ABSTRACT

of the Ph.D. thesis entitled

A STUDY OF INNOVATION ADOPTION & WARRANTY ANALYSIS IN MARKETING AND SUCCESSIVE SOFTWARE RELEASES

to be submitted to
University of Delhi

by
Adarsh Anand

Under the supervision of
Prof. P. K. Kapur
&
Dr. Ompal Singh

Department of Operational Research
Faculty of Mathematical Sciences
University of Delhi
Delhi-110007
The decade 1990’s was the beginning when the industries around the world started to make their footmarks in the global economy and faced global competition. Now day’s human world runs on systems designed by engineers and business people. As a result the quality of decisions made by these two professionals is of critical importance to the health of the world we live in. Making decisions on issues with important consequences is a highly complex problem due to the many competing forces under which the world is operating today, therefore decision makers make efforts to use mathematical models for decision-making and spend considerable money to acquire software systems to solve these models. In the light of this problem the scholars and researchers of the scientific disciple Operational Research targets to make bolder decision with less risk and better outcomes by making mathematical models of the systems and solving them with their knowledge of optimization algorithms.

Mathematical modeling is the art of translating problems from an application area into tractable mathematical formulations where theoretical and numerical analysis provides insight, answers and guidance useful for the originating application. The optimization models developed for the engineering and business professional allow them to choose the best course of action and experiment with the various possible alternative decisions.

Sophisticated computing systems are the backbone and marketing is the core activity for the efficient working of almost every business and industry. Mathematical modeling and optimization for the competent planning, marketing responsibilities of an organization and control of the software development process have become areas of prime interest in the field of Operational Research. This thesis endeavors to apply techniques of operational research in model building and optimization in the field of marketing and quality software development process.

Mathematical Modeling in Marketing
The goal of marketing is to create customer satisfaction profitably by building value-laden relationships with its most important entity – The customers. A marketing department in an organization typically has to perform the functions of product pricing, promotion, distribution, personal planning, etc. working closely with the other departments of the organization. Making marketing decisions in the marketing mix interactions under environmental uncertainties and the challenges of the global economy has become one of the most difficult tasks for the management. Traditionally, managements use their experience to arrive at any decision or they follow their operational doctrines, but decision-making based on judgment method in this formidably
complex situation cannot provide the best alternative decision. Sometimes it may lead to the wrong decisions due to the changing marketing environment, which may be a cause of an organization failure in the market. Therefore managements are becoming crucial enough to make their decisions based on scientific knowledge. Firms are increasingly applying modeling approach to marketing decision-making. Mathematical models in marketing are used to understand, predict and optimize processes.

In today’s time, it has become essential to develop an understanding of the process of diffusion of a product or service and substitution process between the various generations simultaneously. There are four key element of the diffusion process namely -The Innovation, Channels of communication, Time and the social System. The process of diffusion initiates the process of adoption. Challenges of global economy require firms to plan and control their innovation diffusion process efficiently so as to sustain their survival in the marketplace and prevent failure of its new products. Controlling the diverse process of adoption in the presence of environmental uncertainties, competition, dynamic adopters and markets etc. is most difficult task of the marketing. The inability to make accurate decisions and measurements relating to the various controlling variables such as adopter behavior, market size and expansion, sale response, promotional effectiveness, immediate response to competitors strategies and environmental changes etc makes a successful innovation a challenge. Mathematical modeling and optimization are the widely applicable and successful tools in that they provide information and measurements related to the various decisions of innovation diffusion process.

Models of Innovation Diffusion are used successfully by the various practitioners to evaluate the market response over the product life cycles and make valuable decisions related to product modifications, price differentiation and resource optimization etc. A large number of innovation diffusion models have been developed in literature. An innovation diffusion model produces a life cycle sales curve based on certain parameters such as adopter behavior, promotion, market variables etc of the diffusion process. The most famous and widely used diffusion model is by Bass, developed long back in 1969. Since then, it is used to depict the successive increase in the number of adopters and predict the continued development of a diffusion process already in progress. The present work also utilizes the framework developed by Bass and develops some extended frameworks in context of innovation diffusion models.
It is to note that the adoption process may get affected by many factors such as promotional effort, intensity of advertisement campaign, expectations and satisfaction level of the purchasers, after sales customer care etc. It is also affected by change in market conditions such as entry/exit of competitors from the market, launch of better quality/technology product or just change in the management of the company. There are many more factors which can cause change points in sales that may include changes in packaging (outlook), change in product features, increasing more outlets, availability of pack sizes, combo offers and discounts. Thus, the adoption rate may not be smooth and can be changed at some time moment $\tau$, known as change-point. The concept has been utilized in proposing some new models in adoption theory. Furthermore, if we allow for some randomness in some of the coefficients of a differential equation, we often obtain a more realistic mathematical model of the situation. Therefore, Stochastic Differential Equation based IDM’s are developed in this thesis in order to incorporate the irregular fluctuation in the adoption rate due to various uncertainty factors during adoption phase. The developed models are validated on sales data.

Marketing aims to achieve profit maximization through creation of value and a brand name for the company, which is impossible to achieve if one wants to survive with its existing products and services. In the face of marketing challenges innovation has become the path of growth and new product as the lifeblood for the firms fighting for surviving longer and make profit margin in the global market place. Modern manufacturing is characterized by (i) rapidly changing technologies, (ii) global markets, (iii) fierce competition, (iv) often nearly identical products due to common components and technology being used and, (v) better educated and more demanding customers. This has posed serious challenges for buyers, manufacturers and policy makers at national and regional levels. In the purchase decision of a product, buyers typically compare characteristics of comparable models of competing brands. When competing brands are nearly identical, it is very difficult, in many instances, to choose a particular product solely on the basis of the product related characteristics such as product price, special features, perceived product quality and reliability, financing offered by the manufacturer and so on. In such situations, post-sale factors – warranty, parts availability and cost, service, maintenance, and so forth – take on added importance in product choice. This concept forms the crux of the thesis. Studying the dynamics of the technology diffusions under the key determinants that influence the adoption of a technology across time and/or space into the market is crucial to assess the business case for new technologies. The topic diffusion has been widely studied by researchers from different disciplines, including Sociology, Economics, Psychology and Marketing. However a substantial
amount of research has been focused on one dimension: either to examine the individual’s adoption of an innovation or to explain the time path of adoption of technologies typically following an S-shaped curve. The other dimensions of the diffusion of an innovation, has gained less attention. Therefore it’s imperative to study a two-dimensional technology innovation diffusion model which combines the adoption time of technological diffusion and price of the technological product. In doing so, we have made use of the Cobb-Douglas production function. Using the aforesaid concepts, we have formulated some profit maximization models here that examine the significance of warranty as a key marketing tool to promote sales of goods.

In this day and age firms are speeding up their new product development process in order to replace their earlier products with the newer one, thereby introducing generations of new products. All the newer generation products are developed with an objective that they suit the latest needs of the consumers, are economic, cost effective, developed on the basis of most recent technology and are able to give competitive advantage to the manufacturer. Development and introduction of any new product demands firms to devote a significantly large percentage of their total profit for the research, development and marketing. The biggest risk associated with an innovation is the failure. A study has revealed that about 33% of the new products and services fail at launch. Reasons of failure may range from ineffective and inefficient promotion to poorly designed product, overestimated market potential, lagging behind technology, unusual costs etc. Another important issue related to the innovation is the reduction of the product life cycle time. In order to fight this issue the firms must find new products to replace old ones. Therefore, the launch timing of the succeeding generation is a very crucial decision for the management. Consequently the managements try to control the associated risks in the way of organization, professional staffing, marketing research, mathematical model construction and their application for the system optimization. In the present work, we have formulated an optimization model to find out the appropriate timing of launching of the newer generation when the management wants to control/minimize the associated cost with it.

**Mathematical Modeling for Building Quality in Software**

Computer software has moved to the center of our lives. Systems simple or complex are rapidly computerized or updated with latest computing systems due to ease of use and faster performance. Large-scale improvement in hardware performance, profound changes in computing architectures, vast increase in memory and storage capacity, a wide variety of exotic input and output options, has increased demand of software in automation of complex systems,
its use as a problem-solving tool for many complex problems of exponential size and to control critical applications. With this size and design complexity of the software has also increased many folds. Escalation in size, complexity, demand and dependencies on the computer systems has increased the risk of crises due to software failures. The impact of failures ranges from inconvenience to economic damage to loss of life. Even though the ability to design, test and maintain software has grown fairly, lot of further improvements are desired in the field. As such the software development process has become a challenging task for the developers.

In the early seventies, the Software Reliability Engineering (SRE) discipline emerged to establish and use sound engineering principles in order to economically obtain software systems that are not only reliable but also work efficiently on real machines, thus bringing the software development under the engineering umbrella. It focuses on quantitatively characterizing the distinguished attributes such as functionality, usability, reliability, maintainability, availability, testability, portability etc. of the software. Software Reliability is the key attributes of software quality since it quantifies software failures – the most unwanted events and hence is of major concern to the software developers and user. One of the major roles of SRE lies in assuring and measuring the reliability of the software.

With SRE one can deliver just enough reliability and avoid both excessive costs and development time. It emphasizes on avoiding fault introduction during the software development and removing most of the faults manifested in the software by means of software testing. Software testing involves checking processes such as inspections and reviews, at each stage of the software development life cycle (SDLC) start form the requirement specification to coding. Despite using the best engineering methods and tools during each stage of the software development the software is subject to testing in order to verify and validate it and to build the necessary reliability. Bugs if appear during software operation in user environment can be fatal to the users in terms of loss of time, money and even lives depending on criticality of the function as well as to the development in terms of cost of debugging, risk cost of failure and goodwill loss.

Testing is inherent to every phase of the SDLC, however testing in the testing phase aims solely on the reliability improvement and system verification and validation. Software testing in the testing phase is a three stage process in which first the systems individual components, programs and modules are tested called unit testing, followed by integration testing at subsystem and system level which includes top-down and bottom-up testing, stress testing and conclude with the
acceptance testing. Testing is basically of four types: defect testing, performance testing, security testing and statistical testing. Testing of any software may include some or all-testing types depending on the testing requirements. During testing, test cases are executed on the software and if the execution results in a software failure i.e. departure from the specifications is observed, attempts are made to isolate and remove the cause of failure called debugging process. Ideally the test cases that are executed on the software to test it are designed throughout its development life cycle. The purpose of testing is to improve system reliability, but we need to build a methodology to estimate the reliability as the testing progresses and predict it during the field usage. Software reliability growth models are the tools of SRE used successfully to develop test cases, schedule status, resource optimization, to count the number of faults remaining in the software and evaluate software quality quantitatively i.e. estimate and predict the reliability of the software during testing and operational environment. A software reliability model specifies the general form of dependence of the failure process/ reliability metrics and measurements on some of the principle factors that affect it, such as software and development process characteristics, fault introduction, fault removal, testing efficiency and resources, and the operational environment.

Reliability estimation is usually retrospective and determines achieved reliability from a point in the past to present using the failure data obtained during system test or operation. The prediction activity usually involves future reliability forecast based on available software metrics and measures. Depending on the software development stage this activity involves either early reliability prediction using characteristics of the software and software development process (case when failure data is unavailable) or parameterization of the reliability model used for estimation and utilizes this information for reliability prediction (case when failure data is available). In either activity reliability models are applied on the collected information and using statistical inference techniques reliability assessment is carried out.

Software reliability modeling has been a topic of practical and academic interest since the 1970s. Today the number of existing models exceeds hundred with more models developed every year. Most of the research work in software reliability modeling is done on failure count models, Bayesian models and Markov models. The majority of these models are failure count models, which are mainly based upon the Non-Homogeneous Poisson Process (NHPP). NHPP based SRGM describe the failure occurrence and/or failure removal phenomenon with respect to time (CPU time, calendar time or execution time or test cases as unit of time) and/or resources spent.
on testing and debugging during testing and operational phases of the software development.

NHPP based models can also be categorized as concave and S-shaped depending upon the shape of the failure curve described by them. Concave models describe an exponential failure curve while second category of models describes an S-shaped failure curve.

A plethora of software reliability growth models exists that take into consideration only single release of a software product. But this framework of modeling is to be extended when multiple releases has to be considered.

As the first release is the first step for any firm to enter into the market, a company has to pay more attention to it because generally first impression counts for the last impression. Therefore, the testing team has to rigorously test the software with an attempt to remove as many faults as possible. But, because of time and cost constraints it is not practically possible for the testing team to remove all the errors and thus the initial release of the software is made, with some of the fault content remaining in it. Next release is made when some new features or functionalities are being added for the first time in the software, this means adding some new code into the software which results in addition of some new faults in the software system. Also, the testing team cannot neglect the errors reported in operational phase of release 1. Therefore, while testing the newly formed code, the testing team has to consider the error removal of previous releases too. So, in release 2, the total fault removal combines the finite faults of the release which is under testing phase i.e. Release 2 and the remaining faults of release which is under operational phase i.e. Release 1. Moreover, in the study we have emphasized while testing a particular release, the testing team has to take into consideration the just previous release only and not all the previous releases. Also, not all the faults lying in the system have same character, some are easily detected and removed while some takes time to leave the system and are defined as simple and hard faults respectively. We have used different rate functions to remove the different types of faults, based on NHPP and in SDE environment as well. Based on this idea, mathematical modeling framework for successive releases software product has been developed.

The quality of the software system usually depends on how much time testing takes and what testing methodologies are used. On the one hand, the more time people spend on testing, the more errors can be removed, which leads to more reliable software; however, the testing cost of the software will also increase during the process. On the other hand, if testing time is too short, the cost of the software could be reduced, but the customers may take higher risk of buying
unreliable software. This will also increase the cost during the operational phase, since it is much more expensive to fix an error during the operational phase than during the testing phase. Therefore, it is important to determine when to stop testing, and release the software. Hence we have formulated a bi-criterion problem of simultaneously maximizing reliability and minimizing cost. Furthermore, in order to determine optimal release time of a new version of the software, Multi Attribute Utility Theory (MAUT) has been used. This technique attempts to identify relevant objectives for any given decision making problem, where a decision is typified by multiple objectives. It can be difficult to quantitatively compare the objectives like cost, reliability, failure intensity etc. one against another. In order to provide insight into this problem, a utility function is assessed for each of the relevant objectives. This allows for an appropriate multiple-objective utility function that is used to identify trade-offs and compare the various objectives in a consistent manner.

We have used the softwares; MAPLE, a very efficient tool in mathematics and a calculator, SPSS (Statistical Package for Social Sciences) for estimating and predicting reliability of the proposed models. SPSS is a comprehensive, flexible, statistical analytical and data management system and SAS (Statistical Analysis of Software). We have also made use of the Genetic Algorithm Technique in order to solve the optimization problems. The applicability of the proposed models is shown by validating it on actual software failure data sets obtained from different real software development projects. The comparisons with established models in terms of goodness of fit, Mean Square Errors (MSE), Bias, Variance, Root Mean Square Prediction Error (RMSPE) and Coefficient of Determination ($R^2$) have been presented.