ABSTRACT

of the Ph.D. Thesis entitled

MATHEMATICAL PROGRAMMING APPROACHES
FOR COMPONENT SELECTION OF
MODULAR SOFTWARE SYSTEMS

BY

SHILPI VERMA

Under the Supervision of

Dr. PANKAJ GUPTA

Submitted to

DEPARTMENT OF OPERATIONAL RESEARCH
FACULTY OF MATHEMATICAL SCIENCES
UNIVERSITY OF DELHI
DELHI-110007, INDIA

MAY, 2013
Mathematical optimization (alternatively, optimization or mathematical programming) is an important area of research in many fields of science and engineering. The classical optimization framework is the minimization (or maximization) of the objective(s), given the constraints for the problem to be solved. An optimization problem may be classified either as a single-objective optimization problem or as a multiobjective optimization problem. In contrast to a single-objective optimization problem, a trade-off exists among various objectives in a multiobjective optimization problem.

The classical optimization framework requires well-defined parameters (coefficients and right-hand sides). However, in the case of real-world problems, many parameters can be estimated only vaguely. That is, more often than not, real-world situations are characterized by imprecision (fuzziness) rather than exactness. In such situations, some flexibility may be allowed in specification of the objective function(s) and/or the constraints of the problem. Moreover, decision constraints may be relaxed in many situations to achieve decision goals more closely. These types of problems require an extension of the classical optimization framework. The development of fuzzy set theory has offered a practical way to model vague data. In classical optimization models, vague data are replaced by ‘average data’, while in fuzzy optimization models, vague data may be modeled by fuzzy numbers and fuzzy intervals. Fuzzy optimization models offer an opportunity to represent subjective preferences of the decision maker as precisely as a decision maker will be able to explain and describe. Therefore, an important advantage of fuzzy optimization models is the fact that they allow an adequate mapping of real-world problems, thereby reducing the risk of applying a wrong model of the reality and selecting solutions that do not reflect the real problem.

The prospect of reducing time, development cost, unmanageable software quality and risk to move to a new technology associated with software development has led organizations to search for a new, efficient, and cost-effective software development paradigm. The most promising solution in this direction is component-based software development approach. In this approach, software systems are built by acquiring and integrating reusable, pre-existing commercial-off-the-shelf (COTS) components rather than developing from scratch. Therefore, the main development effort is required in selection of the components such that the resulting integrated system must satisfy certain requirements (functional and nonfunctional)
of an application. The selection of suitable components is often a non-trivial task and requires a careful consideration of multiple criteria. The components that are not available in the market or cannot be purchased economically may be developed within the organization and are known as ‘in-house’ developed components. Furthermore, if some requirement(s) cannot be satisfied with COTS components, then the component(s) corresponding to the given system requirement(s) may also be developed in-house. Such decision making situations lead to a decision that is popularly known as ‘build-or-buy’ decision.

The purpose of the present thesis is to study the component selection problem systematically and extensively for the development of component-based software systems by using COTS components and/or in-house developed components. The software system under study consists of several programs, where a specific function of each program can call upon a series of modules. To select best-fit components for modular software systems, several single-objective and multiobjective optimization models are developed in crisp and fuzzy environment.

This thesis comprises five chapters that are summarized as under.

**Chapter 1** is introductory in nature and is devoted to mathematical preliminaries, which are used in subsequent chapters of the thesis. Starting with several fundamental notions, various methods of optimization including single-objective optimization, multiobjective optimization, and fuzzy optimization are briefly discussed. The component-based software development approach for the development of modular software systems by using COTS components and in-house developed components is then reviewed. The chapter concludes with a brief discussion of the research work carried out in the subsequent chapters of the thesis for component selection of modular software systems by using mathematical programming in crisp as well as fuzzy environment.

**Chapter 2** introduces a fuzzy cost-reliability trade-off optimization model of COTS selection which minimizes the total cost of the software system satisfying the constraints of minimum threshold on system reliability, maximum threshold on the delivery time of the software, and incompatibility among COTS components. In practice, system reliability, system delivery time, and the cost of COTS components are often imprecise. In order to deal with uncertainty in real-world
applications of COTS selection, in this chapter, the coefficients of the cost objective function, delivery time constraints, and minimum threshold on reliability are considered as triangular fuzzy numbers. The proposed fuzzy optimization model is converted into a pair of mathematical programming problems parameterized by possibility (feasibility) level $\alpha$ using Zadeh’s extension principle. The lower and upper bounds of the fuzzy objective value at different possibility levels $\alpha$ generate the left-shape function and right-shape function, respectively, of the membership function of the fuzzy cost. The solution approach adopted here completely preserves the fuzziness of model parameters and provides more information for cost-reliability trade-off analysis in COTS selection. A real-world case study is also discussed to demonstrate the usefulness of the solution approach and the efficiency of the model in a fuzzy environment.

In Chapter 3 to deal with the uncertainty in the COTS selection process, we propose fuzzy multiobjective optimization models. These models belong to the category of flexible programming where fuzzy goals and constraints represent the flexibility of the target values of objective functions and the elasticity of the constraints. We use a hybrid approach that combines quality models (ISO/IEC 9126), analytical hierarchy process, and fuzzy mathematical programming. This chapter is divided into two sections.

In the first section, we formulate a biobjective fuzzy optimization model for COTS selection based on vague aspiration levels of the decision maker in respect of the weighted quality and cost of COTS components. To represent vague aspiration levels of the decision maker, we use a nonlinear S-shape membership function. The model is constrained by limitations including the incompatibility among COTS components and the selection of only one COTS component for each module. Further, to reflect the relative importance of the decision maker for the two objectives, a weighted additive model is also presented by assigning different weights to the two objectives. The main advantage of using nonlinear S-shape membership functions is that if the decision maker is not satisfied with any of the COTS selection obtained, more COTS selections can be generated by varying values of shape parameters. With the help of numerical illustrations, we demonstrate the efficiency of the proposed models.

In the second section, we extend the biobjective fuzzy optimization models of the
first section to a fuzzy multiobjective optimization model. The fuzzy optimization model minimizes the cost, development efforts (number of lines of code), and execution time; and maximizes the reliability and weighted quality of the software system. We use linear membership functions to describe the fuzzy goals of the multiobjective optimization model. Furthermore, to determine the preferred compromise solution for the fuzzy optimization model, we use an interactive approach. Here, depending on the preferences of the decision maker for the various objectives, we can modify the compromise solution and this process is continued until the decision maker terminates the process. Numerical illustrations are provided to demonstrate the usefulness of the proposed model and the solution approach.

Chapter 4 deals with a decision making situation in which the software developer selects COTS components when multiple applications are handled concurrently for the development of modular software systems. We consider functional contributions of COTS components toward functional requirements of a component-based software system in the COTS selection process. Furthermore, to develop a good software system, modules with high cohesion and low coupling are required. Thus, in this chapter, we use intra-modular coupling density to measure the relationship between cohesion and coupling of modules. This chapter is divided into two sections.

In the first section, we propose a biobjective optimization model that maximizes the functional requirements of the modular software system and minimizes the total development cost of the software system. The model is constrained by many realistic constraints including a minimum threshold on the intra-modular coupling density for each application, reusability of COTS components, selection of only one COTS component from a set of alternative components for each functional requirement per application, and the selection of more than one component per module if required. In this section, we use the weighted sum method to solve the biobjective optimization problem. The model sensitivity has been shown with respect to changes in the minimum threshold value of the intra-modular coupling density for each application and also by varying the weight parameters of the two objective functions reflecting the preferences of the software development team (here the decision maker). A real-world scenario of developing two financial applications for two small-scale industries is included to demonstrate the efficiency
Mathematical Programming Approaches for Component Selection of Modular Software Systems

of the model.

The biobjective fuzzy optimization model proposed in the second section of this chapter is an extension of the optimization models presented in the previous section. To provide flexibility to the decision maker in the exact estimation of functional ratings and the cost of COTS components, we use fuzzy mathematical programming based on vague aspiration levels of the decision maker characterized by nonlinear S-shape membership functions. Further, to reflect the relative importance of the decision maker for the two objectives, a weighted additive model is solved by assigning different weights to the two objectives. The main advantage of the proposed models is that if the decision maker is not satisfied with any of the COTS selection obtained, more COTS selections can be generated by varying values of shape parameters of nonlinear S-shape membership functions.

Chapter 5 deals with ‘build-or-buy’ decision for the component selection problem in the development of modular software systems under multiple applications development task. In this chapter, we develop a cost minimization model that supports the decision whether to buy COTS components or to build in-house components while maintaining satisfactory values of quality attributes. The model also assists software developers to assemble the system within a specified time while maintaining a minimum level of reliability for each application, maintaining compatibility among components, and maintaining their reusability. In this model, quality constraints are related to the delivery time and reliability of applications. In addition to the optimal component selection, the model solution also provides the amount of unit testing to be performed on each in-house developed component. A small-scale case study in which a software developer undertakes two financial applications for two small-size companies is used to demonstrate the efficiency of the model in real-world applications. A model sensitivity with respect to changes in the component characteristics and constraint thresholds is performed to reflect changes in the component selection corresponding to changes in the input parameters.

Conclusion of the work done and scope for further research has also been briefly discussed. An elaborate list of the references is presented at the end of the thesis.