



Comparative Analysis of Tabu Search and Genetic Algorithms for Software Effort Estimation

Jagjit Kaur¹, Ritu Sindhu²

¹Research scholar, ²Associate Professor,
SGT institute of Engineering & Technology, Gurgaon

Abstract --- As with the growth of technology, competition level increases at a frequent rate. Each software company wants to develop its software in estimated time and budget. So, correct software effort estimation model is required. In this paper COCOMO II is used for estimation of effort in person month for the software project. Furthermore two metaheuristic algorithms like genetic and tabu search are used to optimize the coefficients of COCOMO II to calculate the estimated effort close to actual effort. Experiment is conducted by using both algorithms and it is observed that tabu search provides good effort estimate as compared to genetic algorithm.

Keywords—COCOMO II, GA(Genetic algorithm), Size, Tabu search.

I. INTRODUCTION

Today Softwares are becoming increasingly very important. The most important thing in software for its users is the cost which is very difficult to estimate. Estimation of cost is one of the biggest challenges in the field of software and also it is very expensive. Accurate software estimation of cost helps the company for decision making process and effectively manage the software development process.

This software cost would affect the entire software management process which includes project planning, scheduling and resource allocation. Effort is usually estimated in person month and it can be translated into actual cost. So calculating the cost and effort is the major task which in turn calculates the overall employees required for the project.

Software cost estimation is done for the number of purposes like budgeting, risk analysis, project planning and control, software improvement investment analysis.

There are a lot of cost estimation techniques in the software industry. Here are few techniques:

- 1: algorithmic (Parametric) model
- 2: Expert Judgement
3. top- down
4. bottom-up
5. Estimation by analogy

Software organizations are using various cost estimation models or methods to estimate the exact and accurate cost that will be used in the development of software project. In the proposed paper algorithmic model i.e. COCOMO II is used for effort estimation.

Among many models COCOMO II (Non – Proprietary) is the best used model in calculating the effort for the software project but there are various techniques which are used to optimize the coefficients of COCOMO II model to get the estimated effort equivalent to actual effort and better than current COCOMO II current coefficient effort. These techniques are like such as Genetic algorithm, neural network, simulated annealing and many more were used earlier [5] but in proposed model the author compares the tabu and genetic algorithm. Both Tabu Search and Genetic Algorithm has been applied to a number of problems like instruction scheduling [4], job shop scheduling [10] employee scheduling, time table management. [4] Genetic Algorithms locate optima by using natural search and genetics whereas Tabu search consists of tabus which help in searching an optimum solution.

II. COCOMO II Model

CONstructive COSt MOdel (COCOMO 81) is the parametric software cost estimation model which was developed by Barry W. Boehm. It uses an algorithmic method to evaluate the cost of the software. It is based on waterfall lifecycle model [3]. It is a transparent model which provides all the details like assumptions, definitions and cost estimates of the model. Besides all these it suffers from some limitations like:

- For each phase of project life cycle separate estimation model should be used.
- Not a realistic model as assumptions and requirements vary with time.
- Not suitable for non sequential, reuse case models and object oriented models.

COCOMO II, a successor of COCOMO 81, which was developed in 1995 overcome all these limitations faced by its predecessor [9]. It is more accurate model. It takes qualitative inputs and produce quantitative results. COCOMO II has three sub models application composition model, early design model and post architecture model [9]. Of all these COCOMO II uses post architecture model which is a detailed model and it has been prepared after the architecture has

been designed. COCOMO II post architecture takes cost drivers, scale factors and size as three most important inputs. The values of the cost drivers and scale factors depend upon the model being used. Cost drivers are the characteristics of the software that influence the effort in carrying out a certain project. COCOMO II has as an input a set of seventeen Effort Multipliers (EM) or cost drivers which are used to adjust the nominal effort (PM) to reflect the software product being developed and five scale factors which have rating levels are Very Low (VL), Low (L), Nominal (N), High (H), Very High (VH) and Extra High (XH)

TABLE I
COCOMO II SCALE FACTORS [8]

| Scale factor | Description |
|--------------|-------------------------|
| PREC | Precedentedness |
| FLEX | Development flexibility |
| RESL | Risk resolution |
| TEAM | Team cohesion |
| PMAT | Process maturity |

TABLE II
COCOMO II COST DRIVERS [8]

| Cost Driver | Description |
|-------------|------------------------------|
| SCED | Schedule |
| SITE | Multi site development |
| RUSE | Reusability |
| DOCU | Documentation needs |
| CPLX | Product complexity |
| TIME | Execution time |
| STOR | Storage |
| PVOL | Platform volatility |
| ACAP | Analyst capability |
| APEX | Application experience |
| PLEX | Platform experience |
| LTEX | Language and tool experience |
| PCON | Personnel continuity |
| TOOL | Software tools used |
| PCAP | Programmer capability |
| DATA | database size |
| RELY | Reliability |

COCOMO II calculates the effort in person month by using the following equation:

$$\text{Effort (PM)} = A \times \text{SIZE}^E \times \prod_i \text{EM}_i \quad [1][8]$$

Where A is multiplicative constant having a value of 2.94

As we know size is the most important factor in calculating the effort of the software project and it is measured in Kilo Source Line Of Code (KSLOC), so E can be calculated as

$$E = B + 0.01 \times \sum_i \text{SF}_i \quad [1][8]$$

Where B is a constant= 0.91.

In this way effort is calculated in person month.

But the software companies are also more interested in calculating the duration the project lasts. So it can be calculated from effort as

$$T (\text{development time}) = C \times (\text{PM})^F \quad [1][8]$$

Where C is constant = 3.67

F can be calculated as

$$F = D + 0.2 \times 0.01 \times \sum \text{SF}_i \quad [1][8]$$

Where D is constant = 0.28.

III. TABU SEARCH ALGORITHM

Tabu search is a Meta heuristic search technique created by Fred W. Glover in 1986 for solving the complex reliability problems. It continues iterate in a loop until it gets an accurate result. It makes the use of short term, long term and working memory [7]. It is basically a local search method to escape from local minima. Tabu uses the concept of tabu list which is also known as short term memory which avoids the cycling because all the recent history of previously visited solutions is stored in the tabu list and it forbids the same moves again.

Tabu search algorithm works by initially selecting a random number from a solution space and then put that number in the tabu list as well as to the current best solution. Whenever a new neighbor is chosen from a set of individuals then it is compared with the tabu list. If that is already in the tabu list then it is avoided, if not then it is compared with the individual stored in the current best solution. If its value is better than that then the value gets replaced and tabu list gets populated with the new individual. Whenever tabu list gets completely occupied then it frees its memory by using the concept of first comes first out. In this way tabu search first performs diversification to choose the neighborhood area of the solution space and intensification of search with the help of flexible short term memory.

IV. GENETIC ALGORITHM

Genetic algorithm was originated by John Holland and DeJong. It is a meta heuristic algorithm that was used to solve various problems like TSP, job shop scheduling, Instruction scheduling, flow shop scheduling. Genetic algorithms are classical algorithms that generate feasible solution from infeasible solution. It is used to solve problems about which very little is known. It generate several solutions for a particular problem. Firstly it create a set of individuals which is known as population. On the basis of some fitness function individuals are selected from the population . The selected individuals then produce one or more offspring which are then mutated. In this way it continue to remove individuals from a solution set until the solution is feasible.

V. EXPERIMENT AND RESULTS

The main objective of the Experiment performed is to reduce the uncertainty of current COCOMO II post architecture coefficients i.e. a, b, c and d by using both tabu search and genetic algorithm and make the comparative analysis of both. The experiment is performed on Turkish and Industry data set on 15 different projects to optimize effort. This dataset consist of size of each project in kilo source line of code, actual effort and the COCOMO II current coefficient effort in Person Month (PM) in Table 3.

TABLE III
DATA SETS WITH THEIR SIZE AND EFFORT VALUES [7][6]

| P.No | Size | Actual effort | COCOMO II predicted effort |
|------|--------|---------------|----------------------------|
| 1 | 003.0 | 001.20 | 003.60 |
| 2 | 019.90 | 074.60 | 092.70 |
| 3 | 004.05 | 002.0 | 002.30 |
| 4 | 004.25 | 004.50 | 009.30 |
| 5 | 015.00 | 004.0 | 063.20 |
| 6 | 010.00 | 003.00 | 036.20 |
| 7 | 131.00 | 619.90 | 745.20 |
| 8 | 031.85 | 005.00 | 147.10 |
| 9 | 001.61 | 002.10 | 002.00 |
| 10 | 001.37 | 001.00 | 000.90 |
| 11 | 023.11 | 004.00 | 063.20 |
| 12 | 064.10 | 332.00 | 256.70 |
| 13 | 114.28 | 018.00 | 294.00 |
| 14 | 002.00 | 002.00 | 002.90 |
| 15 | 040.53 | 022.00 | 028.60 |

The working is implemented on NetBeans IDE 8.0. Current COCOMO II PA coefficients are as:

a= 2.94, b= 0.91, c= 3.67, d= 0.28.

The coefficients optimized by Tabu search are as:

a=2.51, b=0.62, c=3.78, d=0.01.

The coefficients optimized by Genetic algorithm taken from [6]

a= 2.814, b=0.804, c=3.518, d= 0.343

By using these coefficients the effort is calculated which is predicted in table IV as shown [6]

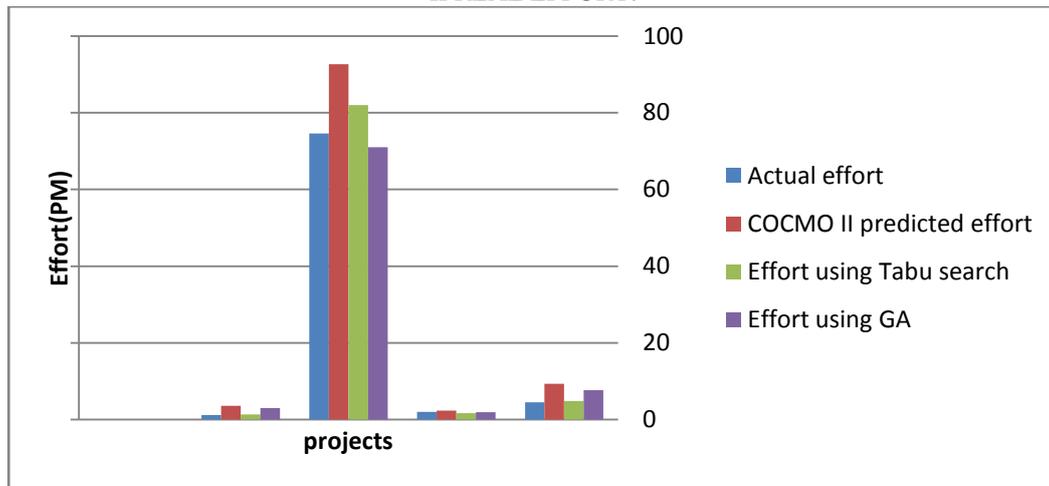
TABLE IV
PREDICTED VALUES OF EFFORT USING TABU SEARCH AND GA

| P.No | Size (KSLOC) | Actual effort (PM) | COCOMO II predicted effort | Estimated Effort using Tabu search | Estimated Effort using GA |
|------|--------------|--------------------|----------------------------|------------------------------------|---------------------------|
| 1 | 003.0 | 001.20 | 003.60 | 1.33 | 2.97 |
| 2 | 019.90 | 074.60 | 092.70 | 82.0 | 71.0 |
| 3 | 004.05 | 002.0 | 002.30 | 1.7 | 1.89 |
| 4 | 004.25 | 004.50 | 009.30 | 4.8 | 7.63 |

It can be seen from the table that tabu search provide more accurate results as compared to Genetic Algorithm in optimizing the COCOMO II effort.

FIG 1

GRAPH SHOWING COMPARISON AMONG EFFORTS USING TABU SEARCH, GA, ACTUAL, AND COCOMO II REAL EFFORT.



VI. CONCLUSION

The algorithms have effectively solved the complex optimization problem and achieve the more accurate results by optimizing the coefficients of COCOMO II.

It has been seen that both the algorithms has effectively optimized the current COCOMO II estimated effort but it is concluded from the figure 1 that tabu search provides the calculated effort much closer to actual effort as compared to Genetic algorithm.

REFERENCES

- [1] Bogdan Stępień, “Software Development Cost Estimation Methods and Research Trends” Computer Science, Vol. 5, 2003, pp. 68-82
- [2] Kavita Choudhary, “ GA Based Optimization of Software Development Effort Estimation “ International Journal Of Computer and Technology, Vol. 1, No. 1, Sep.2010, ISSN: 0976-8491, pp. 38-40
- [3] Barry Boehm, Chris Abts, Sunita Chulani, “Software development cost estimation approaches –A survey” Annals of Software Engineering 10, 2000, pp.182-190
- [4] Steven J. Beaty, “Genetic Algorithm versus Tabu Search for Instruction Scheduling”pp.14-18.
- [5] Caserta M, Uribe AM. Tabu search-based metaheuristic algorithm for software system reliability problems. Computers and Operation Research (2007), doi: 10.1016/j.cor.2007.10.028.
- [6] Astha Dhiman, Chander Diwaker, “Optimization of COCOMO II using Genetic Algorithm”, AIJRSTEM, pp. 208-212.
- [7] Ekananta Manalif, “ Fuzzy Expert – COCOMO Risk Assessment and effort contingency Model in Software Project Management” The university of Western Ontario London, Ontario, Canada.
- [8] A. Yahya, Rodina Ahmad, Sai Peck Lee, “Effects of Software Process Maturity on COCOMO II’s Effort Estimation from CMMI Perspective” In 2008 IEEE, Department of Software Engineering, University of Malaya, pp.255-256.
- [9] nancy Merlo-schett, “seminar on software cost estimation” Requirement Engineering research group, Department of Computer Science, WS
- [10] B.T. Eck. Good solutions to job shop scheduling problems via tabu search. Technical report, Department of Industrial Engineering and Operations Research, Columbia University, New York, May 1989.