An Empirical Model of Mainframe Computer Replacement*

Sung-Jin Cho†
Department of Economics
Yale University

December, 2000

Abstract

This paper formulates a stochastic optimal stopping model for the replacement of mainframe computer systems in the telecommunications industry. It describes the replacement behavior by focusing on unique features of computer systems, which are associated with technological development. The optimal stopping rule is the solution of a stochastic dynamic programming model that specifies the system administrator's objective to maximize profits through three choices: 'keep', 'upgrade', or 'replace'. The model depends on unknown parameters which govern both the profit structures of the task level of the company and the system administrator's expectation of the future values of the state variables.

Using a detailed data set on computer holdings by one of the world's largest telecommunication companies, I investigate the stylized facts of computer replacement and estimate the model with the nested fixed point algorithm. The estimation requires two procedures: (i) a parametric approximation procedure which converts the contraction fixed-point problem into a nonlinear least squares problem; (ii) maximum likelihood estimation method to estimate the unknown parameters. The estimation supports the observed stylized facts of the data in general, allowing for better understanding of the replacement behavior in the era of rapidly growing computer technology. I also show the effectiveness of the parametric approximation method in comparison with the discretization method.

---

*I give special thanks my to advisor, John Rust for invaluable advice and encouragement. I also thank Steven Berry and Martin Pesendorfer. I am indebted to the provider of the computer data who wishes to remain anonymous. I have greatly benefitted from discussions with Janryoul Kim. All errors are my own.

†Contact information: e-mail sungjin.cho@yale.edu, homepage http://www.econ.yale.edu/~sungjcho, phone (203) 432-3559, fax (203) 432-5779

1 Introduction

Despite the importance of computers in the "information economy", comparatively little is known about the factors affecting upgrade and replacement decisions. In the face of rapid technological progress and steadily declining costs, consumers and firms must decide whether to upgrade or replace existing computer systems now, or wait to purchase a faster/cheaper system in the future.

Regarding systems replacement in general, there has been previous research on the replacement of bus engines (Rust, 1987), aircraft engines (Kennet, 1994), cement kilns, and nuclear power plants (Rust, 1997). However, computers unlike the aforementioned systems above have the following features which make them unique. First, in computer systems, upgrade is an alternative to replacement when attempting to improve performance. In case of bus or aircraft engines, there is no upgrade choice. In fact, for computer systems, upgrading is sometimes the first choice over replacement.

Second, the main reason to replace engines is to prevent a future failure and capacity improvement is a secondary reason. In contrast, though the prevention of future failure can be a reason to replace or upgrade computer systems improvement of performance and meeting demand for service is the main reason for replacement.

Third, while bus and aircraft engines are replaced due to physical depreciation, such as natural wear-out and mechanical failure, replacement of computer systems are usually caused by technological depreciation. In research on the replacement of engines, state variables are the hours of operation and the history of engine shutdowns. These state variables represent various measurements of physical depreciation.

However, these variables may not be appropriate in a model for computer systems replacement. In the case of computer systems, the replacement caused by physical depreciation occupies a relatively small fraction of the entire set of replacements. Thus, state variables for replacements of computer systems should be measurements of technical depreciation. One of supporting examples is as follows: according to Moore's law each new chip contains roughly twice as much capacity as its predecessor, and each chip is released within 18-24 months of the previous chip. Currently, this law works properly. Figure 1 illustrates that how Moore's law explains developing trend of computer technology.[1] The time frame of my data starts from1989 and ends on 1998, where 250k

[1] Montemerlo, Michael S. and Love, J. Christopher and Opiteck, Gregory J. and Gordon, David Goldhaber. and Ellenbogen, James C.(1996): "Technologies and Designs for Electronic Nanocomputers," MITRE MSR Program,

transistors per sq cm has changed to over 4M transistor sq cm according to figure 1. In that period, computer technology had been developed tremendously and old technology becomes obsolete in much faster way. Thus, possible candidates of state variables should reflect this trend. Possible candidates are the following: (i) an introduction of new operating system (a new operating system may require a more advanced system to work properly); (ii) the difference of CPU speed between the current system and the fastest system available. The continuous introduction of new CPUs in the market makes the relative speed of old CPUs decrease and thus the relative operating costs become higher than having systems with new CPUs, i.e., technically, the CPUs of the current systems continuously depreciate.

Fourth, replacement of computer systems requires us to deal with a more complicated decision process than that of engine replacements. Here, we first have to decide whether to replace, upgrade, buy an additional system or keep the current system. These decisions are considered as the main choices. Contingent on these main choices, we are confronted with a set of sub-choices. For instance, a replacement decision requires other subsequent choices, such as the capacity and the brand of new computer systems. [2]

Due to these unique features, Rust's optimal rule is not directly applicable to the replacement of computer systems. Therefore, a new methodology that can effectively analyze computer systems is needed to better understand computer systems replacement behavior. The objective of this research is to attain a general optimal upgrade and replacement rule for computer systems by using currently available data and estimation methods.

This paper presents a dynamic programming model of a firm's decision of whether to keep, upgrade, or replace an existing computer subject to uncertainty over the timing and magnitude of future cost reductions for computer systems. I estimate this model using a detailed data set on computer holdings of one of the world's largest telecommunications companies. A number of "stylized facts" are evident from an initial analysis of these data. These facts support that the presence of rapid technological progress affects the firm's replacement and upgrade policy along

---

July 1996., page 2

[2] Other unique features are as follows: Unlike cases of engine replacements, vendors' service support may play an important role in the replacement decision of computers. Since vendors usually do not support old systems without an extra service contract, maintaining old computer systems may be more costly than replacing them with new computers.

Second, we may examine to what extent the replacement behaviors are done individually or on a "fleet replacement" basis due to costs of training administrative staff. In many cases, block purchases of computer can give the firm a quantity discount. These features can be considered in the future extension upon availability of richer data.

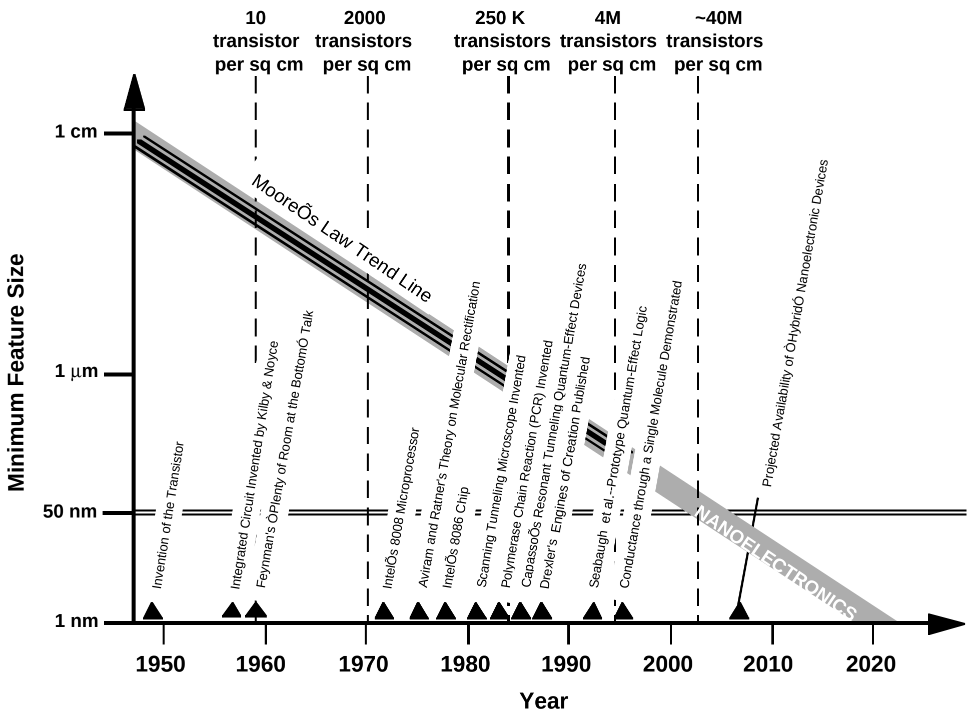

FIG. 1. Genesis of Nanotechnology. A timeline of selected key events plotted versus time with Moore's Law trend line.

2

Figure 1:

with the economic development.

The formal analysis begins in section 4. I develop a stochastic dynamic programming model to see whether these stylized facts of replacement and upgrade behavior can be rationalized as an optimal investment strategy for this firm. In the model, the firm has three possible actions at each time period: keep, upgrade, or replace. The state variables include the processing capacity of the current system, the level of demand for this processing capacity, the age of the current system, and the current market price of a standardized unit of processing capacity. The technological depreciation and the relative performance of each computer system are measured by composite measures of all four state variables in the model. The model depends on the unknown primitive parameters that specify the firm's profit function and its expectation of future values of the state variables, with its expectation of future reductions in the price of computing capacity playing a critical role in the model's predictions of the optimal length of the replacement cycle.

In section 5, I show the needs of a parametric approximation, which greatly reduces the computational burden involved in solving the infinite-horizon version of the model. The parametric approximation procedure converts the contraction fixed-point problem into a nonlinear least squares problem. I show that this latter problem can be solved much more rapidly than standard methods based on discretization of state space.

In section 6, I estimate the model using a nested fixed point algorithm incorporating a parametric approximation method to solve the DP problem. The nested fixed point algorithm is a maximum likelihood estimation, in which outside of maximum likelihood estimation, the above nonlinear least square estimation (NLS) is performed to calculate fixed points and inside of maximum likelihood estimation, based on the NLS, to estimate unknown parameters. The estimation results support the observed stylized facts in general, allowing for a better understanding of replacement behavior of firms in the era of rapidly growing computer technology. I show the effectiveness of the parametric approximation method in comparison with a sample result from discretization. Finally, section 7 concludes and suggests some extensions for future research.

2 Summary of related literature

Rust (1987)'s seminal work on systems replacement provides a general template for approaching this topic. In this paper, he formulates a regenerative optimal stopping model for bus engine replacement to describe the behavior of the superintendent of maintenance at the Madison Metropolitan Bus Company.

In particular, Rust confirms that the superintendent's decision-making behavior on bus engine replacement is consistent with an optimal stopping rule. It is a strategy for deciding when to replace current bus engines, and is given as a function of observed and unobserved state variables. The optimal stopping rule is formulated as the solution to a stochastic dynamic programming problem that formalizes the trade-off between the conflicting objectives of minimizing maintenance costs and minimizing unexpected failures of bus engines.

This paper is important in at least two aspects. First, it provides a general framework that can be used to analyze replacement behavior in various fields. It is the first research that uses the so-called "bottom up approach" for modeling replacement investment. Second, the paper develops a nested fixed-point algorithm for estimating dynamic programming models of discrete choices. The algorithm is very useful in solving problems that arise typically in investigating replacement behavior. The results in the paper have been widely applied since its publication, and have been extended by many authors in numerous directions.

Despite its significant role in replacement research, this paper was not intended for computer systems. In contrast, there has been several articles related to the investment of computer systems, namely, Hendel (1996), Ito (1997), and Greenstein and Wade (1997). Hendel presents a multiple-discrete choice model for the analysis of differentiated products that are durable goods in a continuous process of technological change. This paper develops a model of PC purchasing behavior to deal with the main feature of PC demand, which is multiple-discreteness. The proposed model, along with a new data set on PC holding, permits demand estimation at the micro-level.

Ito (1997) presents an empirical investigation of the source of investment adjustment costs. Since mainframe computers are often the central pieces of hardware in business information systems, the author examines the dynamics of micro-level investment behavior in order to infer the size of implicit adjustment. She identifies the lumpiness of adjustment costs and concludes that the variation in adjustment costs arises due to the different degree of organizational friction in the investment processes of mainframe computers. She also finds that adjustment costs did not increase with the level of engineering adjustment activities, such as development of new software for new computer systems. Though Ito rightly points to the importance of adjustment cost in investment behavior, she pays little attention to the role of technology in the adjustment cost.

Greenstein and Wade (1997) investigate the product life cycle in the commercial mainframe market. In particular, they examine the entry and exit behavior of mainframe computers in the market using the hazard and Poisson models. The hazard model helps to estimate the probability of product exit and the Poisson model helps to estimate the probability of introduction. Additionally, this paper indicates many important market structures which may cause entry and exit of products, such as cannibalization, vintage and degree of competitiveness.

Also, there are several articles by Bresnahan, and Bresnahan and Greenstein (1997) which investigate the structural changes of mainframe computer market regarding to technological changes.

Unfortunately, previous research regarding systems replacement do not focus on the effect of technological development on replacement decisions in general nor on its effect on the replacement of computer systems. Moreover, the literature regarding the investment of computers also does not deal with replacement behaviors of computers. Based on the knowledge of the previous research, a new methodology should be introduced to explain and to understand computer systems replacement.

3 The Data

3.1 Summary of the Data

I received data from one of the biggest telecommunication companies in the world. It claims over 60 percent of the entire phone services market in a country. It also offers several other telecommunication services, such as cellular PCS (Personal Communications Service), internet, cable, and satellite communication services. The company has 864 hosts (including workstations) and about 39,000 PCs as of 1998. These hosts and PCs are spread out in 400 regional headquarters and regional offices.

The computer systems in the company can be divided into two parts according to use: (i) research use, and (ii) service and management use. Since computers for research use are purchased and replaced on project basis, they do not reflect technological depreciation. Thus, I only consider computer systems for only service and management use in the data for this paper. I also do not include the replacement of PCs in the company, since in PC replacement there is no upgrade activity and there only are block purchases and replacements.

The time frame of the data set starts from 1989 and ends on 1998. The data prior to 1989 are incomplete, though some computer systems have a history starting from earlier dates, such as 1977, 1979, 1983, and 1985. Within this time frame (89-98), I have a full history of upgrade and replacement for 123 computer systems in the company. The data consists of dates of introduction, purchase prices, specifications, dates of upgrades and replacements, prices of upgrades and replacements, details of replacements and upgrades, such as system specifications. The numbers of users for services provided by the firm also are acquired as a form of monthly data.

The price data of CPU, hard drive, memory, and other hardware are obtained from several computer databooks, online computer resources, and manufacturers' web sites.

3.2 Stylized Facts of the Data

I divided all computer systems in the sample into two categories in terms of the two different standards of CPU benchmarks, which are MIPS (Million Instructions per Second) and TPC (Transaction Processing Performance Council). Currently, the MIPS standard is in the process of being merged into the TPC standard, which includes the tpm (transactions per minute) and tps (transactions per second). However, since my data set consists of various computer systems and dates, it is very difficult to convert the MIPS standard into the TPC standard. Within each standard, I divide computer systems into different task groups. Once a certain system brand is designated to serve a given task, the later replacement is from the same or at least similar system brand. Table I illustrates the different groups of major tasks and number of systems in terms of the two CPU standards.

Table I

Computers included in the sample in terms of CPU standards

CPU standard	MIPS	TPC
Number*	48	57
Uses	Billing-Development	Business Info-Management
	Billing-Management	Customer Development
	General Management	Total Document
	New Customer Info-system	Pre-Billing
	Super High Speed Printer	Line-Management
		Material Information

*: number of computers in the sample

Table II[3]

Costs for three activities in the sample

Activity	Cost	MIPS	TPC
New purchase	Average	$572,919.9	$968,191.1
	Min	$41,917.7	$20,440.1
	Max	$4,893,545.6	$4,633,600.4
Replacement	Average	$1,082,499.4	$899,340.8
	Min	$16,752.8	$14,854.3
	Max	$7,160,791.3	$3,377,322.2
Upgrade	Average	$263,123.9	$435,181.4
	Min	$2,645.12	$3,251.5
	Max	$3,176,710.1	$2,283,130.1

Table II illustrates the average, minimum, and maximum costs of three activities, namely, new purchase, upgrade and replacement in terms of the two CPU standards.

As I expected, for both standards, the costs of upgrade are less than the costs of new purchase or replacement.

According the computer industry databooks, the cost per capacity decreases over time. For example, with a base year of 1982 as 100, the cost in 1998 is measured as 1. Based on this information, the firm has increased the capacities of computer systems tremendously, since the average price of replacement is the same or higher than the average costs of new purchases. This phenomenon can also be confirmed in the several databooks of the computer industry. That is, costs of high-end computer systems, such as mainframe computers in the market, have not decreased and have at time slightly increased over time.

Table III illustrates the intervals between upgrades and the intervals between replacements.

The first notable fact is that the intervals between replacements are much longer than those of upgrades. Second, the maximum intervals between replacement are 61 months and 53 months for MIPS and TPC standards respectively. This is because one of the major reasons for replacing a computer system is the age of the computer, which has an average 5 year life span for the company. In other words, internally regulated policy restricts the life-span of mainframe computers to a 5 year cycle. Figure 2 shows the replacement frequency of computer systems in the firm.

[3] The reason of big differences between minimum costs and maximum costs in the various activities is as follows: Since the data consist of various computer systesms, such as workstation, server, and mainframe computers. These varieties make the gaps between two costs much wided.

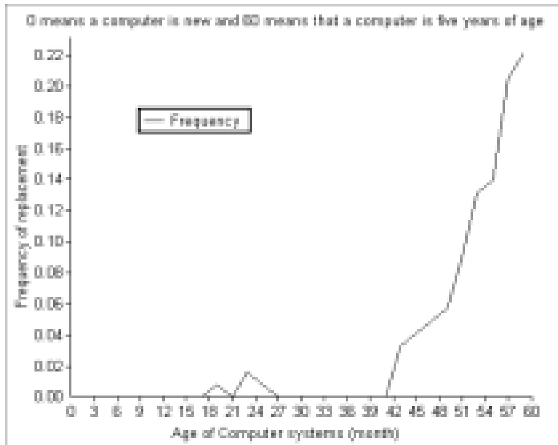

Figure 2: Frequency of replacements in terms of computers' age.

Table III

Intervals between upgrades and intervals between replacements

		MIPS	TPC
Interval between Replacement	Average	48.3	43.2
	Min	24	22
	Max	61	53
Interval between Upgrade	Average	22.1	16.7
	Min	7	10
	Max	39	32

All numbers are months

This reflects the fact that the computer systems become technologically obsolete after 5 years of use, even though not obsolete physically. Furthermore, this policy has been changed from 6 years to 5 years in recent years, which corresponds to the more rapid speed of technological development.

Due to the development of the computer industry in the 80's and 90's and the increases in demand for services, the intervals between the two subsequent actions[4] becomes shorter and shorter. Table IV (a), (b), and (c) illustrate several examples of the shortening of intervals in certain computer systems assigned to major tasks. One reason for shorter intervals is that the pace of development in the computer industry has become significantly faster and thus the current system becomes obsolete much more quickly. Figure 2.1 shows that cost per capacity has decreased

[4]Obviously, there are four combinations of actions: (i) upgrade-replacement; (ii) replacement-upgrade; (iii) upgrade-upgrade; (iv) replacement-replacement.

rapidly from 1994 to the current period. Second, the demand for the services provided by the company is growing tremendously. More frequent upgrades and replacements emerged in 1995, 1996 and 1997, when demand for services increased by greater amounts. However, from mid 1998, there was very little upgrade/replacement observed, since demand decreased significantly due to the economic depression. Figure 2.2 shows trend of total demand. The trends of average capacities are illustrated in figure 2.3 and 2.4.[5] Both figures 2.3 and 2.4 show that capacities increased rapidly from 1994 to 1998. when cost per capacity and demand changed rapidly. However, noticeably since the reduced amounts of cost per capacity is much larger than increased amounts of demand, the effect of cost per capacity on capacity increases seems to be much larger than that of demand.

Table IV (a)

Examples of Activities and brands of computers in various tasks

Task1			
Brand of system*	A	B	B
Region	1	2	3
New purchase→first action**	38 months	37 months	24 months
Interval 1st⟶2nd action	23 months	24 months	19 months
Interval 2nd⟶3rd action	20 months	11 months	22 months
Interval 3rd⟶4th action	17 months	8 months	13 months
Interval 4th⟶5th action	15 months	17 months	11 months
Interval 5th⟶6th action	12 months	11 months	12 months

*: A-Unisys system, B-Honeywell and Unisys system (MIPS standard)

**: Actions includes upgrade and replacement

(b)

Task 2				
Brand of system*	C	C	C	C
Region	1	2	3	4
New purchase → first action	27 months	18 months	18 months	22 months
Interval 1st ⟶ 2nd action	25 months	15 months	16 months	18 months
Interval 2nd ⟶ 3rd action	12 months	12 months	12 months	15 months
Interval 3rd ⟶ 4th action	18 months	15 months	11 months	12 months
Interval 4th ⟶ 5th action	12 months	13 months	12 months	10 months
Interval 5th ⟶ 6th action	11 months	1 0months	.	.

*: C-Tandem system (TPC standard)

[5] They show average capacities in terms of MIPS and TPC standards

(c)

Task 3				
Brand of system*	D	D	D	D
Region	1	2	3	4
New purchase → first action	59 months	36 months	42 months	51months
Interval 1st ⟶ 2nd action	23 months	35 months	38 months	34 months
Interval 2nd ⟶ 3rd action	20 months	15 months	13 months	10 months
Interval 3rd ⟶ 4th action	13 months	13 months	9 months	.

*: D-Toray and Fujitsu system (MIPS standard)

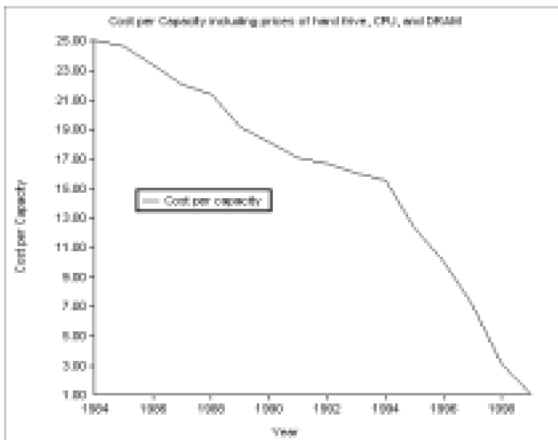

Figure 2.1: Average cost per capacity (All values are normalized)

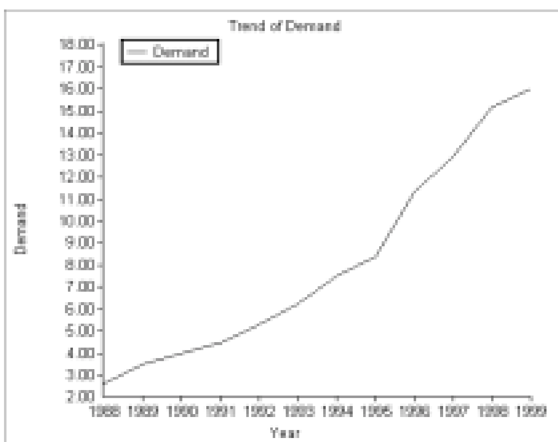

Figure 2.2: Trend of Total Demand (All values are normalized)

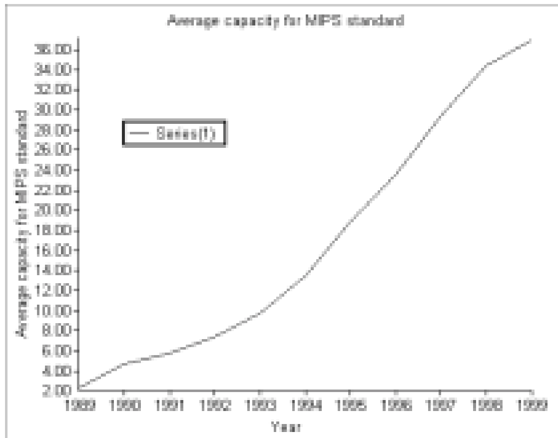

Figure 2.3: Trend of capacity increases (MIPS Standard)- All values are normalized

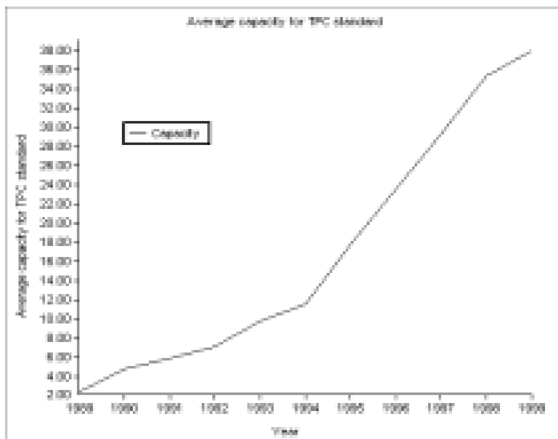

Figure 2.4: Trend of capacity increases (TPC Standard)- All values are normalized

Average frequency of upgrades for an individual computer system is 2.5 times. The maximum frequency of upgrades in turn is five times. This is because each computer has limited slots for upgrade. Once the upgrade slots are full, the system needs to be replaced to increase its capacity or to meet a growing demand. The average frequency of replacements at each task level is approximately two, though some tasks undergo three or more replacements. Also, there are some tasks which do not undergo any replacement.

4 The Model

I develop a stochastic dynamic programming model to investigate whether the aforementioned stylized facts of replacement and upgrade behaviors can be rationalized as an optimal strategy for the firm. My final objective is to explain the data by deriving a stochastic process $\{a_t, X_t\}$ with an associated likelihood function $l\,(a_1,..., X_1,...., \theta)$ formed from the solution to a particular optimal stopping problem.

To begin with, I explain a stochastic dynamic programming (DP) model. The stochastic DP model consists of a vector of state variables X_t, a control variable a_t, a profit function $\pi(X, a)$, a discount factor β, and a Markov transition density $p(X'|\, X,\, a)$, representing the stochastic law of motion for the states of computer systems. I assume that the state variable X_t can be partitioned into two components, $X_t = (x_t,\, \varepsilon_t)$, where x_t is an observed state vector and ε_t is an unobserved state vector. System administrators observe both components of X_t, but the econometrician observes only x_t. The system administrators weigh the consequences of various operating decisions given the states of various computer systems and attempts to perform the best actions. I assume that the result of this decision process can be summarized by a vector of current net benefits (or costs, if negative) corresponding to each operating decision.

4.1 Choice variables

Suppose that, at every month of the year, a system administrator investigates the status of each computer system and decides whether to upgrade, replace, or keep. Thus, the choice set is $A_t = \{-1,0,1\}$, where $(a_t = -1)$ is to keep the system unchanged, $(a_t = 0)$ is an upgrade, and $(a_t = 1)$ represents a replacement of system. When the choice is to replace, the system administrator needs to choose the capacity of the new system, i.e., there are n sub-choices of capacities, $Ca_1, ..., Ca_n$. Each Ca_i is a capacity choice for replacement.

The final choice set is as follows: $a : A_t = \{-1, 0, (Ca_1,..., Ca_n)\}$, i.e., keep $= -1$, upgrade $= 0$, and replace $= (Ca_1, ..., Ca_n)$.

4.2 State variables

I assume that two of the state variables are discrete, which are the capacity and the age of a current computer system. Two additional variables are continuous, being the demand for services and the cost per capacity in the computer market.

The state set in the model is $x : X_t = \{D_t,\, k_t,\, g_t,\, pc_t\}$, where D_t = demand for services, k_t = current capacity of the computer system, g_t = age of each computer system, pc_t = real cost per capacity, which can be seen as a market price of capacity. The two state variables g_t and k_t

explain internal states of computers and the remaining variables D_t and pc_t represent external states of computer systems.

The demand for services is assumed to follow an $AR(1)$ process, i.e., $\ln(D_t) = a + \rho \ln(D_{t-1}) + \mu_t$ with $\mu \tilde{} IID\ N(0.\varpi^2)$ and $|\rho| < 1$ for stationarity. Therefore, $\ln(D_t)$ is distributed as normal with mean $\frac{\alpha}{1-\rho}$ and variance $\frac{\varpi^2}{1-\rho^2}$. Demand D_t consists of the sum of the individual demands for services provided by each task. That is, total demand at time t, $D_t = \sum_j \xi_j d_{t,j}$, where $d_{t,j}$ is a demand for a task j at time t and $0 < \xi_j < 1$.[6]

The real cost per capacity, pc_t is bounded by zero. The pc_t evolves as follows:

$$pc_{t+1} = \left\{ \begin{array}{cc} (1 - \frac{\delta_t}{100})pc_t \text{ with } & 1- \ pb \\ pc_t \quad \text{with} & pb \end{array} \right\} \tag{1}$$

δ_t has a log normal distribution with mean μ and ν^2 with a range of $0 < \delta_t < 100$.

Therefore, we have the following probability. $P(pc_{t+1} \leq pc_t | pc_t) = pb \times I\{pc_{t+1} = p_t\} + (1 - pb) \times I\{pc_{t+1} < pc_t\} \times F(\delta_t)$.

The age variable, g_t represents the age of each computer system. Since the firm has the predetermined rule of replacement according to the age of each system, I intend to keep track of the age of each system.

4.3 Profits function

The profit function for a task j is as follows:

$$\pi(x_t, a_t, \theta_1) + \varepsilon(a_t) = \left(\begin{array}{c} R_d(d_{t,j},\ k_{t,j}, g_t, \theta_1) \\ -C_1(k_{t,j}, g_t, d_{t,j}, l, \theta_1, \varepsilon) \qquad \text{when } a_t = -1 \\ R_u(d_{t,j}, k_{t,j}, g_t,\ yp, \theta_1) \\ -C_2(k_{t,j}, g_t, yp, pc_t, \theta_1, \varepsilon) \quad \text{when } a_t = 0 \\ R_r(Ca_i, d_{t,j}, \theta_1) + scv(k_{t,j}, \theta_1) \\ -C_3(pc_t, Ca_i, \theta_1, \varepsilon) \qquad \text{when } a_t = \ Ca_i \end{array} \right) \tag{2}$$

where

$$C_1 = \left\{ \begin{array}{c} c_1(k_{t,j}, um(g_t), \theta_1 |\ k_{t,j} \geq d_{t,j}) + \varepsilon(-1) \\ c_1(l,\ k_{t,j},\ d_{t,j}, um(g_t), \theta_1 |\ k_{t,j} < d_{t,j}) + \varepsilon(-1) \end{array} \right\} \tag{3}$$

$$C_2 = c_2(yp, k_{t,j},\ pc_t, um(g_t), \theta_1) + \varepsilon(0) \tag{4}$$

[6] In order to calculate a fraction ξ_j for a demand d_t which a specific task serves, I sum up all capacities of computer systems and assume that a proportion for capacity of a system corresponds to a fraction of demand for a system.

$$C_3 = c_3(pc_t, Ca_i, \theta_1) + \varepsilon(Ca_i). \tag{5}$$

and θ_1 is a set of unknown parameters for profits function.

In the equation 2, $R_d(d_{t,j}, k_{t,j}, g_t, \theta_1)$ is a revenue function incurred by each task after choosing to keep. $R_u(d_{t,j}, k_{t,j}, g_t, \ yp, \theta_1)$ is a revenue function produced by each task after the choice of upgrading. $R_r(Ca_i, d_{t,j}, \theta_1)$ is a revenue function incurred by an action of replacement. In this model, as mentioned earlier, I assume that each task uses only one computer system and the purchase of additional computers as an alternative to replacement is prohibited. In the equations $R_u(d_{t,j}, k_{t,j}, g_t, \ yp, \theta_1)$, yp is an upgrade capacity.

scv is a value from a scrapped computer. The firm considers any scrapped computer systems to have no resale value. This is in fact not the case and these systems maintain a small resale value on the open resale market.

Equations 3, 4, and 5 show the cost structures of the model. C_1, C_2 and C_3 are the total costs of 'keep', 'upgrade', and 'replacement' respectively. c_1, c_2 and c_3 are the observable costs in the cases of 'keep', 'upgrade', and 'replacement'. Similar to the revenue functions, these three cost functions are costs of each task in the firm and the assumption of one task to one computer holds as well. In the cost for keeping, C_1, l is a unit labor charge per capacity in order to compensate the shortage of the current capacity. When demand exceeds the current capacity, the firm usually hires more labor to make up the shortage. In the costs for 'keep' and 'upgrade', $um(g_t)$ represents a unit maintenance cost, which increases with g_t. Once upgrade is chosen, the amount of yp will be added to the current capacity of computer systems.

I incorporate unobserved state variables $\varepsilon(a)$ by assuming that unobserved costs $\{\varepsilon(-1), \varepsilon(0), \varepsilon(Ca_i)\}$ follow a specific stochastic process, which will be described. $\varepsilon(-1)$ is an unobserved cost from keeping, such as managerial cost to prevent systems failures, cost for service contracts and some other tolerance costs from not replacing. $\varepsilon(0)$ is an unobserved cost from upgrade, such as service contract costs and some administrative costs. $\varepsilon(Ca_i)$ is also an unobserved cost when the action of replacement occurs, such as an opportunity cost for stopping services during replacement.

4.4 Dynamic Programming model

The optimal value function V_θ for each task is defined by

$$V_\theta(x,\varepsilon) = \max_{a\in A}[\pi(x_t,a,\theta_1) + \varepsilon_t(a) + \beta EV_\theta(x_t,\varepsilon_t,a)] \tag{6}$$

where $EV_\theta = \int_y \int_\eta V_\theta(y,\eta)p(dy,d\eta|x_t,\varepsilon_t,a,\theta_0)$

Then, as an optimal policy rule, a stationary decision rule is defined as

$$\mathsf{a}_t = z(x_t,\varepsilon_t,\theta) \tag{7}$$

where

$$z(x_t,\varepsilon_t,\theta) := \underset{a\in A(x_t)}{\text{argmax}}\ [\pi(x_t,\mathsf{a}_t,\theta) + \varepsilon_t(a) + \beta E\mathsf{V}_\theta(x_t,a_t,\varepsilon_t)] \tag{8}$$

and z $(x_t$ $,\varepsilon_t,$ $\theta)$ is the optimal control.

4.4.1 Markov transition probability

I follow Rust(1987) in making the standard simple assumption that the transition probability η can be factored as

$$\eta(\mathsf{x}_{t+1},\ \varepsilon_{t+1} \mid \mathsf{x}_t,\varepsilon_t,\ \mathsf{a}_t,\theta_0) = p(x_{t+1}|\ \mathsf{x}_t,\ \mathsf{a}_t,\ \theta_0)q(\varepsilon_{t+1}|\ \mathsf{x}_{t+1}), \tag{9}$$

where θ_0 is a vector of unknown parameters characterizing the transition probability for the observable part of the state variables. From the setup of choice variables, θ_0 is defined as follows. $\theta_0 = \{a,\rho,\mu,\nu,pb\}$.

Rust(1987) refers to the above equation as the "Conditional Independence Assumption (CI)", since the density of x_t is independent on ε_t, and ε_{t+1} is independent upon ε_t conditional on (x_t, a_t) as well.

In order to reach $p(x_{t+1}|x_t,a_t)$, I assume that all state variables are independent on one another. Therefore,

$p(x_{t+1}|x_t,a_t) = p(x^1_{t+1}|x^1_t,a_t)\times$ $p(x^2_{t+1}|x^2_t,a_t)$ $\times p(x^3_{t+1}|x^3_t,a_t)$ $\times p(x^4_{t+1}|x^4_t,a_t)$.

where $x^1_t = k_t$, $x^2_t = D_t$, $x^3_t = g_t$ and $x^4_t = pc_t$.

However, because of the assumption that deterministic evolutions of capacity and age variables depend on the choices, I can focus only on $p(D_{t+1}|D_t,a_t)$ and $p(pc_{t+1}|pc_t,a_t)$.

4.4.2 Policies of the actions

ε is assumed to have i.i.d multivariate extreme distribution, i.e.,

$$q(\varepsilon|X) = \Pi_{j\in A(X)} \exp\{-\varepsilon(j)\} \exp\{-\exp\{-\varepsilon(j)\}\}. \tag{10}$$

With this assumption of ε, we can rewrite V_θ in the equation 3.6 as follows:

$$V_\theta(x,a) = \{\pi(x,a,\theta) + \beta \int_y \sigma \log[Val1 + Val2] p(dy|x,a,\theta_0)\} \tag{11}$$

where $Val1 = \sum_{a'\in\{-1,0\}} \exp[(\pi(y,a',\theta_1) + \beta EV_\theta(y,a'))/\sigma]$,

$Val2 = \sum_{C'_i\in\{Ca_1\ldots Ca_n\}} \exp[(\pi(y,Ca'_i,\theta_1) + \beta EV_\theta(y,Ca'_i))/\sigma]$, and σ is a standard deviation of ε_t.

Then, conditional choice probabilities P(a$_t$| x, θ) are given by

$$\begin{aligned} \mathrm{P}(a &= -1,\ \text{keep}\ |x,\theta) = \\ &\frac{\exp\{R_d(x,\theta_1) - c_1(x,\theta_1) \quad + \beta EV_\theta(x,\ a=-1)\}}{Val1 + Val2} \end{aligned} \tag{12}$$

$$\begin{aligned} \mathrm{P}(a &= 0,\ \text{upgrade}\ |x,\theta) = \\ &\frac{\exp\{R_u(x,yp,\theta_1) - c_2(x,yp,\theta_1) \quad + \beta EV_\theta(x,\ a=0)\}}{Val1 + Val2} \end{aligned} \tag{13}$$

$$\begin{aligned} \mathrm{P}(a &= C_i,\ \text{replace}\ |x,\theta) = \\ &\frac{\exp\{R_r(x,Ca_i,\theta_1) + scv(k,\theta_1) - c_3\ (x,\theta_1) + \beta EV_\theta(x,\ a\ =\ Ca_i)\}}{Val1 + Val2} \end{aligned} \tag{14}$$

4.4.3 Log Likelihood Function

Then, following Rust(1987), we have the maximum log likelihood function at time t as follows:

$$l_t = \ln(P(a_t|x_t,\theta)) \tag{15}$$

and thus, we have

$$l(x_1, ..., x_T, a_1, ..., a_T | x_0, a_0, \theta) = \sum_{t=1}^{T} \ln(P(a_t | x_t, \theta)) \tag{16}$$

5 Parametric Approximation

The general method to solve the fixed point problem is a discretization of observed state variables. When the observed state variable is continuous, the required fixed point is in fact an infinite dimensional object. Therefore, in order to solve the fixed point problem, it is necessary to discretize the state space so that the state variable takes on only finitely many values. But there are limits regarding this method: (i) "curse of dimensionality"; (ii) the limits it places on our ability to solve high-dimensional DP problems. Despite these limits, this method have been used in many literature.

The discretization method may not be applicable to computer replacement research to solve the fixed point problem, because of the aforementioned problems. The details are in the following:

5.1 An attempt of discretization of the state variables

The most conservative dimension of a possible combination of state variables resulting from discretization in the proposed model is 540,000. Discrete variables, capacity and age are discretized as follows. First, I discretize the age variable, g_t, into bimonthly cycle, even though I have monthly data. Thus, age 1 represents a new computer,[7] and an absorbing state 30 means 5 years of age.[8]

Second, regarding the capacity level, the current data set of the capacity consists of the three elements of CPU, hard drive and memory size. In order to concretize and transform the capacities into actual numbers which can represent the capacity of each computer system, I take a weighted average of these three elements. Since CPU is the most important factor in the capacity of computer systems, I give it a weight of 0.5. On the other hand, I give equal weights to Hard Drive and Memory size, namely 0.25. At this time, I do not separate the capacity into the two standards of CPU benchmark, TPC and MIPS. Even though the weights were confirmed with the system administrators in the firm, their appropriateness will be verified in further research. With transformed capacities of computer systems, I discretize the capacity from 1 to 40. The

[7] literally 2 months old.

[8] For estimation purpose, I discretize the age variables into months instead of bi-monthly cycle.

last state 40 is the absorbing state. Difference between each step is 30. Therefore, 1 represents (1,...,30), and 2 represents (31,...,60), and 40 represents the range, (1171,...,1200). These two discrete variables should be discretized regardless of the parametric approximation.

The continuous variables, demand and cost per capacity, can be discretized as follows. First, I discretize demand from 1 to 30. Like the actual capacity, the last state 30 is the absorbing state. Demand 1 represents 100,000 to 105,000 users and the absorbing state 30 is from 245,001 to 250,000 users.

Second, I discretize the cost per capacity into 15 possible costs such as $\{15, 14, ..., 1\}$. Difference between subsequent prices is a 20% price drop. I restrict maximum price drops in one period to just 2 steps. These assumptions are based on several research data, computer industry databooks, and Moore's Law [9].

The transition probability matrices, $p(D_{t+1}|D_t)$ and $p(pc_{t+1}|pc_t)$ are in the appendix.

Therefore, the resulting dimension from the discretization is $540,000 = 30 \times 40 \times 30 \times 15$.

5.2 Computational Burden

First, solving the fixed point problem requires calculation of the expected value function. That is, $EV_\theta = \int_y \int_\eta V_\theta(y, \eta) p(dy, d\eta | x_t, \varepsilon_t, a, \theta_0)$. Even though the Markov transition probability from discretization is a sparse matrix, it still requires extensive time to calculate expectation of value function. Second, the polyalgorithm method by Rust (1987) takes advantage of the complimentary behavior of the two iterations, which are a combination of contraction iteration and policy iteration[10] This algorithm enjoys a substantial reduction in time calculating the fixed point. However, it is not applicable to solving a dynamic programing model. The reason is as follows: One must have a Frechet derivative $(I - T_\theta^{'})$ in order to use policy iteration method[11]. But, the dimensionality problem makes it impossible to get the derivatives of T_θ. Thus, the algorithm for the DP problem consists solely of a backward induction, which is simple but takes more time to

[9] The index was created by the informations from SIA (Semiconductor Industry Association)'s annual databooks, the 8th Annual Computer Industry Almanac, ZDnet.com, and Cnet.com.

[10] Newton Kantorovich method.

[11] The idea of the policy iteration method, i.e., the Newton Kantorovich iteration is to find a zero solution of the nonlinear operator $F = (I - T_\theta)$ instead of finding a fixed point $EV_\theta = T_\theta(EV_\theta)$. With invertibility of $(I - T_\theta)$ and existence of a Frechet derivative $(I - T_\theta^{0})$, one can do a following Taylor expansion:

$$0 = [I - T_\theta](EV_l)\tilde{}[I - T_\theta](EV_{l-1}) + [I - T_\theta'](EV_l - EV_{l-1}).$$

$$\Longrightarrow EV_l = EV_{l-1} - [I - T_\theta']^{-1}(I - T_\theta)(EV_{l-1}).$$

solve. Therefore, the extended time caused by the two aforementioned reasons seriously affects the calculation time of a nested fixed point algorithm, because the nested fixed point algorithm uses the fixed point algorithm outside of the maximum likelihood estimation.

5.3 Parametric Approximation

First, one needs functional forms for the three value functions, keep, upgrade, and replacement. To find the parametric forms of value functions, I use the simple linear OLS estimations, such as

$$\begin{aligned} V(a &= -1, x) = f(x, \lambda_{-1}) + \psi_{-1} \\ V(a &= -0, x) = f(x, \lambda_0) + \psi_0 \\ V(a &= Ca_i, x) = f(x, \lambda_{Ca}) + \psi_{Ca_i} \end{aligned} \tag{17}$$

I choose the best functional forms for each value function according to the criteria, $\overline{R}^2$. After extended search for the appropriate functional forms of the three value functions, I have the following results.$V(a = -1, x)$ has 12 parameters($= \lambda_{-1}$) with 0.983 of $\overline{R}^2$, $V(a = 0, x)$ has 15 parameters($= \lambda_0$) with 0.962 of $\overline{R}^2$ and $V(a = (Ca_1 ... Ca_n), x)$ has 18 parameters($= \lambda_{Ca}$) with 0.962 of $\overline{R}^2$. Therefore, we have $f(x, \lambda_{-1})$ u $\sum_{i=1}^{12} \lambda_{-1}^i \vartheta_{-1,i}(x)$, $f(x, \lambda_0)$ u $\sum_{i=1}^{15} \lambda_0^i \vartheta_{0,i}(x)$, and $f(x, \lambda_{Ca})$ u $\sum_{i=1}^{18} \lambda_{Ca}^i \vartheta_{Ca,i}(x(Ca))$.

With the approximated functional forms of the three value functions, I estimate all 45 parameters $(\lambda_{-1}, \lambda_0, \lambda_c)$ with nonlinear least square estimation, such as

$$\min_{\lambda_{-1}, \lambda_0, \lambda_c} \sum_j \sum_a [V_a(x_j) - U_a]^2 \tag{18}$$

where,

$U_1 = [(\{u(x_t, a_t = -1, \theta_1)$

$+\beta \int_y \sigma \log \left(\sum_{a' \in A(y)} \exp[V_{\alpha'}(y)/\sigma] \right) p(dy | x_t, a_t = -1, \theta_0)\})]$

and

$U_2 = [(\{u(x_t, a = 0, \theta_1)$

$+\beta \int_y \sigma \log \left(\sum_{a' \in A(y)} \exp[V_{\alpha'}(y)/\sigma] \right) p(dy | x_t, a_t = 0, \theta_0)\})]$

and

$U_3 = [(\{u(x_t, a = Ca_i))$

$+\beta \int_y \sigma \log \left(\sum_{a' \in A(y)} \exp[V_{\alpha'}(y)/\sigma] \right) p(dy|x_t, a_t = Ca_i, \theta_0)\})]$.[12]

Solving the above minimization problem enables us to estimate all parameters $\hat{\lambda}_{-1}\hat{\lambda}_0$, and $\hat{\lambda}_c$.In fact, a parametric approximation procedure converts the contraction fixed-point problem into a nonlinear least squares problem.

6 Estimation

Incorporating the parametric approximation, the estimation requires the nested fixed point algorithm, which is intended to find parameters that maximize the likelihood functions, subject to the constraint that function EV_θ is the unique fixed point.

Table V

Quadratic functional forms used in the model

Revenue	Specifications[13]
Keep	$\theta_{111} + \theta_{112} f_1(k_{t,j}, g_t, rm^*) \times d_{t,j} + \theta_{113}(f_1(k_{t,j}, g_t, rm) \times d_{t,j})^2$
Upgrade	$\theta_{111} + \alpha_{112} f_1(k_{t,j}, g_t, rm) \times d_{t,j} + \theta_{113}(f_1(k_{t,j}, g_t, rm) \times d_{t,j})^2$
	$+\theta_{121}(yp \times rm \times d_{t,j}) + \theta_{122}(yp \times rm \times d_{t,j})^2$
Replacement	$\theta_{131} + \alpha_{132}(Ca_i \times rm \times d_{t,j}) + \theta_{133}(Ca_i \times rm \times d_{t,j})^2$.
Scrap	$\theta_{141} + \theta_{142}((k_{t,j} \times rm) \times d_{t,j}) + \theta_{143}((k_{t,j} \times rm) \times d_{t,j})^2$
Cost	
Keep	$I(k_{t,j} \geq d_{t,j})\{\theta_{151} + \theta_{152}(k_{t,j} \times um_t) + \theta_{153}(k_{t,j} \times um_t)^2\}$
	$+I(k_{t,j} < d_{t,j})\{\theta_{151} + \theta_{152}(k_{t,j} \times um_t) + \theta_{153}(k_{t,j} \times um_t)^2$
	$+\theta_{154}(l \times (d_{t,j} - k_t)) + \theta_{155}(l \times (d_{t,j} - k_t))^2\}$
Upgrade	$\theta_{151} + \theta_{152}(k_{t,j} \times um_t) + \theta_{153}(k_{t,j} \times um_t)^2 + \theta_{161}(yp \times (pc_t \times cp^{**}))$
	$+\theta_{162}(yp \times (pc_t \times cp))^2$
Replacement	$\theta_{171} + \theta_{172}((pc_t \times cp) \times Ca_i) + \theta_{173}((pc_t \times cp) \times Ca_i)$

*: rm is a scale parameter for revenue functions

**: cp is a scale parameter for cost functions

$f_1(k_{t,j}, g_t, rm)$ is a simple function which is increasing in $k_{t,j}$, decreasing in g_t, such as

$$f_1(k_{t,j}, g_t, rm) = (rm \times k_{t,j})/\sqrt{g_t}$$

and $um_t = \sqrt{g_t}$.

One of the benefits of a parametric approximation is that discretization of continuous state variables is no longer required. The two discrete state variables are still discretized in the manner

[12] The above three expectations are calculated by a quadrature method

[13] The assumptions imposed on these functional forms, $f_1(k_t, g_t, rm)$ and um_t are due to the large number of unknown parameters. These assumptions can be released in further research.

suggested in section 5. Additionally, the age variable is discretized in much finer dimension. It is discretized into months instead of bimonthly cycle. That is, the age variable ranges from 1 to 60, where the absorbing state, 60 represents that computer system is five years of age.

The functional forms for revenue and cost equations are shown in the table V.

$f_1(k_{t,j}, g_t, rm)$ illustrates how capacity contributes to the revenue functions. For example, the contributions of capacity will decline as the computer gets old. However, in the revenue function for replacement, the replacement capacity Ca_i will fully contribute to R_r.

6.1 Maximum Likelihood Estimation

The estimation procedure is that, outside of maximum likelihood estimation, the above nonlinear least square estimation (NLS) is performed and fixed points EV_θ is calculated. Based on the fixed points, the maximum likelihood estimation is performed.[14]

The log likelihood function in this model is as follows:

$$l_f(x_1, ...x_T, a_1, ...a_T | x_0, a_0, \theta) = \sum_{t=1}^{T} \ln(P(a_t | x_t, \theta)) \tag{19}$$

6.1.1 Results of Estimation

Parameters for demand and cost per capacity For simplicity, I estimate the parameters $\theta_0 = \{a, \rho, \mu, \nu, pb\}$ which govern the transition probabilities for demand and cost per capacity separately from the parameters of profits function. First, as I mentioned, total demand D_t equals the sum of an individual demand for a task j, such as $D_t = \sum_j \xi_j d_{j,t}$. In order to calculate a fraction ξ_j for a demand d_t which a specific task serves, I sum up all capacities of computer systems[15] and assume that a proportion for capacity of a system corresponds to a fraction of demand for a system. Table VI shows the estimation result.

[14] The Berndt, Hall, Hausman, and Hall (BHHH) alogorithm is used, along with numerical derivatives.
[15] Note that a task uses only one computer system.

Table VI

Parameter estimates for state D_t

Parameter	D_t
	Estimate
α	14.023 (0.348)
ρ	0.9793 (0.058)

(standard errors in parentheses)

The parameters of the cost per capacity, pc_t are obtained through calibration method. The calibration results are as follows: $pb = 0.76$, $\mu = 9.25$, and $\nu = 8.82$.

Structural estimates of revenue and cost functions

Table VII (a)

Structural parameters (θ_1) estimates

Parameters	MIPS		TPC	
β	0.999			
Revenue	Estimate	Std.Err	Estimate	Std.Err
θ_{111}	15.209	(2.455)	14.031	(2.165)
θ_{112}	3.276	(0,012)	3.284	(0.891)
θ_{113}	3.213	(0.347)	2.762	(0.925)
θ_{121}	3.774	(0.238)	3.256*	(3.001)
θ_{122}	2.341	(0.019)	2.319	(0,152)
θ_{131}	6.133	(1.721)	5.983	(1.327)
θ_{132}	1.482	(0.003)	1.351	(0.026)
θ_{133}	1.695	(0.072)	1.771	(0.344)
θ_{141}	12.561	(2.093)	12.385	(3.102)
θ_{142}	1.709*	(1.457)	1.983*	(2.013)
θ_{143}	0.391*	(1,014)	0.652*	(1.320)

Continued in Table VII (b).

* Not significant at the 95% level.

Table VII (a) and (b) report the structural parameter estimates computed by maximizing the likelihood function l_f in equation (19) using the nested fixed point algorithm. I present structural estimates for the unknown parameters for the quadratic specifications suggested in Table V.[16] The estimation results for $\beta = 0.999$ corresponds to a dynamic model in which the present value of current and future profit streams is maximized by the investment decisions of the firm.

Most parameters of revenue and cost functions are precise and have the expected sign.

[16] When I tried the estimation additionally with $\beta = 0.99$ and $\beta = 0.95$, there was no distinguishable difference.

In table VII (a), parameters (except the constant term) of the revenue function for scrapped computers are insignificant at the 95% level. This may be caused by two possible reasons. First, the proposed functional form is misspecified. Second, any scrapped computer has a fixed value regardless of its remaining capacities. In table VII (b), only one revenue parameter and two cost parameters are reported as insignificant. However, they show the correct sign.

Table VII (b)

Structural parameters (θ_1) estimates

Parameters	MIPS		TPC	
β	0.999			
Cost	Estimate	Std.Err	Estimate	Std.Err
θ_{151}	7.345	(1.134)	6.753	(1.941)
θ_{152}	1.673	(0.021)	1.594	(0.515)
θ_{153}	0.831	(0.048)	0.902	(0.032)
θ_{154}	0.875	(0.103)	0.983	(0.121)
θ_{155}	1.342*	(2.355)	1.084	(0.264)
θ_{161}	2.151	(0.135)	2.757	(0.313)
θ_{162}	3.385	(0.251)	1.349*	(1.217)
θ_{171}	11.673	(2.857)	9.967	(2.522)
θ_{172}	6.783	(1.486)	4.398	(1.924)
θ_{173}	8.317	(2.334)	7.964	(1.355)
l	5.245 (1.267)			
Likelihood	-5969.68		-6226.93	
Obs. Size	5760		6840	

The scale parameters for revenue and cost functions, rm and cp are calibrated from the data: $rm = 20$ and $cp = 12$.

* Not significant at the 95% level.

Figures 8.1, 8.2, 8.3, and 8.4 show the three policies (keep, upgrade, and replace) and their expected value functions, plotted against various cost per capacity, in the case when demand is lower than the current capacity with age fixed. In the figures 8.1 and 8.2, as cost per capacity increases, the value function of keeping does not decrease. The value functions of upgrade fall slightly, as cost per capacity increases due to the amount of upgrade. However, since replacement requires change of the current system as a whole, the cost of replacement will increase tremendously, as cost per capacity increases. Thus, the likelihood of replacement falls and reaches zero eventually, as cost per capacity increases. The best choice for keeping up with the current demand becomes the choice of upgrade, when cost per capacity is high enough. Figures 8.3 and 8.4 show where the condition is identical to figures 8.1 and 8.2 with exception of the age variable. As the computer gets older, replacement becomes more profitable to upgrade. However, as cost per capacity increases, the probability of replacement falls and the probability of upgrade becomes the best choice.

Figures 8.5, 8.6, 8.7, and 8.8 show how the three polices (keep, upgrade, and replace) and three value functions of old computer system depend on various demands, when capacity and cost per capacity are fixed. As demand increases, the value functions for keep, upgrade and replacement are expressed in smooth increasing curves. However, each policy shows different behavior. Until the points where the capacity is slightly over the current demand, choice of keep is more likely to occur with decreasing tendency. But, beyond the point of the demand, if the system is relatively new, upgrade should be more likely to occur for maximizing profits. However, since the system is relatively old in this case, the choice of replacement outperforms the choice of upgrade and thus, replacement is more likely to occur. Comparison with figures 8.7 and 8.8 show how cost per capacity affects the above situation. When cost per capacity becomes higher (the figures 8.8 and 8.9), upgrade becomes a more reasonable choice than replacement. However, this situation will change, when the demand is much bigger than the current capacity.

Policy for replacement capacity. Figure 8.9, 8.10, 8.11, and 8.12 illustrate how replacement capacity should be chosen depending on future cost and demand, when replacement is considered as an optimal strategy.

Figures 8.9 and 8.10 illustrate effects of capacity choices on expected value functions of replacement according to two cases of demand, high and low. Two figures are plotted against cost per capacity variable. Figure 8.9 is based on the situation of high demand and figure 8.10 illustrates a low demand situation. When demand is small enough, there is a little change among expected value functions of replacement with various capacity choices (figure 8.9). However, when demand is large, the situation changes. Differences among value functions become larger. Value function with large capacity choice falls rapidly (figure 8.10). This situation suggests that when replacement capacity is decided, future expected demand should be considered. The intuition is that, when demand is expected to be large in the future, the firm should increase its capacity choice of replacement.

Figures 8.11 and 8.12 shows how capacity choices affect expected value functions of replacement subject to two cases of cost per capacity, low and high. Two figures are plotted against demand variable. Figure 8.11 shows the case of low cost per capacity and figure 8.11 shows the opposite case. Comparison between figures 8.11 and 8.12 reveals the fact that when cost per capacity is relatively high, increases of capacity choice raise expected values with decreasing rate, because high cost per capacity increases replacement costs more than low cost per capacity does. In contrast, in case of low cost per capacity, increasing capacity choice raises expected values with increasing rate. This fact suggests that when cost per capacity is expected to be high, relatively low capacity is more preferable to high capacity choice.

Simulation based on Estimation results Based on estimated parameters, a simple simulation is performed to generate simulated data to be compared with real data. Instead of simulating the life of a task, I simulate the life of a computer system. The actual decision process is assumed to have randomness, i.e.,. the actual decision varies, even though there is a most probable choice among the three options at each period.

Figures 8.13 and 8.14 present three simulated policies in two different situations. Figure 8.13 illustrates the situation in which the cost per capacity decreases rapidly with relatively small starting capacity and demand. Figure 8.14 shows the situation in which the cost per capacity decreases relatively slowly with a large demand for capacity.

The differences between two figures 8.13 and 8.14 are as follows: The former figure shows relatively higher likelihoods of keep and replacement than those of figure 8.14. This is because small capacity requires relatively small maintenance costs. Also, as the computer gets older, replacement will be more profitable than upgrade, due to relatively small cost per capacity. In contrast to the figure 8.13, the situation is different in figure 8.14. In the initial phase, keeping is the proper choice. But, the likelihood of keeping is higher than that of figure 8.13, because large capacity means there is no need for upgrade and replacement. Moreover, upgrade is more profitable than replacement over time in the latter figure, due to the relatively high cost per capacity.

Figure 8.15 illustrates the comparison between the simulated data and the actual data in terms of frequency of replacement with various age of computers. In general, the shape and tendency of replacement are similar to each other. Most replacement activities occur in the range of four and five years of age. Even though frequency of the simulated data is slightly larger than that of the actual data, the difference is minimal and acceptable. In general, the simulated data from the estimated parameters show more frequent tendency to replace computer system than the actual data do.

Comparison between parametric approximation and discretization Based on the parameters in table VI, VII (a), and VII (b), I calculate a fixed point by a discretization method. Comparisons between two value functions from discretization and parametric approximation are illustrated in figures 3, 4, and 5, which represent cases of keep, upgrade, and replacement, respectively. In each figure, graph (a) presents the expected value function[17] by discretization, and graph (b) shows the differences between two value functions from parametric approximation and discretization. Though there is a slight discrepancy in comparisons between two methods, the differences seem to be negligible.

7 Conclusion and further research

7.1 Conclusion

The paper presents an empirical model to analyze a system administrator's forward looking behavior regarding computer system replacement. Taking into account the unique features of computer systems, especially technical depreciation, a DP model is proposed to explain the decision behavior. The stylized facts from the actual data provide evidence for the observation that the firm's replacement and upgrade behaviors reflect the technical depreciation.

In the empirical part of the paper, the proposed model is applied to the actual data on computer holdings of one of the world's largest telecommunication companies. First, I show the inappropriateness of the discretization method for the model. Second, in the estimation part, the paper utilizes a series of estimation techniques, such as a parametric estimation and a nested fixed point algorithm. The parametric approximation is used to circumvent problems, such as curse of dimensionality and computational burden incurred by discretization and substitutes a non-linear least squares estimation method for a fixed point iteration. The speed up in solution time is sufficiently large to make it feasible to estimate the unknown parameters of the model by maximum likelihood estimation method.

[17] In order to graph value functions in terms of the demand and the capacity variables, I fix the cost per capacity and the age of computer at certain values, such as relatively high cost per capacity and a fairly new computer.

The estimation results confirms that the proposed model can explain replacement and upgrade behavior reasonably. The estimation results confirms that the proposed model can explain replacement and upgrade behavior reasonably. Demand variable also play an important role in the firm's decision behavior. The simulated data also confirm the model by comparison with the actual data. As a result, the firm seems not to use an arbitrary rule of thumb in deciding to upgrade and replace its mainframe computers so rapidly, but rather the firm appears to have a very sophisticated understanding of the impact of technological progress resulting from Moore's Law and is taking advantage of this progress to significantly reduce its operating costs and provides better service to its customers.

The comparisons between sample results of discretization and those of parametric approximation method show the effectiveness of the parametric approximation in general.

7.2 Further Research

First, since I only examined the model based on the assumption of the "dynamic aspect", I want to investigate whether a system administrator decides his decision based on "myopic model" or "dynamic model". Myopic model requires us to calculate the model with $\beta = 0$, in which only current operating profits are considered. I will also test the result of $\beta = 0$ with that of $\beta = 0.999$.

Second, various specifications of revenue and cost functions should be tried to reach the best-fit model to the actual data. Additional assumptions imposed on the functional forms of unit maintenance cost, um_t, and capacity realization, $f_1(k_t, g_t, rm)$ should also be released in future research.

Third, I want to estimate unknown parameters of state variables by using full likelihood function, such as $\sum_{t=2}^{T} \ln(P(a_t|x_t, \theta_1)) + \sum_{t=2}^{T} \ln(P(x_t|x_{t-1}, \theta_0))$. This procedure will enable us to compare a result from the full likelihood function with the result of the current estimation.

Fourth, the simulated data generally show a more frequent tendency of replacement than the actual data show. Therefore, this situation should be examined in future research.

Fifth, introduction of new software, such as a new operating system, is one of main reasons to replace computers, since a new operating system may require a more advanced system to work properly. Even though I am fully aware of importance of role of new software, limits of information regrading software which the company uses prevent me from including this factor as a state variable of the model. In future research, this should be considered as well.

Sixth, Since replacement of PCs is also significant decision problem in the current economy, I will try to rationalize PC replacement behavior in future research. In order to deal with block purchases and replacements, which are the main characteristic of PC holding, I should gather more data and make a different model to explain it.

References

[1] Biglaiser, Gary. and Giordan, Michael. (1998): "Dynamics of Price Regulation," Preliminary notes.

[2] Berndt, E., J. Hausman, B. Hall, and R. Hall (1976): "Estimation and Inference in Nonlinear Structural Models," *Annals of Economic and Social Measurement*, 3,653-665.

[3] Billingsley, Patrick.(1979): *Probability and Measure*, John Wiley, New York, pp. 309-310; p. 320.

[4] Bresnahan, Timothy F. (1986): "Measuring the Spillovers form Technical Advance: Mainframe Computers in Financial Services," *The American Economic Review*, 1986, 76, 4, 742-755.

[5] Bresnahan, Timothy F.(1998): "The Changing Structure of Innovation in Computing: Sources and Threats to the Dominant U.S. Position." July, 21 1998.

[6] Bresnahan, Timothy F and Green Shane. (1997): "Technological Competition and the Structure of the Computer Industry." December,14 1997.

[7] Gandal, Neil., Green, Shane. and Salant David.(1998): "Adoptions and Orphans in the Early Microcomputer Market," *forthcoming Journal of Industrial Economics*,

[8] Greenstein, Shane M. and Wade, James B(1997): "Dynamic Modeling of The Product Life Cycle in The Commercial Mainframe Computer Market, 1968-1982," *Working paper 6124, NBER*, August 1997.

[9] Hall, George. and Rust, John. (1999): "An Empirical Model of Inventory Investment by Durable Commodity," Intermediaries, Department of Economics, Yale University, 1999.

[10] Heckman, J. J. (1981): "Statistical Models for Discrete Panel Data," in *Structural Analysis of Discrete Data with Econometric Applications*, edited by C. Manski and D. McFadden. Cambridge: M.I.T. Press.

[11] Hendel, Igal. (1996): "Estimating Multiple-Discrete Choice Models: An Application to Computerization Returns," Department of Economics, Princeton University.

[12] Hernandez-Lerma, Onesimo. and Lasserre, Jean. B, (1996): Discrete-Time Markov Control Processes: Basic Optimality Criteria, Springer, New York.

[13] Ito, Harumi. (1997): "The Structure of Adjustment Costs in Mainframe Computer Investment," Department of Economics, Brown University. 1997.2.

[14] Jorgenson, Dale W, and Landau, Ralph. (1989): "Technology and Capital Formation," The MIT Press, 1989.

[15] Kennet, D. Mark. (1994): "A Structural Model of Aircraft engine Maintenance," Journal of Applied Econometrics, 1994, 9, 351-68.

[16] Khanna, Tarun.(1993): "Racing Behavior: Technological Evoluction in the High-End Computer Industry." September 1993.

[17] McFadden, Daniel.(1973): "Conditional Logit Analysis of Qualitative Choice Behavior" in Frontiers of Economics, ed. by P. Zarembka. New York: Academic Press.

[18] McFadden, Daniel. (1981): "Econometric Models of Probabilistic Choice," in Structural Analysis of Discrete Data, ed. by C. Manski and D. McFadeden. Cambridge: M.I.T Press.

[19] Montemerlo, Michael S. and Love, J. Christopher and Opiteck, Gregory J. and Gordon, David Goldhaber. and Ellenbogen, James C.(1996): "Technologies and Designs for Electronic Nanocomputers," MITRE MSR Program, July 1996.

[20] Perska-Juliussen, Karen and Juliussen Egil. (1997): "The 8th Annual Computer Industry Almanac,". Computer Industry Almanac Inc.

[21] Rothwell, Geoffrey. and Rust, John.(1997): "On the Optimal Lifetime of Nuclear Power Plants," Journal of Business & Economic Statistics, April, 1997,15, No. 2, 195-208.

[22] Rust, John. (1985): "Stationary Equilibrium in a Market for Durable Assets," Econometrica, 1985, 53,733-806.

[23] Rust, John. (1986): "When is it Optimal to kill off the Market for Used Durable Goods," Econometrica, 1986,54,65-86.

[24] Rust, John. (1987): "Optimal Replacement of GMC Bus engine: An Empirical Model of Harold Zurcher," Econometrica, 1987, 55, 999-1033.

[25] Rust, John. (1988): "Maximum Likelihood Estimation of Discrete Control Processes," SIAM Journal of Control and Optimization, 26, 1006-1024

[26] Rust, John. (1988b): "A Dynamic Programming Model of Retirement Behavior," in The Economics of Aging, D. Wise, ed., National Bureau of Economic Research, University of Chicago Press, IL.

[27] Rust, John. (1995): "Nested Fixed Point Algorithm Document Manual," v5, November, 1995.

[28] SIA Annual databook, (1999) : Semiconductor Industry Association.

8 Appendix

8.1 Transition matrices of demand and cost per capacity for discretization purpose.

$p(D_{t+1}|D_t)$ is as follows.

	1	2	3	...	29	30
1	h_0^*	$1-h_0$	0	...	0	0
2	h_1	h_0	$1-h_0-h_1$	...	0	0
3	0	h_2	h_0	...	0	0
...	0	0	0	...	...	...
29	0	0	...	0	h_0	$1-h_0$
30	0	0	...	0	0	1

*: $h_0 = 0.89$.

$p(pc_{t+1}|pc_t)$ is as follows.

	15	14	13	...	2	1
15	g_0^*	g_1^{**}	$1-g_0-g_1$	...	0	0
14	0	g_0	g_1	$1-g_0-g_1$	0	0
13	0	0	g_0	...	0	0
...	...	...	0	...	0	0
2	0	0	...	....	g_0	$1-g_0$
1	0	0	...	0	0	1

*: $g_0 = 0.28$, **: $g_1 = 0.7$

8.2 Graphs of estimation and simulation results.

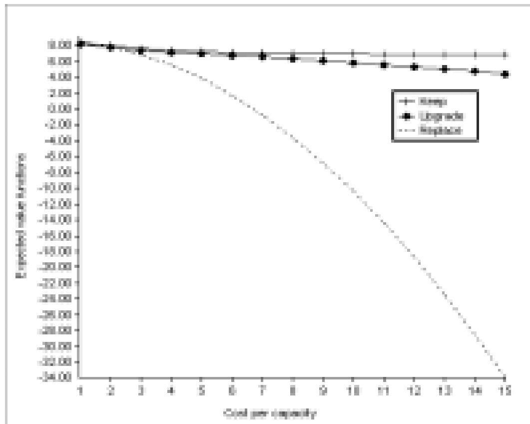

Figure 8.1:Expected value functions of keep, upgrade, and replacement

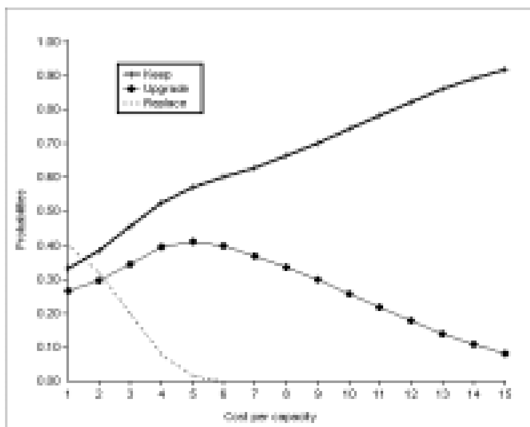

Figure 8.2: Three policy rules with various cost per capacity

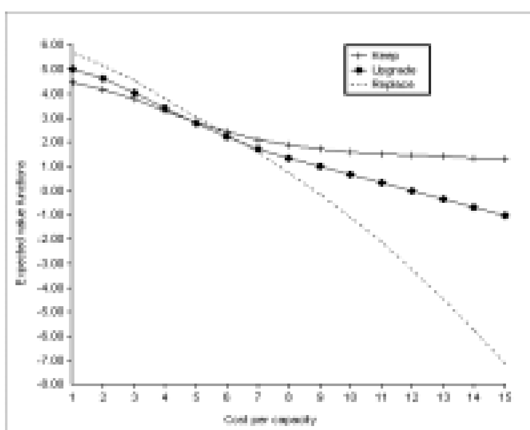

Figure 8.3: Expected value functions of keep, upgrade, and replacement:

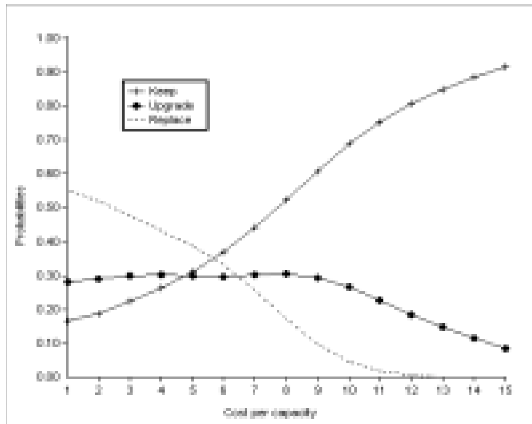

Figure 8.4: Three policy rules with various cost per capacity

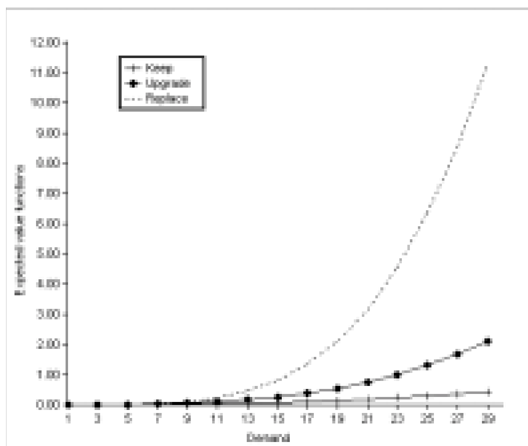

Figure 8.5: Expected value functions of keep, upgrade, and replacement:

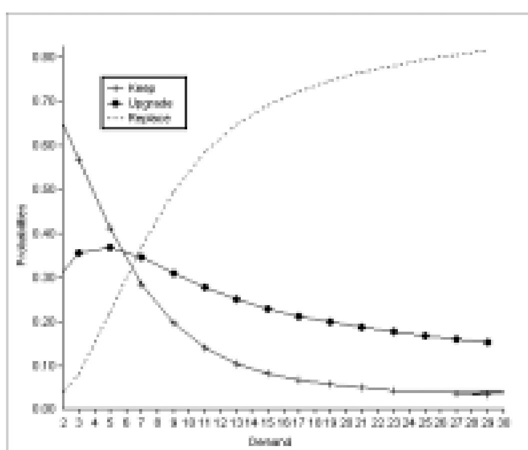

Figure 8.6: Three policy rules with various demand

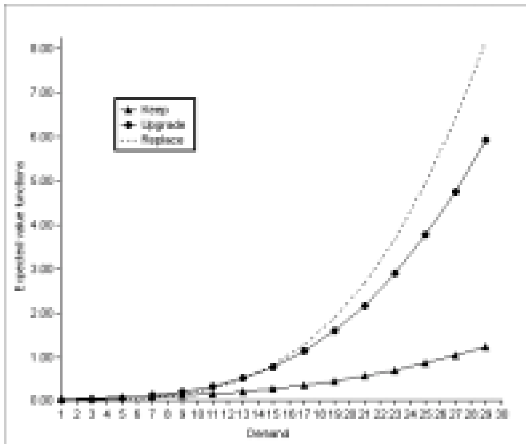

Figure 8.7: Expected value functions of keep, upgrade, and replacement:

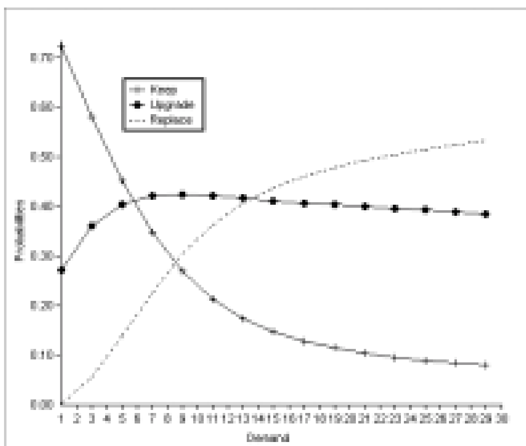

Figure 8.8: Three policy rules with various demand

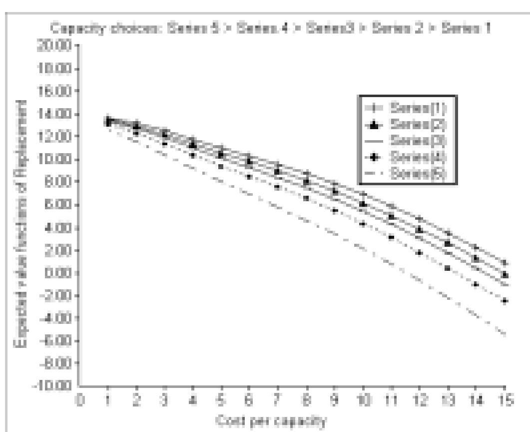

Figure 8.9: Expected value functions with various cost per capacity

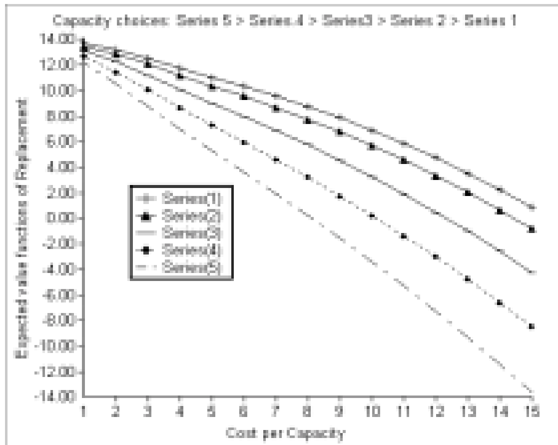

Figure 8.10: Expected value functions with various cost per capacity

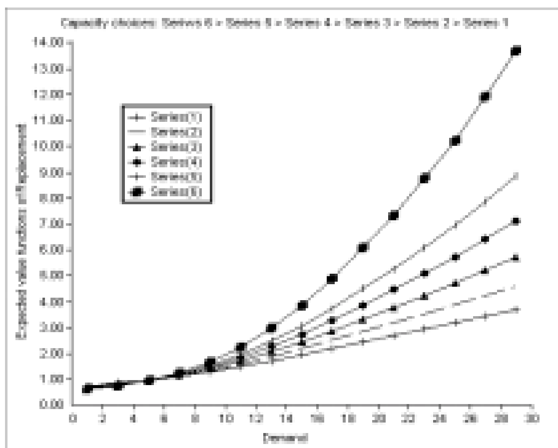

Figure 8.11: Expected value fucntions of replacement depending capacity choices

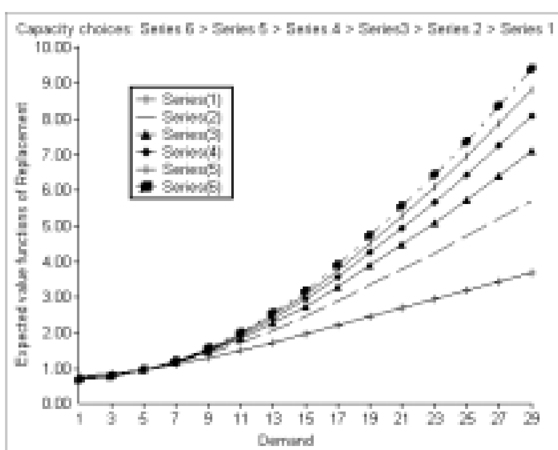

Figure 8.12: Expected value fucntions of replacement depending on capacity choices

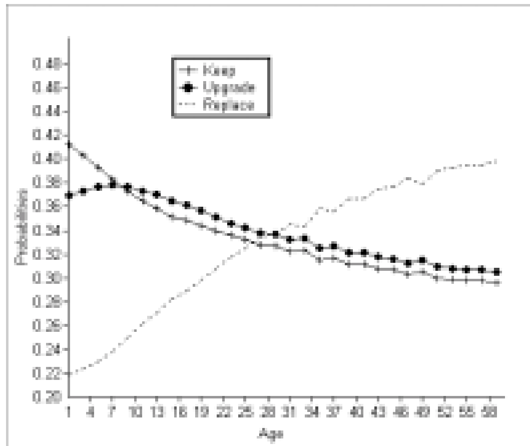

Figure 8.13: Simulated policy rules of keep, upgrade, and replacement

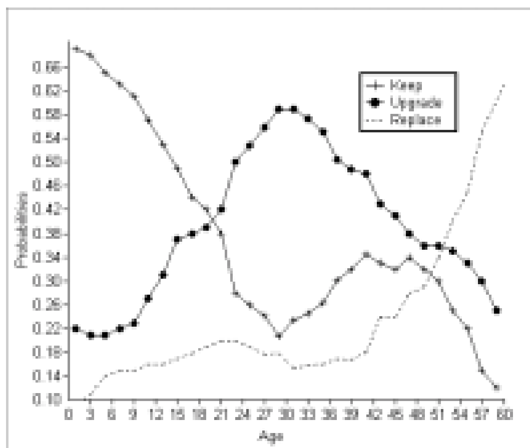

Figure 8.14: Simulated policy rules of keep, upgrade, and replacement

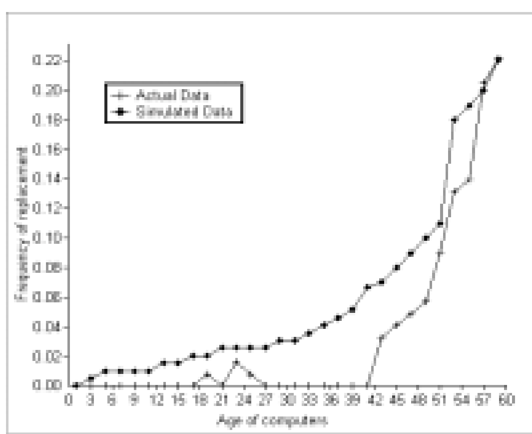

Figure 8.15: Comparison between the simulated data and the actual data

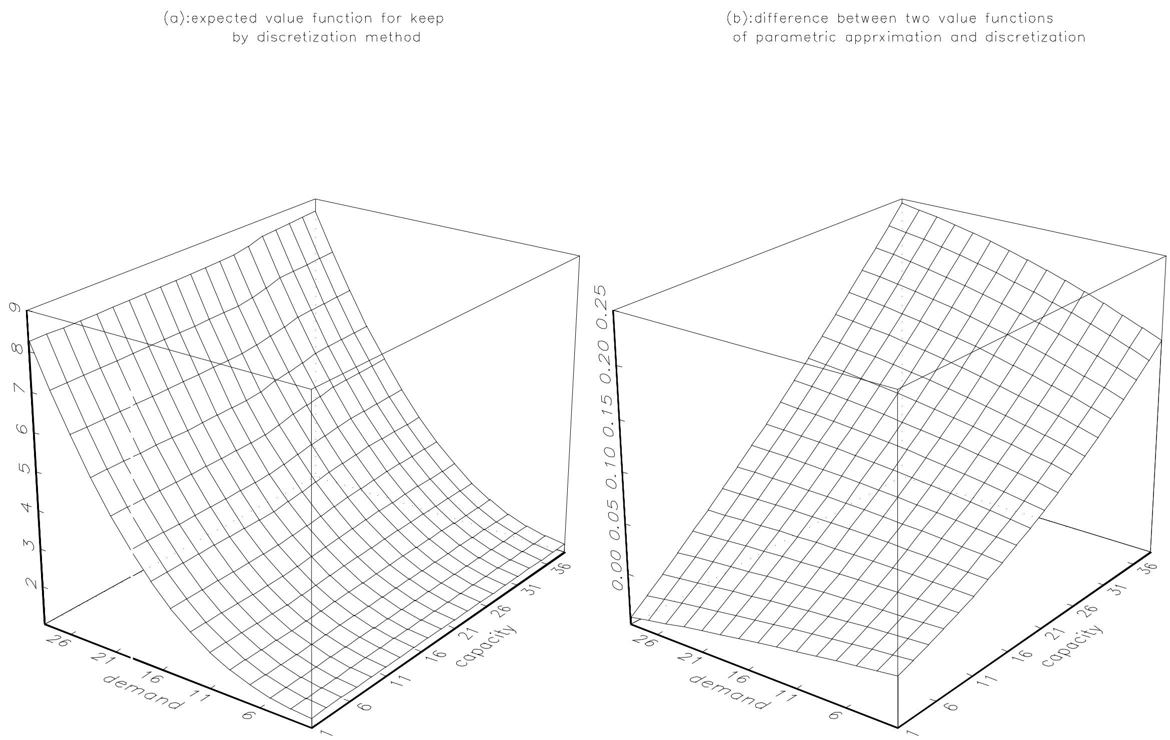

Figure 3: Differences between parametric approximation and discretization

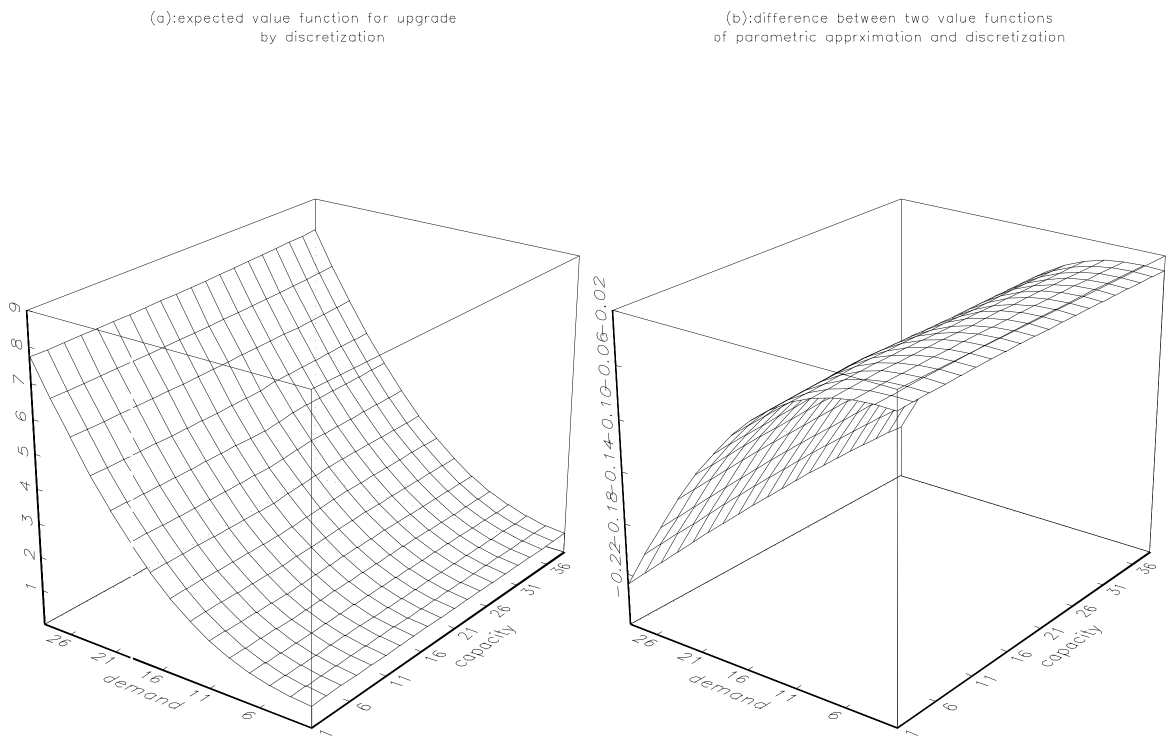

Figure 4: Differences between parametric approximation and discretization

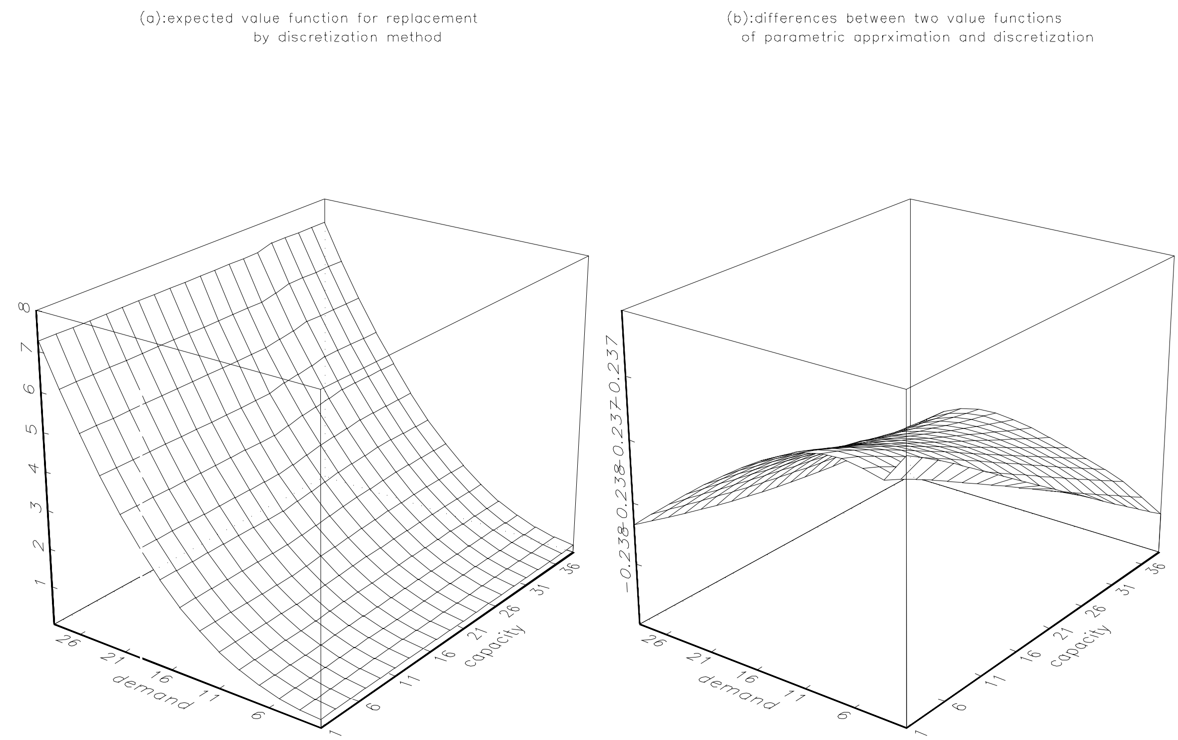

Figure 5: Differences between parametric approximation and discretization