RC24678 (W0810-120) October 21, 2008
Computer Science

IBM Research Report

An Optimal-Control Based Decision-Making Model and Consulting Methodology for Services Enterprises

Liang-Jie Zhang, Zhe Shan
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598
(zhanglj@us.ibm.com, zshan@us.ibm.com)

Zhi-Hong Mao
Department of Electrical and Computer Engineering
and
Department of Bioengineering
University of Pittsburgh
Pittsburgh, PA 15261
(maozh@engr.pitt.edu)

This work was supported in part by the NSF under Grant CMMI-0727256.

Research Division
Almaden - Austin - Beijing - Cambridge - Haifa - India - T. J. Watson - Tokyo - Zurich

Abstract

Services industry has become a primary growth point in most countries. However, the current services business is largely constrained by human factors and lacks automated and quantitative techniques for operation and decision-making. In this paper, we propose an optimal-control based decision-making model that facilitates performance analysis and strategy planning for services enterprises. Using approximate dynamic programming, we are able to handle the model complexity and obtain a near-optimal solution for the decision making of employee management, advertisement, and asset investment. We validate our model and approach in a business simulation practice. Furthermore, we propose a consulting methodology based on the optimal-control model and describe its application in services consulting practices.

Index Terms

Approximate dynamic programming, consulting methodology, decision-making, optimal control, services enterprises.

I. INTRODUCTION

Services industry has been a fast-growing industry in many countries. In the United States, for example, the services industry accounts for 55% of the total economic activity in 2006 [1], and contributes about 78.7% of the GDP in 2007 [2]. However, most services businesses today are labor-intensive and rely on skilled workers to deliver services to consumers. Investment decisions and service deliveries are still heavily subject to personal knowledge, expertise, and biases. So as to improve productivity of services business, analytical and quantitative methods are desirable that enable services enterprises to deliver cost-effective services.

Specifically, there has been a substantial need for formalized ways to represent services enterprises and facilitate decision-making in services enterprises. Business architecture has been introduced to componentize an enterprise based on business functionalities and time-varying requirements from the market [3], [4]. However, most work in this direction only provides text-rich descriptions and suggestions. It is still a challenging issue to formalize quantitatively a way of guiding effective decision-making in services enterprises.

In this paper, we aim to formalize a unified decision-making model for services enterprises based on the optimal control theory. Optimal control has been widely used in economics [5].

Dorfman illustrated in [5] that optimal control is formally identical to capital theory, and control-theoretic frameworks can be developed to represent and analyze the generic economic ecosystems. Following that idea, we apply optimal control to investigate a specific type of services enterprises that are project-based and manpower-centric business systems.

The rest of this paper is organized as follows: Section II presents an optimal-control model for decision-making in services enterprises. The model captures major behaviors of a services enterprise. Section III utilizes dynamic programming to solve the optimal control problem, and develops a variant of the forward induction algorithm using cubic spline interpolation to construct value functions. Section IV illustrates the capability of the optimal-control model via a case study. Section V introduces a consulting methodology that helps decision makers or consultants to make investment and planning decisions based on the optimal-control based decision-making model. Section VI presents discussions on several related issues and ongoing efforts. The last section concludes the paper.

II. Decision-Making Model for Services Enterprises (DMM4SE)

In this section, we model the decision-making of a services enterprise based on optimal control. Our objective is to help business managers or consultants to quantify their available options and make decisions in an effective and managed fashion.

Over a certain period of time, say from period 1 to K, in a services enterprise, the decision makers need to choose a sequence of actions so as to maximize some "reward" or optimize the enterprise performance with respect to some predetermined criterion. Without loss of generality, the time interval is set to a year in this paper, but it can be chosen based on specific management requirement, e.g., a month or a quarter. At time $k \in [1, K]$, the enterprise inherits a certain asset and other conditions from its previous period. These can be described by a set of variables, and these variables correspond to a system's *state variables*, or simply *states*. Denote the state variables at time k by a vector $x(k)$. Correspondingly, the initial state is denoted as $x(0)$ or x_0, and the terminal state is $x(K)$. With this state x and at that particular time k, the enterprise takes some decisions concerning investment for advertisement and policies for hiring, training, and laying off employees, etc. Denote the decisions taken at any time k by a vector $u(k)$. This decision vector u corresponds to the input to the dynamic system modeling the operation of the enterprise. From the state at a specified time together with the specified current decisions, the enterprise derives a certain reward:

$$r(x(k), u(k)).$$

This reward determines the benefit earned at time k as a result of having the state of $x(k)$ and taking decisions $u(k)$.

Note that the decisions taken at any time influence not only the rewards earned at that time but also the system state. The current state of a services enterprise can be expressed as a function of the previous state and current decisions:

$$x(k) = f(x(k-1), u(k)). \tag{1}$$

Given the initial state x_0, the total rewards that will be earned from the initial time ($k = 1$) to the terminal time K is given by

$$V_K(x_0, \mathbf{u}) = \sum_{k=1}^{K} r(x(k), u(k)) \tag{2}$$

where $\mathbf{u}$ denote the entire time path of the decision variable u from the initial time to K.

The above formulas express the essence of the problem of making decisions in a dynamic context. The problem is to select the time path $\mathbf{u}$ so as to make the total reward V_K as large as possible. This problem is exactly an optimal control problem. In the following subsections, we will introduce the details of this model.

A. Key Components of DMM4SE

Most of the enterprises today are project-based businesses. The projects can be internal projects on basic research, product development, communication, IT infrastructure transformation, and business process reengineering, and can also be external projects including, for example, IT outsourcing and on-site service delivery. Most of the revenues of service providers come from service projects. Without loss of generality, we assume that the annual gross income of a services enterprise under consideration is contributed from the projects (or contracts) that the enterprise receives within a year.

To describe the dynamics of the operation and decision-making in a services enterprise, we introduce the following notation:

- *Indices:*

- k: year number;
- j: year of experience of an employee.

- *Parameters:*
  - K: maximum decision period;
  - $n_{\max}$: maximum years of experience beyond which an employee does not stay;
  - $w_{\mathrm{H}}(j)$: cost of hiring an employee with j years of experience;
  - $w_{\mathrm{L}}(j)$: layoff cost for an employee with j years of experience;
  - w_{R}: allowance paid to a retiring employee;
  - $w_{\mathrm{S}}(j)$: annual salary paid to an employee with j years of experience;
  - $\delta_{\mathrm{S}}(c, j)$: average incremental strength of an employee with j years of experience after receiving a training with cost c per person.
- *Decision (or input) variables:*
  - $u_{\mathrm{A}}(k)$: advertising expenses in year k;
  - $u_{\mathrm{LE}}(k, j)$: in year k, the number of employees that are laid off with j years of experiences ($1 \leq j \leq n_{\max} - 1$);
  - $u_{\mathrm{NE}}(k, j)$: in year k, the number of new employees that are hired with j years of experience ($0 \leq j \leq n_{\max} - 1$);
  - $u_{\mathrm{P}}(k)$: project price offered by the enterprise in year k;
  - $u_{\mathrm{T}}(k, j)$: in year k, the expense that the enterprise spends in training employees with j years of experience ($0 \leq j \leq n_{\max} - 1$);
  - $u_{\mathrm{Z}}(k)$: asset investment in year k.
- *State variables:*
  - $x_{\mathrm{A}}(k)$: advertising capital in year k, which summarizes the effects of current and past advertising investment;
  - $x_{\mathrm{E}}(k, j)$: in year k, the number of employees that are kept with j years of experience ($0 \leq j \leq n_{\max}$);
  - $x_{\mathrm{S}}(k, j)$: average strength of an employee with j years of experience in year k ($0 \leq j \leq n_{\max}$);
  - $x_{\mathrm{Z}}(k)$: asset of the enterprise in the year k.
- *Output variables:*

- $y_{\mathrm{O}}(k)$: operational cost in year k;
- $y_{\mathrm{P}}(k)$: number of projects received by the enterprise in year k.

Furthermore, the state vector $x(k)$ and decision vector $u(k)$ are defined as

$$x(k) = (x_{\mathrm{A}}(k), x_{\mathrm{E}}(k,0), ..., x_{\mathrm{E}}(k,n_{\max}), x_{\mathrm{S}}(k,0), ..., x_{\mathrm{S}}(k,n_{\max}), x_{\mathrm{Z}}(k))' \tag{3}$$

$$\begin{aligned} u(k) = & (u_{\mathrm{A}}(k), u_{\mathrm{LE}}(k,0), ..., u_{\mathrm{LE}}(k,n_{\max}-1), u_{\mathrm{NE}}(k,1), ..., u_{\mathrm{NE}}(k,n_{\max}-1), \\ & u_{\mathrm{P}}(k), u_{\mathrm{T}}(k,0), ..., u_{\mathrm{T}}(k,n_{\max}-1), u_{\mathrm{Z}}(k))'. \end{aligned} \tag{4}$$

In the following subsections, we will introduce the submodels characterizing the dynamics of the state variables.

B. Advertising Capital

We follow the model by Nerlove and Arrow [6] to describe the dynamics of advertising capital (or called market goodwill). We assume that the advertising capital depreciates over time at a constant proportional rate δ_{A}. Then we have

$$x_{\mathrm{A}}(k) = (1-\delta_{\mathrm{A}})x_{\mathrm{A}}(k-1) + u_{\mathrm{A}}(k). \tag{5}$$

The above equation can also be written as

$$x_{\mathrm{A}}(k) - x_{\mathrm{A}}(k-1) = u_{\mathrm{A}}(k) - \delta_{\mathrm{A}}x_{\mathrm{A}}(k-1)$$

which implies that the net investment in advertising is the difference between gross investment $u_{\mathrm{A}}(k)$ and depreciation [7].

C. Manpower Dynamics

Inspired by Aksin's work on optimal policies for improving workers' productivity [8], we develop a set of equations for managing manpower dynamics. Let us first consider the update of employee numbers:

$$x_{\mathrm{E}}(k,j) = \begin{cases} u_{\mathrm{E}}(k,0) & \text{if } j=0 \\ x_{\mathrm{E}}(k-1,j-1) + u_{\mathrm{NE}}(k,j) - u_{\mathrm{LE}}(k,j) & \text{if } 1 \le j \le n_{\max}-1 \\ x_{\mathrm{E}}(k-1,j-1) & \text{if } j = n_{\max}. \end{cases} \tag{6}$$

The above equation can be interpreted as follows. For the case of $1 \le j \le n_{\max}-1$, $x_{\mathrm{E}}(k,j)$, the number of employees that are kept with j years of experience in the k-th year, should include

(i) the number of employees that had $j-1$ years of experience in last year (year $k-1$) and (ii) the number of new employees that are hired with j years of experience in this year, but should exclude the number of employees that are laid off this year with j years of experience. For the case of $j=0$, $x_{\mathrm{E}}(k,0)$ simply means the number of employees without any work experience in year k, and it should equal the number of new employees that are hired this year without any previous work experience. Finally, for $j=n_{\max}$, it is unlikely that an enterprise recruits or lays off someone who will retire immediately in next year, so $x_{\mathrm{E}}(k,n_{\max})$ includes just one component, i.e., the number of employees that had $n_{\max}-1$ years of experience in the past year.

Next we consider the update for the average strength or productivity of an employee:

$$x_{\mathrm{S}}(k,j)=\begin{cases}\delta_{\mathrm{S}}\left(\frac{u_{\mathrm{T}}(k,0)}{u_{\mathrm{NE}}(k,0)},0\right) & \text{if } j=0\\ x_{\mathrm{S}}(k-1,j-1)+\delta_{\mathrm{S}}\left(\frac{u_{\mathrm{T}}(k,j)}{x_{\mathrm{E}}(k,j)},j\right) & \text{if } 1\leq j\leq n_{\max}-1\\ x_{\mathrm{S}}(k-1,j-1) & \text{if } j=n_{\max}.\end{cases} \tag{7}$$

For the most common case where $1\leq j\leq n_{\max}-1$, $x_{\mathrm{S}}(k,j)$, the average strength of an employee with j years of experience in the k-th year, should inherit the strength from last year, i.e., $x_{\mathrm{S}}(k-1,j-1)$, and at the same time add on it the incremental strength after receiving a training with cost $u_{\mathrm{T}}(k,j)$. This incremental part is denoted $\delta_{\mathrm{S}}\left(\frac{u_{\mathrm{T}}(k,j)}{x_{\mathrm{E}}(k,j)},j\right)$, whose form is to be determined via investigation of some empirical data. For the case of $j=0$, $x_{\mathrm{S}}(k,0)$ cannot inherit anything from the past, and has to purely rely on the incremental strength obtained from training. For $j=n_{\max}$, it is unlikely that an enterprise would train someone who will retire immediately in next year, so $x_{\mathrm{S}}(k,n_{\max})$ is exactly the strength from last year, i.e., $x_{\mathrm{S}}(k-1,n_{\max}-1)$.

D. Assets

The service assets include both hardware and software facilities supporting service operations, such as IT servers, storages, and information databases. Business process is another type of assets. It formalizes a set of service operations, and coordinates human and IT resources. A good business process can help optimize service operations. These service assets are complementary to the capability of human employees. In this paper, we assume that the assets depreciate over time at a constant rate δ_{Z} but will increase following additional investment $u_{\mathrm{Z}}(k)$:

$$x_{\mathrm{Z}}(k)=(1-\delta_{\mathrm{Z}})x_{\mathrm{Z}}(k-1)+u_{\mathrm{Z}}(k). \tag{8}$$

E. Services Projects

As mentioned previously, the services enterprises considered here are project-based business. To win projects, a services enterprise has to compete with its peers in the market. The market position of the enterprise determines how many customers in the market will purchase its services. We define the amount of services orders from customers as *project demand*. Sometimes not all the orders can be handled by the services enterprise, because the capability of the enterprise is limited by its assets as well as the number and skill-levels of its employees. We define this capability of an enterprise in service as *project capacity*. If the services project demand is higher than services project capacity, the superfluous services orders will be lost. Consequently, the customer satisfaction-level will drop, and the advertising capital will decrease. If the project demand is lower than services project capacity, some employees have to be idle, and the enterprise wastes its human resources. Therefore, a services enterprise needs to match the dynamic demand with its services capacity.

We assume that all projects need to go through a bidding process. According to [7], the rate of project demand depends on the market position of the advertising capital, project price, and other variables not under the control of the services enterprise, such as consumer incomes, population, etc. Following [6], we use a simplified model for project demand:

$$D(k) = n_{\mathrm{P}}(k)d(k) \tag{9}$$

where $D(k)$ represents the enterprise's project demand in year k, $n_{\mathrm{P}}(k)$ denotes the total number of projects available in year k, and $d(k)$ is determined by

$$d(k) = a\left[\frac{u_{\mathrm{P}}(k)}{U_{\mathrm{P}}(k)}\right]^{-\eta}\left[\frac{x_{\mathrm{A}}(k)}{X_{\mathrm{A}}(k)}\right]^{\beta} \tag{10}$$

where η and β are defined as the elasticities of project demand with respect to price and advertising capital, respectively; $u_{\mathrm{P}}(k)$ is the project price offered by the enterprise in year k, and $U_{\mathrm{P}}(k)$ the sum of project prices offered by all the enterprises in the corresponding market; $x_{\mathrm{A}}(k)$ denotes the advertising capital in year k, and $X_{\mathrm{A}}(k)$ the total advertising capital of the whole market; and a is a normalization parameter such that the sum of $d(k)$ of all the services enterprises in the same market equals 1. This model has been empirically observed in many industries [6].

The project capacity of a services enterprise in year k, denoted $C(k)$, relies on the number of employee, their experiences and strengths, and the assets of the enterprise:

$$C(k) = C\left([x_{\rm E}(k,0),...,x_{\rm E}(k,n_{\max})]\,,[x_{\rm S}(k,0),...,x_{\rm S}(k,n_{\max})]\,,x_{\rm Z}(k)\right) \tag{11}$$

where the function $C(\,\cdot\,)$ needs to be determined from the empirical data.

The number of projects received by the enterprise in year k, i.e., $y_{\rm P}$, should be limited by both the project demand $D(k)$ and project capacity $C(k)$:

$$y_{\rm P}(k) = \min\{D(k), C(k)\}. \tag{12}$$

If $D(k)$ is larger than $C(k)$, the services enterprise does not have enough resources to support all the demands. For simplicity, we assume in the model that all the back orders will be lost. If $C(k)$ is larger, the enterprise is able to handle all the project demands, and thus $x_{\rm P}(k)$ equals $D(k)$. In the latter case, the extra resources will be idle, but the enterprise still needs to pay the expenses of these extra capacity, e.g., the salary of idle employee and the maintenance fee of hardware.

F. Income, Cost, and Reward

The total profit made by the enterprise in year k should equal the revenue gained from the projects received by the enterprise subtracted by

- Human resource expenses, which include (i) salaries paid to employee, (ii) training cost, (iii) hiring cost, (iv) layoff cost, and (v) retirement cost.
- Marketing spending, which includes advertising expense.
- Operational cost, which includes the office administration fee, project supporting expenses, and computing infrastructure cost. More people or more projects the enterprise has, more money it needs to pay for service operations. Meanwhile, the advanced assets such as SOA (service-oriented architecture)-based enterprise architecture and state-of-the-art service delivery processes can reduce the operational cost. Therefore, the operational cost is a function of the employee number, project number, and assets:

$$y_{\rm O}(k) = f_{\rm O}(x_{\rm E}(k), y_{\rm P}(k), x_{\rm Z}(k)) \tag{13}$$

  where $f_{\rm O}(\cdot)$ need to be determined from the empirical study.

Therefore, the profit (reward) of a services enterprise in year k can be represented as

$$\begin{aligned} r(x(k),u(k)) &= u_{\mathrm{P}}(k)y_{\mathrm{P}}(k) - u_{\mathrm{A}}(k) - y_{\mathrm{O}}(k) \\ &\quad - \sum_{j=0}^{n_{\max}} w_{\mathrm{S}}(j)x_{\mathrm{E}}(k,j) - \sum_{j=0}^{n_{\max}-1} u_{\mathrm{T}}(k,j) - \sum_{j=0}^{n_{\max}-1} w_{\mathrm{H}}(j)u_{\mathrm{NE}}(k,j) \\ &\quad - \sum_{j=1}^{n_{\max}-1} w_{\mathrm{L}}(j)u_{\mathrm{LE}}(k,j) - w_{\mathrm{R}}x_{\mathrm{E}}(k,n_{\max}). \end{aligned} \tag{14}$$

G. Optimality Criteria and Services Enterprise Index

As a result of choosing and implementing a policy, the services enterprise receives rewards in year $1, ..., K$. The business objective is to choose a series of decisions in order to maximize the accumulated rewards. In the services industry, the performance measures consist of multiple assessment elements, e.g., efficiency, productivity, and enterprise-wide profitability. We need to build a multi-factor quantitative performance evaluation framework to monitor the performance of services enterprise.

Multiple indicators of performance evaluation obviously lead to conflicts. Short-term and long-term trade-offs need to be considered. For example, by reducing marketing and training expenses, current costs will decrease and profits will increase. However, these might be exactly the wrong things to do to maximize long-term profitability. Considering the short-term and long-term trade-offs, we propose to use *services enterprise index* (SEI) to reflect a specific business goal. For this paper and specifically the case study in Section IV, we consider a simplified SEI, which is the total profits in a given decision period. Our objective is to

$$\text{maximize} \quad V_K(x_0, \mathbf{u}) = \sum_{k=1}^{K} r(x(k), u(k)) \tag{15}$$

subject to (5)-(14).

III. Dynamic Programming Formulation and Algorithm for DMM4SE

In this section, we formulate the above optimal-control problem into a dynamic programming (DP) problem. However, the computation required for finding the optimal solutions increases exponentially with the number of state variables. Therefore, we develop an approximate dynamic programming (ADP) algorithm which uses cubic-spline interpolation to simplify the computation.

A. DP Formulation

The firm starts the business period with an initial state $x(0) = x_0$. Let policy $\mathbf{u}$ denote the series of decisions at all periods (from 1 to K). Denote $\mathbf{\Pi}$ the space of the policies, and define the optimal value function of the total profits as

$$V_K^*(x_0) = \max_{\mathbf{u}\in\mathbf{\Pi}} V_K(x_0, \mathbf{u}). \tag{16}$$

Policy $\mathbf{u}^*(x_0)$ is said to be the optimal policy (given the initial state x_0) if

$$V_K(x_0, \mathbf{u}^*(x_0)) = V_K^*(x_0). \tag{17}$$

We use forward induction to solve the optimal control problem, since our problem has a fixed initial state x_0 and a floating terminal state. For any $k \in \{1, ..., K\}$ and given state $x(k)$, let $U_k(x(k))$ denote the optimal profit accumulated from year 1 to year k while the state variable evolves from x_0 to $x(k)$. Then we have the following optimality condition for $U_k(x(k))$:

$$U_k(x(k)) = \max_{\substack{x(k-1), u(k) \text{ subject to} \\ f(x(k-1), u(k)) = x(k)}} \{r(x(k), u(k)) + U_{k-1}(x(k-1))\} \tag{18}$$

where $f(\cdot)$ is defined in (1). In year 0, the enterprise receives no profit, so the boundary condition is $U_0(x_0) = 0$. Based on the above definitions, $V_K^*(x_0)$ equals the maximum value of $U_K(x(K))$ over all possible final state $x(K)$ evolved from x_0.

B. ADP Algorithm

Solving the above DP problem is computationally intensive for high dimensional state space. Therefore, we develop a heuristic method based on ADP (see [9] for a comprehensive review of ADP). We utilize an interpolation-based algorithm to reduce the time and memory required of of computation. The standard forward induction method [10] stores the value function for all possible states of a DP in each period; our method stores the optimal values for only a small subset of the entire state space (which we refer to as anchor values) and approximates the values of other states through *ad hoc* spline interpolation. The spline interpolation method produces interpolants that are simple, yet flexible and smooth piecewise polynomial functions [11]. In particular, we use cubic splines to interpolate the multidimensional value function of the DMM4SE. Our algorithm embeds these interpolated values within a standard forward induction method and generates near-optimal solutions. Similar algorithms have been successfully implemented to reduce computational complexity in large-dimensional dynamic programs [12], [13].

Let us introduce some notation. Define Γ as the state space, which contains all possible values of the state variable x, and let n be the total number of the dimensions of Γ. Denote γ_i the number of anchor points for the cubic spline interpolation in the i-th dimension of the state space, and denote $\hat{\Gamma}_i$ the set of these anchor points in the i-th dimension. Define $\hat{\Gamma}$ as $\hat{\Gamma}_1 \times \hat{\Gamma}_2 \times \ldots \hat{\Gamma}_n$. This set is a subset of Γ and contains all the points whose coordinates are some anchor points in $\hat{\Gamma}_i$, $i = 1, ..., n$.

The basic idea of our ADP algorithm is as follows. We compute the approximate values of $U_k(x)$, denoted by $\hat{U}_k(x)$, for those anchor points in $\hat{\Gamma}$, using the estimates of $\hat{U}_{k-1}(x)$ and the optimality equation (18). Then we find a multidimensional cubic spline function passing through $\hat{U}_k(x)$ for all $x \in \hat{\Gamma}$. The value of $U_k(x)$ for $x \notin \hat{\Gamma}$ can be approximated by interpolation using the cubic spline function. Initially, we can exactly solve the optimal $U_1(x)$ and let $\hat{U}_1(x) = U_1(x)$. Iteratively, we get the near-optimal values $\hat{U}_k(x)$ for $k = 2$ through K. Then we use backward induction to compute the optimal decision series, i.e., $u^*(K)$, $u^*(K-1)$, ..., $u^*(1)$. Details of the interpolation based ADP algorithm are described in the following five steps:

1) For $k = 1$, compute $U_1(x)$ based on the optimality equation (18) with boundary condition $U_0(x_0) = 0$, and then let $\hat{U}_1(x) = U_1(x)$.
2) For $k = 2$ to K:
   a) Determine the anchor points for interpolation:
      i) Determine the minimum (denoted $x_{i,\min}$) and maximum (denoted $x_{i,\max}$) for all possible values of x_i, where x_i denotes the i-th element of x.
      ii) Choose γ_i points from $[x_{i,\min}, x_{i,\max}]$ such that these points divide the interval into approximately equal partitions.
   b) For each $x \in \hat{\Gamma}$, compute $\hat{U}_k(x)$ using the optimality equation (18), i.e.,

$$\hat{U}_k(x) = \max_{\substack{x', u \text{ subject to} \\ f(x', u) = x}} \left\{ r(x, u) + \hat{U}_{k-1}(x') \right\}.$$

   c) For $x \notin \hat{\Gamma}$, calculate the approximate value of $\hat{U}_k(x)$:
      i) Construct a cubic spline function $S(x)$ such that $S(x') = \hat{U}_k(x')$ for all $x' \in \hat{\Gamma}$.
      ii) For all $x \in \Gamma \backslash \hat{\Gamma}$, interpolate $\hat{U}_k(x) = S(x)$.

3) For $k = K$, compute $x^*(K) = \arg\max_{x \in \Gamma}\{\hat{U}_K(x)\}$.

4) For $k = K$ to 2:

   a) Construct the cubic spline function $S(x)$ such that $S(x') = \hat{U}_{k-1}(x')$ for all $x' \in \hat{\Gamma}$.

   b) For all $x \in \Gamma \backslash \hat{\Gamma}$, interpolate $\hat{U}_{k-1}(x) = S(x)$.

   c) Compute $x^*(k-1)$ and $u^*(k)$:

$$\{x^*(k-1), u^*(k)\} = \underset{\substack{x,\, u \text{ subject to} \\ f(x,u) = x^*(k)}}{\arg\max} \left\{ r(x^*(k), u) + \hat{U}_{k-1}(x) \right\}.$$

5) For $k = 1$, compute $u^*(1)$ such that $f(x_0, u^*(1)) = x^*(1)$.

IV. Case Study

To illustrate the usage of our optimal-control based decision-making model, we analyzed a sample dataset from the Beacon business simulation practice [14]. This simulation practice was conducted in the IBM Research's Micro-MBA Program in 2006, and involved four enterprises run by four teams of participants. The four teams competed against each other in the market for various products and services. In the simulation practice, each enterprise had both manufacturing business and services business, but for this study we focused on the services business. In each year, the teams made decisions on the project pricing, marketing spending, employee hiring or laying-off, and employee training. Each team aimed to maximize their total profit within a fixed period. The participants in this simulation practice were all professionals and experts in business operations and service management. Therefore, the dataset can reflect the behavior and dynamics of services enterprises in the real world.

Fig. 1 shows the performance of all the four teams or services enterprises from year 8 to year 12. One of the enterprises, the South Service Inc., or the South for short, outperformed the other three enterprises in terms of profits. Although the project sales [Fig. 1(b)] of the South was not the highest, this team successfully leveraged the project pricing [Fig. 1(c)] and employee power [Fig. 1(d)]. With a relatively higher project price, it maintained the service capacity with a stable employee team. In doing so, it beat other teams in the simulation practice. Therefore, in this case study we chose the South as the target of our analysis so as to demonstrate the effectiveness of our approach: We calculated the optimal policy for this enterprise based on our decision-making model and ADP algorithm, and then compared our solution with the recorded performance in the simulation practice.

Fig. 1. Performance comparison of four services enterprises

A. Data Analysis and Model Fitting

The dataset provides us the information about operations of the South from year 5 to year 12. The data include information about the employees, their starting and ending skill-levels, history of their status (whether working or idle), number of employee per contract, and cost per contract. The data also provide information of project sales, price, total revenue, cost of working and idle employees, gross profit, administration cost, expenses in advertising and training, severance payment, operating profit, and lost orders. Furthermore, the data include all the decisions made by the South about project pricing, marketing spending, and employee hiring, firing, and training.

In this case study, the employee's annual salary (w_S) was $\$100K$, no matter working or idling. The employee layoff cost (w_L) was $\$30K$. There was no cost for hiring new employees. The Beacon dataset does not have the detailed information about the experiences and skill-levels of individual employees. However, the average skill-level of all employees is available for each year.

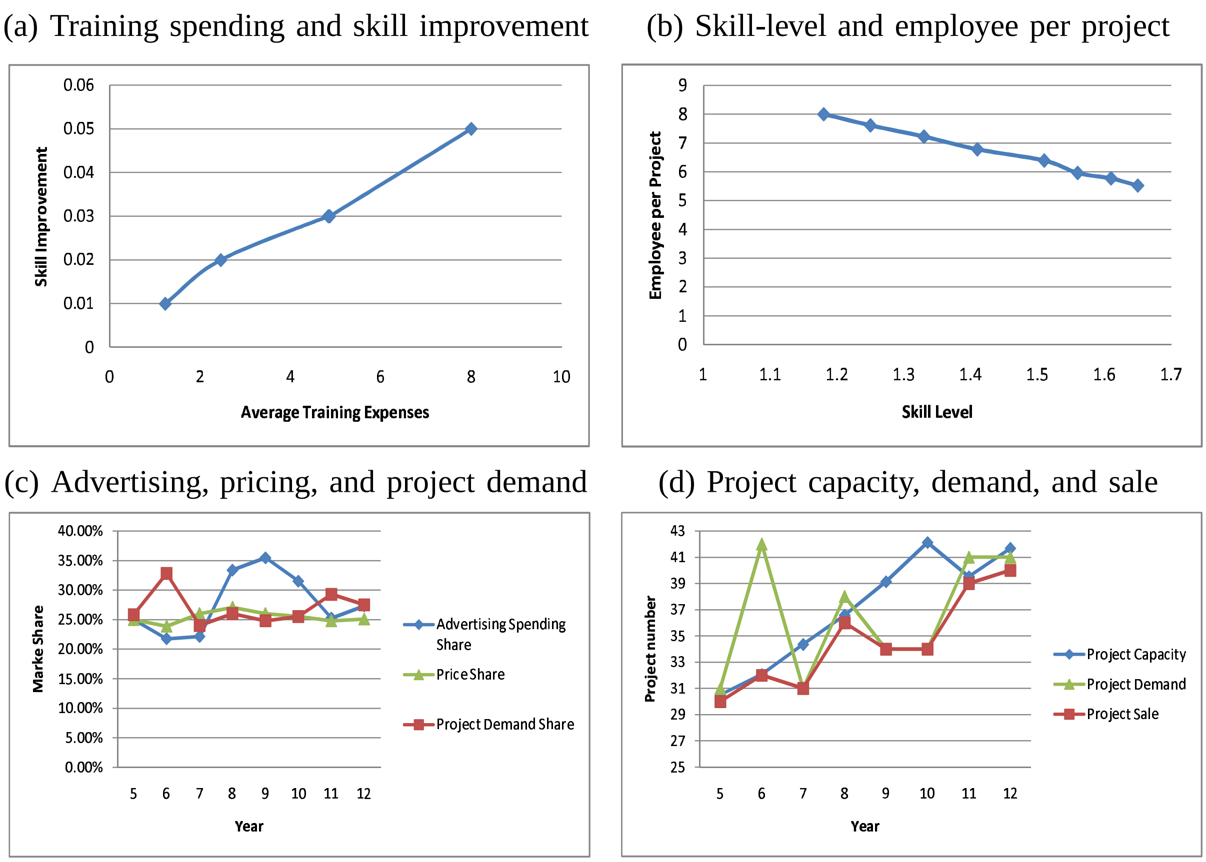

Fig. 2. Data analysis of the Beacon project.

Fig. 2(a) shows the relation curve for employee skill-level improvement with respect to training expense. We ran a regression and obtained the following function for the skill-level improvement of employees:

$$\delta_{\mathrm{S}}\left(\frac{u_{\mathrm{T}}(k)}{x_{\mathrm{E}}(k)}\right) = \frac{-3.81}{\frac{u_{\mathrm{T}}(k)}{x_{\mathrm{E}}(k)} + 21.6} + 0.226$$

where $u_{\mathrm{T}}(k)$ is the total training expense in year k, $x_{\mathrm{E}}(k)$ is the total number of employees in year k, and $\delta_{\mathrm{S}}(\cdot)$ is the average incremental strength of an employee after receiving training of expense $u_{\mathrm{T}}(k)/x_{\mathrm{E}}(k)$ per employee. The above equation was used in the update for the average strength of an employee [see (7)].

Fig. 2(b) shows that the employee number per contract decreased linearly with the increase of the employee skill-level. Therefore, the project capacity of the South can be determined by the number and skill-levels of its employees. Using a linear regression, we had

$$C(k) = \frac{x_{\mathrm{E}}(k)}{-5.21 x_{\mathrm{S}}(k) + 14.1}.$$

Fig. 2(c) shows how the project demand varied with the price and advertising spending. Here we use the term "share" to represent the portion of a quantity in the market. For example, price share means the project price offered by the South divided by the sum of project prices offered by all the four enterprises in the simulation practice. We used the curves in Fig. 2(c) to fit (5) and (10), and obtained

$$x_{\mathrm{A}}(k) = 0.99\ x_{\mathrm{A}}(k-1) + u_{\mathrm{A}}(k),$$

and

$$D(k) = 0.22 \left[\frac{u_{\mathrm{P}}(k)}{U_{\mathrm{P}}(k)}\right]^{-0.43} \left[\frac{x_{\mathrm{A}}(k)}{X_{\mathrm{A}}(k)}\right]^{0.33}.$$

Knowing the project capacity, demand, and sales [i.e., $y_{\mathrm{P}}(k)$, the number of projects received by the enterprise], we plot them in Fig. 2(d). The figure shows that project sales were bounded by both the project capacity and demand [see (12)]. In year 5, 7, 8, 10 and 12, the sales were constrained by the project demand. In year 6, 8 and 11, the project sales were limited by the project capacity of the enterprise; in other words, the South did not have enough resources to support all the potential projects in those years. This observation supports our model built in (12).

B. Performance Analysis

Based on the model obtained in Section IV-A, we implemented our interpolation-based ADP algorithm (in MATLAB [15]) to find the optimal policy for the South. In this case study, we had five decision variables: project price, number of hiring employees, number of layoff employees, advertising expense, and training spending; and we had three state variables: employee number, employee average skill-level, and advertising capital. A simplified SEI used here was the total profit over a period of five years.

Fig. 3 presents the human-resource related decisions and consequences produced by our algorithm. For comparison, the figure also plots the original decisions and performance of the South. The employee number was significantly larger in our solution than in the simulation practice of the South [Fig. 3(a)]. Except for year 11, the decision suggested by our algorithm

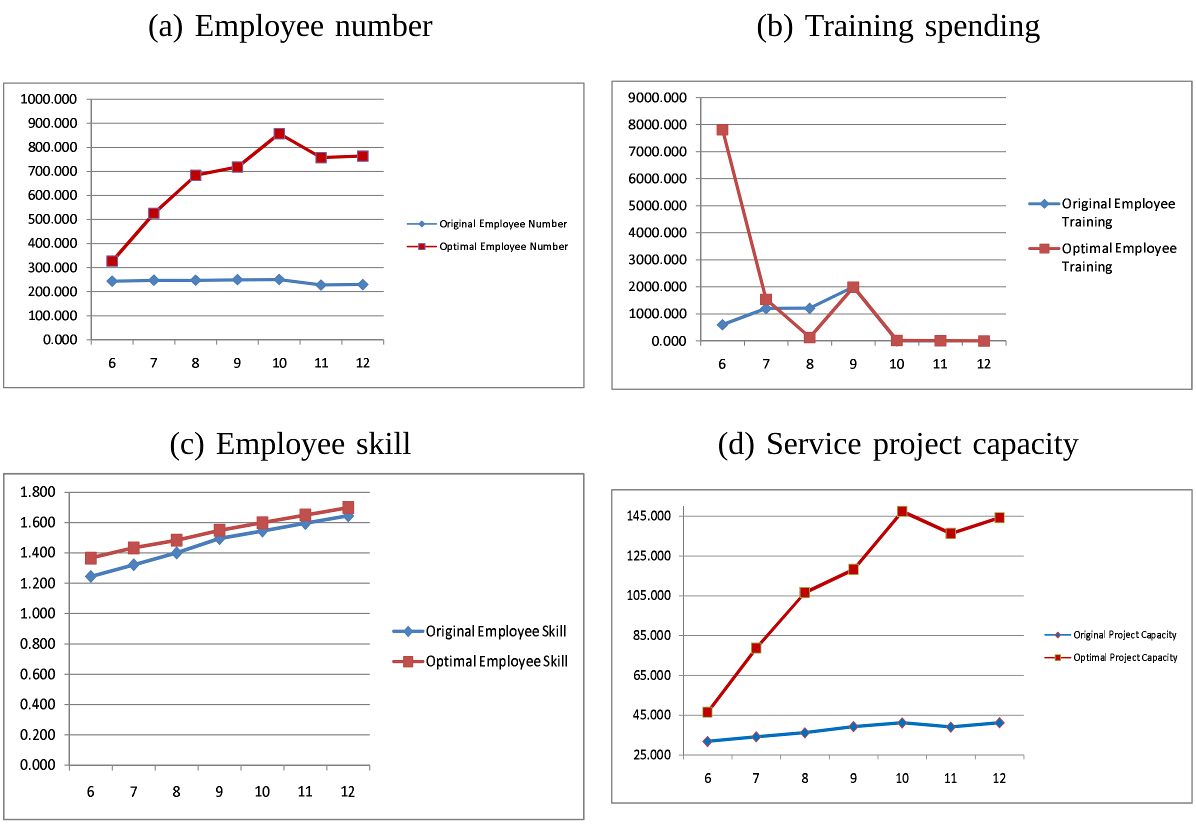

Fig. 3. Comparison of human resource operations.

was to hire new employees in each year. The training expense in the our solution experienced a big jump in year 6, but remained at low levels afterwards [Fig. 3(b)]. As a result, the average employee skill-level was improved steadily [Fig. 3(c)], and the project capacity was expanded to a very high level [Fig. 3(d)].

Fig. 4 presents the marketing related decisions and consequences produced by our algorithm. A straightforward way of gaining profit is to increase the project demand. By adjusting the price and increasing the marketing spending, the project demand can be increased. For marketing spending, the decisions suggested by our algorithm had little difference with the original decisions of the South in the simulation practice [Fig. 4(a)]. However, for decisions on pricing, the difference was significant [Fig. 4(b)]. The strategy suggested by our algorithm was to seize large market share by offering low prices. In doing so, the project demand was increased dramatically [Fig. 4(c)]. Note that our solution was derived based on the simplified price-marketing-demand model in a closed market. The real business situation may be much more complicated.

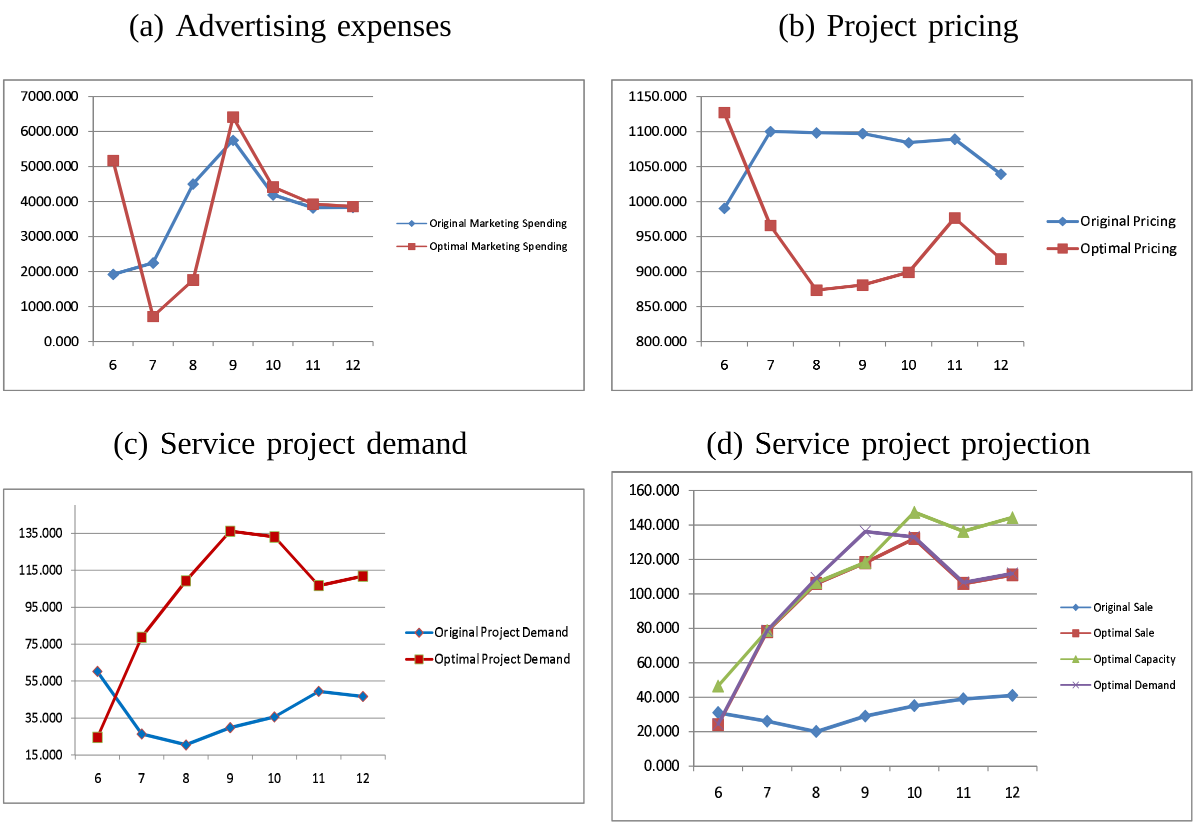

Fig. 4. Comparison of marketing operations.

Compared with the results shown in Fig. 2(d), our solution offered a better match between the project capacity and project demand, as demonstrated in Fig. 4(d). This implies that our solution made better use of the enterprise resources (with less wasting or idling of human resources). As a result, there was a big improvement in project sales, which almost tripled the original sales.

Fig. 5 shows the predicted profit in our solution and the original profit of the South in the simulation practice. Except for year 6, the profit produced by our algorithm was much larger than the original profit. In year 6, our solution suggested the enterprise put more efforts in building a strong workforce and augmenting the market share—these efforts established a solid ground for the development of the enterprise in the following years. In summary, the total predicted profit in our solution was $\$106,792K$, which doubled the original profit, $\$53,208K$. This case study has demonstrated the effectiveness of our decision-making model and solution.

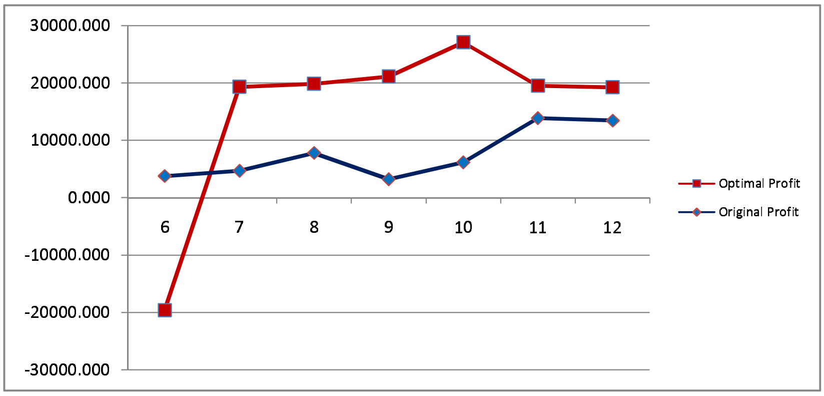

Fig. 5. Comparison of profit.

C. *Effectiveness of ADP Algorithm*

To assess the effectiveness of our ADP algorithm, we compared its performance with that of the standard forward-induction based DP algorithm. The DP algorithm may find the optimal solutions to the optimal control problems formulated in this paper, but is highly computationally demanding. In the previous case study, our interpolation-based ADP algorithm yielded an outstanding performance: It obtained a near-optimal solution with a gap of less than 0.5% to the actual optimal solution, while using less than 1% of the computer memory and 15% of the CPU time required by the standard DP algorithm.

V. A Consulting methodology Based on DMM4SE

In this section, we propose a consulting methodology based on our decision-making model for services enterprises (DMM4SE). Our methodology aims to coordinate and integrate different components of an services enterprise (e.g., manpower, asset, advertising, and pricing) so as to achieve centralized business goals. The proposed consulting methodology consists of the following steps (Fig. 6).

Step 1. Model training: In this step, historical operation data and related market information are collected. Given the past running information of an enterprise, each submodel in DMM4SE can be fitted as discussed in Section II. Sometimes services enterprises may not maintain a comprehensive historical data, and some enterprises may lack the evaluation system for abstract

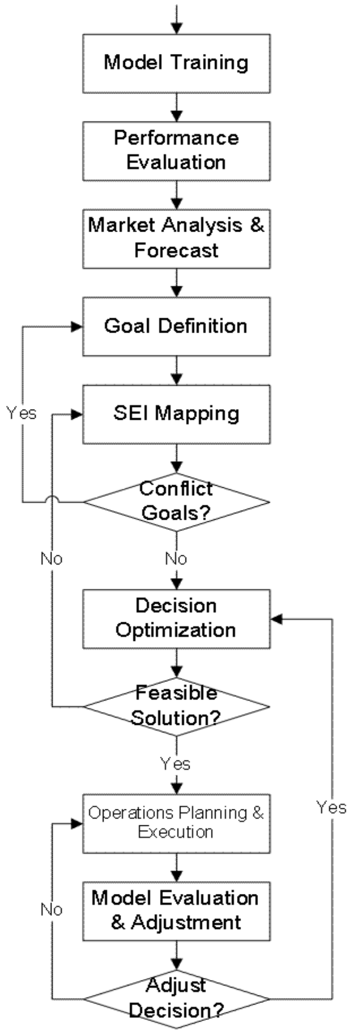

Fig. 6. DMM4SE-based consulting methodology.

data such as advertising capital, employee skill-level, and asset value. Under these circumstances, the consulting team needs to mine the abstract data from other information sources. Furthermore, market information may not be accessible to public. To obtain pricing and advertising data of a specific market segment, the consulting team needs the support from third-party data service firms.

Step 2. Performance evaluation: Based on the fitted model, optimal decisions for the past periods are calculated by running the proposed ADP algorithm. A comparison between the actual and the optimal operations can provide evaluation of the past actions of the enterprise.

Step 3. Market analysis and forecast: Strategic planning of services enterprises relies on accurate forecasting of market demands. The services market can be highly dynamic. Different market sectors may behave differently, and even in the same market sector competitors may follow significantly different strategies. However, since we focus on strategic decision-making, market demands and competitor strategies can be aggregated and smoothed—this lowers the risk and uncertainty in forecasting.

Step 4. Goal definition: Competing enterprises may have different goals and consequently take different strategies. For example, three generic competitive strategies for services firms are proposed in [16]. They are *Overall Cost Leadership*, *Differentiation*, and *Focus*. A low-cost position relies on efficient-scale asset and skilled employees. Differentiation aims at customer loyalty and lies in creating novel services. The focus strategy rests on the premise that a firm can serve its narrow target market more effectively and efficiently.

Step 5. SEI mapping: By using the decision map (Fig. 7), each strategic goal can be mapped into one or more SEIs. Each SEI consists of four perspectives, i.e., finance, operations, marketing, and innovation, and each perspective is linked to the operation details. With this decision map, the target values of specific operation variables can be created. Sometimes, different goals are not compatible with each other. For example, to achieve a cost leadership, a services enterprise should reduce redundant resources such as idle employees and assets. However, to dominate a market segment and attract more customers, a services enterprise needs to respond to unexpected events such as peak demand. Therefore, a certain amount of redundant resources is necessary. Via SEI mapping, the consulting team identifies goal congruency, i.e., how well the goals match with each other. If some conflicts are not amendable, the consulting team has to go back to Step 4 and modify the goal definitions.

Step 6. Decision optimization: Based on the forecast of market demands and competitors' strategies, optimal decisions for the future periods are calculated using the proposed ADP algorithm. Since multiple constraints are added via SEI mapping, feasible solutions may not always exist. For those cases, the consulting team revisits Step 5 and adjusts the SEI mapping. If the SEI adjustment cannot reconcile the conflicts, the consulting team should go back to Step 4 and redesign the strategic goals. Generally, constraints can be categorized into different levels, from critical to optional. For instance, in a competitive market segment, demand is the most influential constraint. It is not easy to leverage the demand via pricing and advertising.

Therefore, maintaining existing customers is the most essential task. However, in an emerging market segment, innovation is the differentiation power. Domain experts and superior assets are scarce resources in the industry. Therefore, the service capacity is the primary constraint. The priority of constraints can be incorporated into the decision model by designing corresponding penalty functions.

Step 7. Operations planning & execution: The DMM4SE focuses on decision-making in the strategic level. Each submodel in Section II is aggregated, summarized, and simplified to represent the major characteristics of the full model of the enterprise dynamics. Following the optimal decisions calculated for the full model, the enterprise operates its service business, subject to dynamic changes of the real business environment.

Step 8. Model evaluation and adjustment: At the end of each decision period, the services enterprise compares the realized results with the predicted results. If the realized results conform to the prediction, the model is validated and the calculated optimal decisions are reliable. If there is a significant difference between the realization and prediction, the consulting team needs to determine whether this is due to incorrect market forecasting or incorrect model. If the model is not accurate, it needs to be refined, and the consulting team should return to Step 3 and generate new optimal decisions. Besides the model evaluation, the services enterprise also needs to evaluate its business performance, and identifies its position changes in industry. If necessary, the strategy is adjusted to fit the dynamic services market.

VI. Discussion

A. Services Ecosystem

The proposed DMM4SE can be generalized to facilitate collaboration in a services ecosystem. In the services ecosystem, customers raise the requirement and pay for the services; services enterprises provide the services; and service-component providers focus on specific functionality that is reusable in a variety of industries and customers. Services business, unlike the traditional business, relies on the collaboration of all these “players” in the services ecosystem. Interaction between customers and enterprises determines the result of a services engagement. Different services enterprises may compete in some market segments but collaborate in other market segments to provide advanced solutions, which are beyond the resource or capability limits of individual enterprises. The service component providers also need to interact with clients so as to create

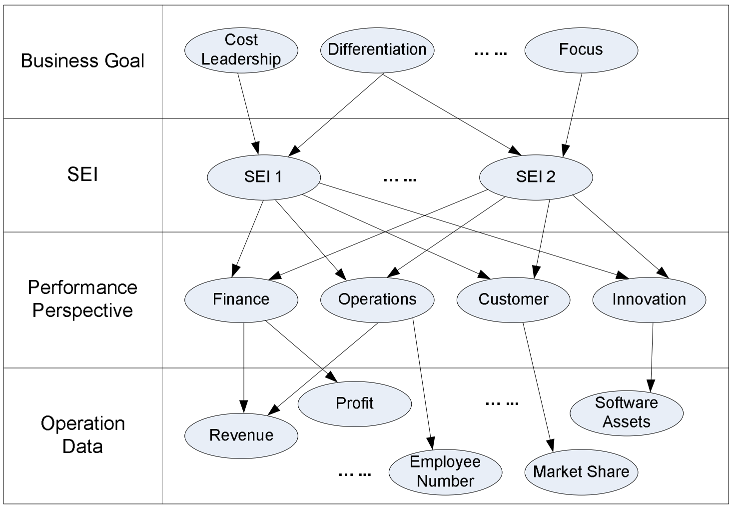

Fig. 7. Decision map of service business.

service innovations. Therefore, how to manage partnership, outsourcing relationship, and parent-child relationship affects the decision-making of a services enterprise. Service enterprises have to consider their roles in the entire ecosystem in order to achieve optimal business performance. In our future work, we will incorporate in the DMM4SE the above mentioned collaboration among different components of the services ecosystem.

B. Prototype System

Using the newly released IBM Lotus Mashup Center [17], we can build a prototype system to deliver consulting services for services enterprises. Fig. 8 shows the architecture of the prototype system.

- Via BizRSS Feed [18], *Data Collector* retrieves operation data of a specific enterprise. By connecting to the public data portal, the data collector can obtain the finance and marketing information from related industry. The attained data is saved in internal data warehouse for short-term and long-term analysis.
- *Learning Engine* analyzes the data and fits the parameters of the sub-models for manpower dynamics, asset capacity, advertising capital, and market demand. Then, the four sub-models are integrated into the complete decision-making model.

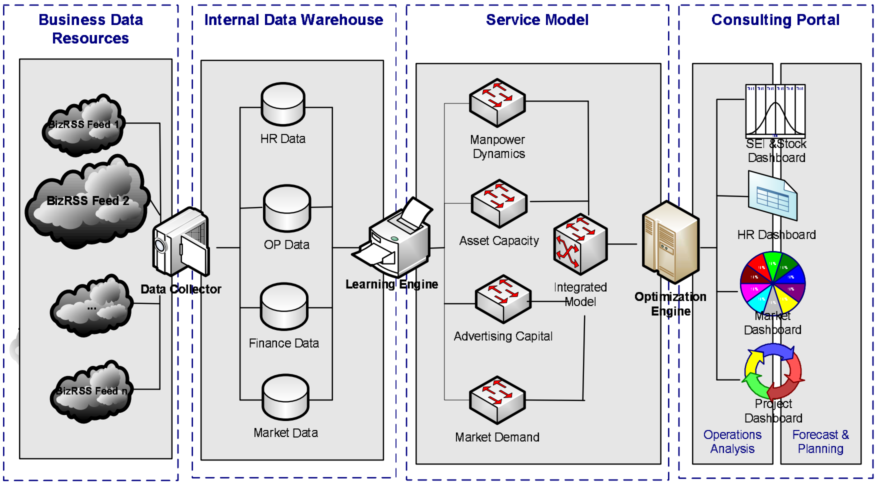

Fig. 8. Prototype system architecture.

- Given the decision-making model, *Optimization Engine* runs the proposed ADP algorithm and output the optimal decisions.
- *Consulting Portal* provides for consultants a set of dashboards, which include SEI & Stock, human resource, market & customer, and project operations. These dashboards help consultants to prepare the business planning reports for services enterprises.

VII. Conclusion

This paper formulated the decision-making of a services enterprise as an optimal control problem. This allowed mathematical description of services dynamics, and facilitated quantitative analysis for the operation and strategic planning of services enterprises. An approximate dynamic programming algorithm was proposed to solve the optimal control problem, and successfully handled the model complexity by simplifying the computation of value functions using cubic spline interpolation. The optimal-control based decision-making model was validated in a simulation practice of the Beacon business. The results demonstrated the feasibility and effectiveness of the proposed model and solution approach. Furthermore, a DMM4SE-based consulting methodology was introduced and its usages for services consulting practices were discussed.

REFERENCES

[1] (2008) The Service Annual Survey, U.S. Census Bureau. [Online]. Available: http://www.census.gov/econ/www/servmenu.html

[2] (2008) The 2008 World Factbook, U.S. Central Intelligence Agency. [Online]. Available: https://www.cia.gov/library/publications/the-world-factbook/fields/2012.html

[3] J. Gharajedaghi, *Systems Thinking*, 2nd ed. New York: Butterworth-Heinemann, 2005.

[4] D. W. McDavid, "A standard for business architecture description," *IBM Systems Journal*, vol. 38, no. 1, pp. 12–31, 1999.

[5] R. Dorfman, "An economic interpretation of optimal control theory," *American Economic Review*, vol. 59, no. 5, pp. 817–831, Dec. 1969.

[6] M. Nerlove and K. J. Arrow, "Optimal advertising policy under dynamic conditions," *Economica*, vol. 39, pp. 129–142, 1962.

[7] S. P. Sethi, "Dynamic optimal control models in advertising: a survey," *SIAM Review*, vol. 19, no. 4, pp. 685–725, Oct. 1977.

[8] O. Z. Aksin, "On valuing appreciating human assets in services," *Naval Research Logistics*, vol. 54, pp. 221–235, 2007.

[9] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken, New Jersey: Wiley Interscience, 2007.

[10] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, New Jersey: Wiley Interscience, 2005.

[11] E. V. Shikin and A. I. Plis, *Handbook on Splines for the User*. New York: CRC Press, 1995.

[12] S. A. Johnson, C. A. Shoemaker, Y. Li, J. A. Tejada-Guibert, and J. R. Stedinger, "Numerical solution of continuous-state dynamic programs using linear and spline interpolation," *Operations Research*, vol. 41, no. 3, pp. 484–500, May-June 1993.

[13] M. A. Trick and S. E. Zin, "Spline approximations to value functions: a linear programming approach," *Macroeconomic Dynamics*, vol. 1, no. 1, pp. 255–277, Jan. 1997.

[14] (2005) Beacon Business Simulation, Nth Degree Systems Inc. [Online]. Available: http://www.nthdegreeonline.com/

[15] (2008) MATLAB, MathWorks. [Online]. Available: http://www.mathworks.com/products/matlab/

[16] M. E. Porter, *Competitive Strategy*. New York: Free Press, 1980, ch. 1, pp. 11–15.

[17] (2008) IBM Lotus Mashup Center. [Online]. Available: http://www-306.ibm.com/software/info/mashup-center/

[18] L.-J. Zhang, A. Allam, and C. A. Gonzales, "Service-oriented order-to-cash solution with business RSS information exchange framework," in *Proc. IEEE Internationl Conference on Web Services*, Chicago, IL, Sept. 2006, pp. 841–848.