function approximation, 5) going back to step 3). Therefore, the implementation of ADP requires training data to obtain reasonable value function approximation.

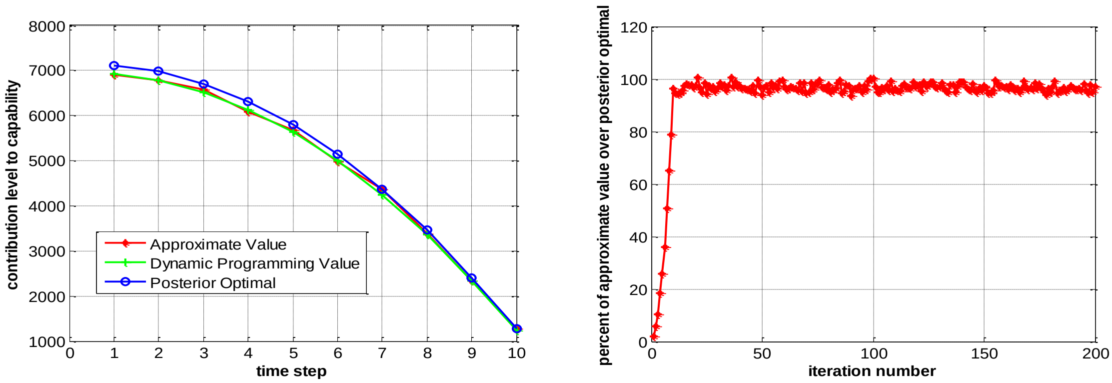

Fig. 4. (a) validation of ADP results under uncertainty; (b) percent of approximate objective over posterior optimal objective.

This simple example only demonstrates us the ability of ADP to generate reasonable results compared to exact solutions. In fact, the advantages of ADP reveal more when the problems become more complex. For example, it avoids the explosion of state space, decision space and sample space, and it can explicitly describe the decision-dependent uncertainties, and it can apply to scenarios where uncertainties cannot be expressed as probability distributions. We will explore more of these in the future work and next section we will investigate the effect of transfer contract coordination mechanism.

4.2. Decentralized Case – The Effect of Transfer Contracts

This section explores the effect of transfer contracts in multi-stakeholder planning under decentralization. We begin with re-stating the process in the decentralized scenario. The SoS-level stakeholder firstly announces the budget limits of each year as well as the transfer contract coordination scheme to the individual stakeholders. Then, under the budget limits, each component stakeholder proposes the potential systems to the SoS-level stakeholder and other peer stakeholders. Notwithstanding, the individual stakeholders tend to keep information of their own system capability and cost to themselves or only partially reveal the information to other stakeholders in order to, for example, attempt to gain the more funding or not lose funding. Therefore, when each component stakeholder aims to solve the optimal long-term development planning problem, it needs to estimate some values (e.g., system cost) of other stakeholders based on the information it already possesses. It is possible that the optimal decisions from each individual stakeholder may conflict with each other and conflict with the optimal decisions at the SoS-level. Thus we employ the transfer contract mechanism to coordinate between the individual stakeholders to generate global optimal solutions while incentivizing the local stakeholders.

According to the assumption that stakeholder A knows potential systems under stakeholder B and C but does not know the exact system cost and associated decisions, stakeholder A estimates the value of cost and decisions of other stakeholders. We denote the estimated costs as $\widetilde{c\_s}_t^{AB}$, $\widetilde{c\_s}_t^{AC}$, $\widetilde{c\_s}_t^{BA}$, $\widetilde{c\_s}_t^{BC}$, $\widetilde{c\_s}_t^{CA}$, $\widetilde{c\_s}_t^{CB}$ and the decision variables as $\tilde{x}_t^{AB}$, $\tilde{x}_t^{AC}$, $\tilde{x}_t^{BA}$, $\tilde{x}_t^{BC}$, $\tilde{x}_t^{CA}$, $\tilde{x}_t^{CB}$, where the first letter in the superscript represents the stakeholder who makes the estimation and the latter letter represents the stakeholder being estimated. We add the set of transfer contracts in the objective function of each stakeholder. Again the transfer contract represents the compensation (transferrable utility) that one stakeholder needs to pay to other stakeholders. The impact of transfer contract is to modify the objective function and influence the decisions of each component stakeholders. We denote the transfer contracts as $t\_c_t^{AB}$, $t\_c_t^{AC}$, $t\_c_t^{BA}$, $t\_c_t^{BC}$, $t\_c_t^{CA}$, $t\_c_t^{CB}$ where the superscript 'AB' means transferring from A to B. We formulate the equation in Table 4.

Table 4. Formulation with transfer contract coordination

Stakeholder A	Stakeholder B	Stakeholder C
Objective: max $\sum_{t=0}^{10}[c\_l_t^A R_t^A - (t\_c_t^{AB} + t\_c_t^{AC})x_t^A + t\_c_t^{BA}\tilde{x}_t^{AB} + t\_c_t^{CA}\tilde{x}_t^{AC}]$ Subject To: $c\_s_t^A x_t^A + \widetilde{c\_s}_t^{AB}\tilde{x}_t^{AB} + \widetilde{c\_s}_t^{AC}\tilde{x}_t^{AC} \le b_t$ $R_{t+1}^A = R_t^A + x_t^A$	Objective: max $\sum_{t=0}^{10}[c\_l_t^B R_t^B - (t\_c_t^{BA} + t\_c_t^{BC})x_t^B + t\_c_t^{AB}\tilde{x}_t^{BA} + t\_c_t^{CB}\tilde{x}_t^{BC}]$ Subject To: $c\_s_t^B x_t^B + \widetilde{c\_s}_t^{BA}\tilde{x}_t^{BA} + \widetilde{c\_s}_t^{BC}\tilde{x}_t^{BC} \le b_t$ $R_{t+1}^B = R_t^B + x_t^B$	Objective: max $\sum_{t=0}^{10}[c\_l_t^C R_t^C - (t\_c_t^{CA} + t\_c_t^{CB})x_t^C + t\_c_t^{AC}\tilde{x}_t^{CA} + t\_c_t^{BC}\tilde{x}_t^{CB}]$ Subject To: $c\_s_t^C x_t^C + \widetilde{c\_s}_t^{CA}\tilde{x}_t^{CA} + \widetilde{c\_s}_t^{CB}\tilde{x}_t^{CB} \le b_t$ $R_{t+1}^C = R_t^C + x_t^C$

We start with the decentralized decision making without any coordination (i.e., remove the transfer contract terms in the objective functions), and the results are illustrated in Fig. 5. Fig. 5(a) shows the capability of the global stakeholder and individual stakeholders in the decentralized case without any coordination. Fig. 5(b) illustrates the budget constraint violations due to the inaccuracy of model estimates and lack of coordination. In general, without coordination the stakeholders tend to have budget requests that diverge greatly from the budget constraints. In this case, when individual stakeholders submit their development plans, conflict occurs due to the limited available funding. The stakeholders need to negotiate to arrive at a balanced solution. Unfortunately, as stated previously, back and forth negotiation largely decreases the communication efficiency and probably reduces the quality of the solution. Therefore, we seek for mechanism that could guide the individual stakeholders to coordinate between themselves in a more automatic manner.

We next examine the effect of transfer contracts on the capability and budget violation by giving a set of transfer contracts with arbitrary numbers. Fig. 6 displays the capability value and budget violation at each stage. This set of transfer contracts still cannot lead to SoS-level optimality within constraints, but through the coordination, we can find the capability change at each stage of different stakeholders and corresponding budget violation change. By comparing the left plots of Fig. 5 and Fig. 6, we find the capability reduction for SoS-level stakeholder when transfer contracts are incorporated. However, the right plot in Fig. 6 demonstrates that the budget violation becomes much less with the inclusion of transfer contracts. Although we have only employed an arbitrary set of transfer contracts so far, we find the large chance to achieve valuable results.

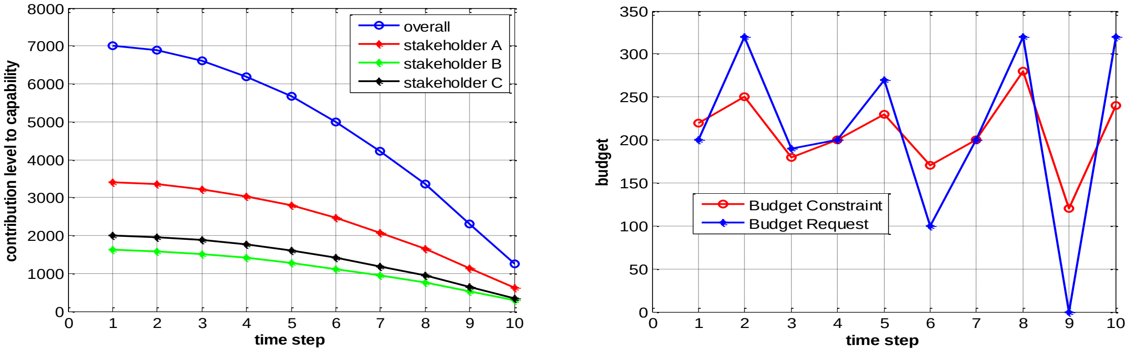

Fig. 5. (a) capability in decentralized setting without coordination; (b) budget violation without coordination.

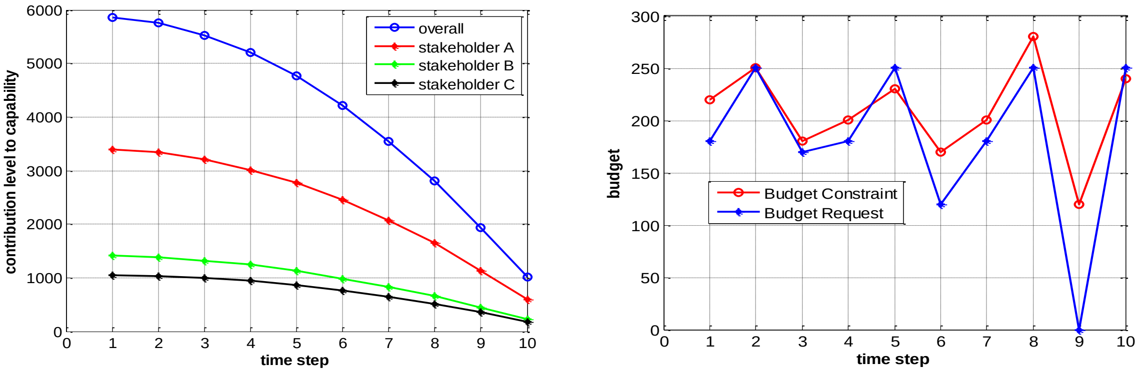

Fig. 6. (a) capability in decentralized setting with given transfer contracts; (b) budget violation with given transfer contracts.

5. Conclusion and Future Work

This paper proposed a decentralized planning framework for SoS architecture analysis in acquisition context and demonstrated its applicability through a simplified example. The framework contributes to the current acquisition practices by addressing the decentralized natures of stakeholders and the potentials of solving complex problems. In addition to the high level (e.g., DoD, Army, etc) scenario described in this paper, the approach can potentially apply to the program management level as well (e.g., contracting strategies) when independent stakeholders (e.g., aerospace companies) involved are bounded by the shared resources. Near term work is focusing on characterizing the transfer contracts that lead to SoS-level optimality when individual stakeholders are making sequential decisions. Currently we conjecture that the transfer contract received by an individual stakeholder equals the approximated future value for this stakeholder losing the resources consumed by another stakeholder. Further work will concentrate on the practical application of the framework in acquisition field and extend to commercial perspective.

Acknowledgements

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract H98230-08-D-0171. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

References

1. Maier M. Architecting Principles for System-of-Systems. Systems Engineering. 1998; 1(4): p. 267-284.
2. OUSD(AT&L). Systems Engineering Guide for System of Systems. Washington, D.C.: Pentagon:; 2008.

3. Defense Acquisition Guidebook. ; 2013.
4. Dahmann J, Rebovich G, Lowry R, etc. An Implementers' View of Systems Engineering for Systems of Systems. In ; 2010. p. 1-6.
5. Mun J. Capturing the Strategic Flexibility of Investment Decisions Through Real Options Analysis. In The Strategic CFO.; 2012. p. 69-84.
6. Davendralingam N, Mane M, DeLaurentis D. Capability and Development Risk Management in System-of-Systems Architectures: A Portfolio Approach to Decision-Making. In Proceedings of Ninth Annual Acquisition Research Symposium; 2012.
7. DOD Architecture Framework Version 2.0. ; 2009.
8. Christensen MB. A Method For Measuring Programmatic Dependency and Interdependency Between DoD Acquisition Programs. ; 2011.
9. USAF. Development Planning (DP) Guide. ; 2010.
10. M.Ross A, H.Rhodes D. Using Natural Value-Centric Time Scales for Conceptualizing System Timelines through Epoch-Era Analysis. In INCOSE; 2008.
11. A.Schaffner M, Wu M, M.Ross A, H.Rhodes D. Enabling Design for Affordability: An Epoch-Era Analysis Approach. In Proceedings of the Tenth Annual Acquisition Research Symposium Acquisition Management; 2013.
12. Chattopadhyay D, M.Ross A, H.Rhodes D. Combining Attributes for Systems of Systems in Multi-Attribute Tradespace Exploration. In Conference on Systems Engineering Research 2009 (CSER); 2009.
13. Davendralingam N, DeLaurentis D. Acquisition Management for System of Systems: Affordability through Effective Portfolio Management. In Proceedings of the Tenth Annual Acquisition Research Symposium; 2013.
14. de Neufville R. Real Options: Dealing with Uncertainty in Systems Planning and Design. Integrated Assessment. 2003; 4(1): p. 26-34.
15. Buurman J, Zhang S, Babovic V. Reducing Risk Through Real Options in Systems Design: The Case of Architecting a Maritime Domain Protection System. Risk Analysis. 2009; 29(3): p. 366-379.
16. Acheson P, Pape L, Dagli C, Kilicay-Ergin N, Columbi J, Haris K. Understanding System of Systems Development using an Agent-Based Wave Model. In Complex Adaptive Systems; 2012.
17. Ergin N, Acheson P, Colombi J, Dagli C. Modeling System of Systems Acquisition. In ; 2011.
18. Acheson P, Dagli C, Kilicay-Ergin N. Fuzzy Decision Analysis in Negotiation between the System of Systems Agent and the System Agent in an Agent-Based Model. In The Proceeding of International Conference on Soft Computing and Software Engineering; 2013.
19. Defourny B, Ernst D, Wehenkel L. Multistage Stochastic Programming: A Scenario Tree Based Approach to Planning under Uncertainty. In.; 2009.
20. Dupacova J, Sladky K. Comparison of Multistage Stochastic Programs with Recourse and Stochastic Dynamic Programs with Discrete Time. ZAMM.Z.angew.Math.Mech. 2001;: p. 1-15.
21. B.Powell W. A Unified Framework for Stochastic and Dynamic Programming. Informs Computing Society Newsletter. 2012.
22. Bertsekas D, Tsitsikilis J. Neuro-Dynamic Programming: Athena Scientific, Belmont, MA; 1996.
23. B.Powell W. Approximate Dynamic Programming: Solving the Curses of Dimensionality. 2nd ed.; 2010.
24. B.Powell W, George A, Simao H, Scott W, Lamont A, Stewart J. SMART: A Stochastic Multiscale Model for the Analysis of Energy Resources, Technology, and Policy. INFORMS Journal on Computing. 2011;: p. 1-18.
25. Webster M, Santen N, Parpas P. An Approximate Dynamic Programming Framework for Modeling Global Climate Policy under Decision-Dependent Uncertainty. ; 2011.
26. Olfati-Saber R, Fax JA, M.Murray R. Consensus and Cooperation in Networked Multi-Agent Systems. In Proceedings of IEEE; 2007.
27. L.Raffard R, J.Tomlin C, P.Boyd S. Distributed Optimization for Cooperative Agents: Application to Formation Flight. In IEEE Conference on Decision and Control; 2004.
28. Drechsel J. Cooperative Lot Sizing Games in Supply Chains; 2010.
29. Mour A, DeLaurentis D. Bandwidth Allocation in Tactical Data Links via Mechanism Design. In Conference on Systems Engineering Research (CSER); 2014.
30. P.Wright C, Groenevelt H, A.Shumsky R. Dynamic Revenue Management in Airline Alliances. Transportation Science. 2009; 44(1): p. 15-37.
31. Cai H, E.B.Lim A. Decentralized Control of A Multi-Agent Stochastic Dynamic Resource Allocation Problem. In The 50th IEEE Conference on Decision and Control and European Control Conference; 2011.
32. Hirshleifer J. On the Economics of Transfer Pricing. The Journal of Business. 1956; 29(3): p. 172-184.
33. Ceran Y, Dawande M, Liu D, Mookerjee V. Optimal Software Reuse in Incremental Software Development: A Transfer Pricing Approach. Management Science. 2014; 60(3): p. 541-559.
34. Toktay LB, Wei D. Cost Allocation in Manufacturing-Remanufacturing Operations. Production and Operations Management. 2011; 20(6): p. 841-847.