Novel Meta-Heuristic Algorithmic Approach for Software Cost Estimation

Ruchi Puri
Department of Computer Science and Engineering
Chandigarh University

Iqbaldeep kaur
Associate Professor
Department of Computer Science and Engineering
Chandigarh University

Abstract-The Software Cost Estimation is very important task for completing the project successfully. The estimation in software development depends on various factors especially on cost and effort factors for which further AI (Artificial Intelligence) and Algorithmic models have been put into usage. This paper aims to discuss the methods for calculating the effort by using meta-heuristic algorithms for software development. This work lists two major aspects. First a thorough survey of already implemented meta-heuristic algorithms like MOPSO, Bee Colony Optimization and Firefly, have been presented. In the subsequent section, two novel techniques viz., BAT algorithm and Human Opinion Dynamics algorithms have been proposed for software cost estimation using effort parameter.

Keywords- COCOMO, MOPSO, Bat, Bee Colony Optimization, Firefly, Human Opinion Dynamics, etc.

I. INTRODUCTION

A successful software project development not only relies on the product efficiency but also the accurate estimation of its cost. For better resource utilization and project progress evaluation project manager need to know the accurate cost estimation. Nowadays, many companies and organization are giving great importance to software development and production and the main focus is on customer satisfaction and simultaneously the production cost should be kept in consideration so that it does not lead to any financial loss or customer dissatisfaction.

Nowadays, many cost estimation models have been developed based on size measure, such as lines of code, function point. Researchers have proposed several methods to solve this problem, the most primitive of these solutions are mathematical and algorithmic methods .The process of improvement in estimation methods has grown steadily; estimation began with very simple assumptions and today it includes complicated equations and techniques. A discussion of existing estimation methods requires dividing the methods into main groups: algorithmic and non algorithmic.

A.COCOMO

Constructive Cost Model (COCOMO) the formal effort estimation model developed by Boehm in 1981 is used as an algorithmic model to calculate effort. Three basic types are: Basic COCOMO, Intermediate COCOMO and Detailed COCOMO. Intermediate model is considered by various researchers. The values of parameters of COCOMO are fixed for organic, semi-detached and embedded. But these parameters vary from organization to organization .We need to tune the parameters so as to get the optimal results. For fine tuning of parameters the methods of meta-heuristic algorithms are used to get optimized solutions.

B. META-HEURISTIC ALGORITHMS

Nowadays, to find out the optimized solution of complex problem evolutionary algorithms are used over algorithmic models. Meta-heuristic Algorithms are based on natural phenomenon and are suitable for complex optimization problems. These algorithms solve the optimization problem according to population and keep on searching and evaluating for a number of times until an optimised result is obtained.

II. RELATED WORK

A Research on software cost estimation is going on as software companies are growing most rapidly so there is a need to find more accurate estimation technique. Many Researchers worked in this field and in this section of paper illustrates some previous work.

Benala et.al [6] used PSO to train the functional link artificial neural network PSO-FLANN for effort prediction proves better when compared with FLANN. Chebyshev polynomial is used by author for mapping the input features from one form to another form. Maxwell, Nasa 93, Cocomo81 datasets used for the evaluation process. Rao, G et.al [1] proposed MOPSO algorithm for multi-objective optimisation problem. Effort estimation with MOPSO gives better results compared to COCOMO as it worked on multi-objective problem so MARE (mean absolute relative error) is minimised and prediction accuracy is maximised. The author has performed two experiments were performed and the results are good for small projects size less than 50 KDLOC and for experiment two in large projects the accuracy is good in some cases only . Bardsiri et.al [12] used Analogy Based Estimation Approach for the estimation process combined with PSO. The proposed idea consists of testing and training stages in which an estimation model is developed and evaluated. The results show that combination of PSO and ABE gives better results and improve the performance of existing models. Dizaji et.al [13] proposed a bee colony optimization algorithm for effort estimation and results are compared to intermediate COCOMO and the results implies that the proposed approach reduce the mean absolute relative error to 0.1619.

Sheta et.al [6] applied genetic algorithm for the estimation of COCOMO model parameters. The developed GA based model is evaluated with fitness function VAF (Variance Accounted For) and the modified COCOMO model considered. The developed model tested on NASA software projects and provides good estimation capability but it can find a more advanced function so that estimate effort will be more accurate. The Author of this paper Aljahdali et.al [7] proposed an alternative technique of Differential Evolution (DE) as COCOMO-DE model for the estimation .The developed model was capable of providing good estimate compared to other such as Halstead, Doty, Walston-Felix. The Author Magne et.al [3] done a systematic review and identifies 304 cost estimation papers in 76 journals and classify them according to estimation approach, research topic, research approach, study context and data set

III. META-HEURISTIC ALGORITHMS TO FIND EFFORT FOR SOFTWARE COST ESTIMATION

These Meta-heuristics Algorithms have already been implemented in the field of software cost estimation. Significant work has been done for calculating effort using these algorithms recently.

A. Multi-Objective Particle Swarm Optimization (PSO)

Particle Swarm Optimization (MOPSO) was introduced by Eberhart and Kennedy in 1995. Swarm Intelligence is an innovative distributed intelligent paradigm for solving optimization problem which took its inspiration from the biological examples by swarming, flocking and herding phenomenon in vertebrates [1].There are two best values Pbest and Gbest. Each particle is stored according to its fitness value. With each iteration the particle position is updated to first to Pbest as local best then to Gbest as global best. Single objective optimization problem defined as maximizing or minimizing we use PSO but in some problems there is a need for optimisation of two more objectives. A multi objective Optimisation is defined as X=[X1,X2....Xn] where X is the control variable vector, and n is no. of control variables. Objective function is min/max.

F= {f1(X) ,f2(X).......fm(X)}

Each objective combined with weight is given by the formula:

W1 * f1(X) + W2 *f2(X)+.....+Wm*fm(X)..equation(3.1)

And normalize the weights using W1+W2+.......Wm=1.....................................equation(3.2)

$V^{t+1} = V_i^{t+} c_1 * rand()_1 * (Pbest - S_i^t) + c2 * rand()_2 * (Gbest - S_i^t)$ .....equation(3.3)

$S_i^{r+1} = S_i^r + V_i^{r+1}$ ....equation(3.4)

Whereas, S_i^r is current search point, S_i^{r+1} is modified search point.

- , V_i^r is current velocity, V_i^{r+1} is modified velocity.
- cj is weighting factors.
- Rand() is uniformly distributed random number.

Table1.Pseudo Code of MOPSO Algorithm

```
Initialize n
{
(particle random position and velocity vectors)
      Do
         For (n=0,n<=1,n++)
             {
               Evaluate  f_n(X)    where x is any instance from equation 3.1 and 3.2, where n=1 or 2
              }
              If
                   {
                      Pbest=p_i.        (i=integer number for particle)
                      Set Gbest = best of Pbest.
                    }
  Else
           Update V_i and P_i
           (Velocities and position of particles using equations 3.3 and 3.4...(explained above)
            For (n=0,n<=1,n++)
                  {
                    Evaluate  f_nb(X)   f_nb updated function, x is any instance from equation 3.1 and 3.2
                   }
             If
                  {
                     Gbest = o_solution  where  o_solution  is optimal solution
                      Result found
                     }
              Else
                      Return(0)
}
```

B. Firefly Algorithm

It is a meta-heuristic algorithm developed by Dr. Xin-Shi Yang based on flashing characteristics of fireflies. It is a multimodal optimization algorithm, belongs to behaviour of fireflies or lightning bugs. FA (Firefly Algorithm) has three basic rules:

- All Fireflies are attracted to each other with disrespect to gender.
- Attractiveness is associated with light emission or brightness such that bright flies attract to less bright ones and in their absence the movement becomes random.
- Last rule is that brightness is proportional to objective function

Table 2.Pseudo Code of firefly algorithm

Objective Function f(x)
Generate Initial population of fireflies x,(i=1,2,....,n)
Light Intensity I_i at x_i is determined by $f(x_i)$
while
(t< MaxGeneration)
do
for (i=1 : n all n fireflies **do**
for (j-1) : i all n fireflies **do**
if ($I_j > I_i$) **then**
Move firefly i towards j in d-dimension;
End if
Attractiveness varies with distance
Evaluate new solutions and update light intensity
end for
end for
Rank the fireflies and find the current best
end while
Post process results and visualisation

C. BEE-COLONY OPTIMIZATION

This Optimization Algorithm was first introduced by Teodorivic. This Algorithm is based upon the natural phenomenon of getting food by bees which are performed in two stages as moving backward and moving forward. In the first stage bees find many basic solutions and at the second stage they present there solutions in a meeting and then they decide according to food quality .The probability of selecting a solution is calculated by:

V_(j)= (Max(F)-Fj)/(Max(F)-Min(F)), J =1, 2....., N.....................................equation(1)

Whereas N denotes the number of solutions, F denotes all solutions and Fj denotes the current solution.

Table 3. Pseudo code of bee colony optimisation

Initialize V(j)
J=1 to n
Do
{
For (B_j=1, $B_{j<=}$n, B_j++)
{
F_{pass} (Forward Pass)
$C_{Solution}$= C_1 + C_2 +........+ C_N (where C_N constructive moves)
For ($C_{N=}$1, C_N<= $C_{last\text{-}constructive\ move}$, C_{N++})

{
Find solution
}
}
For (B_j=1, $B_{j<=}$n, B_j++
{
Case B:
B_{pass} (Backward Pass)
For each F_i
(where f is fitness function)
If
{
fitness value<random number
Then select other solution
Solution=best value
Stopping Condition is met= value of a, b parameters
}
else
Jump B
}

IV. NOVEL META-HEURISTIC APPROACH INTRODUCED FOR COST ESTIMATION

The previous sections detailed the meta-heuristics for effort calculation that have been empirically used. This section throws light on a novel genre of meta-heuristic techniques bringing forward the impact of social approaches into technological computation. There has been introduced two novel models namely, BAT and HOD (Human Opinion Dynamics). There has been work done on the application of impedance -tongue using HOD [17] and BAT is used in the application areas of complex global numerical optimisation problems. However, there has not been any concrete work done using these two for effort estimation.

A. BAT Algorithm

In this algorithm search is inspired by social behaviour of bats and phenomenon of echolocation. It is a novel meta-heuristic technique for global numerical optimization problems. In Bat algorithm, the position of each bat is defined by x_i^t and velocity v_i^t, frequency f_i, loudness A_i^t, and the emission pulse rate r_i^t in a D-dimensional search space. The solutions x_i^t and v_i^t at time step t are given by

$$f_i = f_{min} + (f_{max} - f_{min})\beta \text{ ....equation(1)}$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*) f_i \text{ ....equation(2)}$$

$$x_i^t = x_i^{t-1} + v_i^t \text{ ....equation(3)}$$

Table 4. Pseudo Code of BAT Algorithm

Step 1: Set the generation counter t=1; Initialize the population of number of bats P randomly and each bat according to the given problem; define loudness A_I, pulse frequency Q_i and initial velocities v_i (i=1,2…..p); set pulse rate r_(i.)
Step 2: While the termination criteria t < Max Generation
Do
Generate new solutions by adjusting frequency, velocities and locations....equation [(1)-(3)]

If (rand > 〖r〗_i)
Then
Select a solution among the best solutions;
Generate a local solution around selected best solution
End If
Generate a Solution again by flying bat randomly
If (rand < 〖A_(i) & f(x〗_i) <f(x))
Then
Accept the new solutions
Increase r_i and reduce A_i
Rank the bats and find the current best x
t=t+1;
Step 3: End While
Step 4: Process the results and visualize

B. HUMAN OPINION DYNAMICS

It is an inspiration to solve complex optimization problems based upon human creative problem solving process. As human beings are considered the most intelligent social animal in the world, the algorithm is based upon opinion formation of human beings. Opinion formation is an Evolutionary process. A real valued Optimizer CODO (Continuous Opinion Dynamics Optimizer) is developed and henceforth it is also called as CODO Algorithm. The Algorithm has four basic essential elements mainly:

- Social Structure
- Social Influence
- Opinion Space
- Updating Rule

Social Structure: It is an important aspect of social dynamics which governs the interaction between individuals, group of individuals the frequency of interaction and the way of interaction e.g. small world, cellular Automata model, random graphs etc have been proposed and simulated in social physics [2].

Social Influence: It is that influence in which individual act according to expectation of others. Social influence has been formed by taking into account these two factors:

- Social Ranking
- Distance between two individuals

Social influence $w_{ij}(t)$ of individual j on individual i is given by equation 1

$$w_{ij}^{t} = \frac{SR_j(t)}{d_{ij}(t)}$$ ....equation(1)

Whereas d_{ij} is Euclidean Distance between two individuals.

- SR (Social Rank) of individual is according to their fitness value. Highest SR is assigned to the individual with minimum fitness value.
- Opinion Space: The Individuals Opinion is of two types: discrete and continuous whereas continuous could be any real value and discrete may take values as [0,1] or [-1,1].
- Updating Rule: Updating Rule is used to update the positions of individuals in search space. It is the one of the most important factor of any iterative optimisation algorithm. The Update Rule according to the equation is given by:

$$\Delta O_i = \frac{\sum_{j=1}^{N} (O_j(t) - O_i(t)) w_{ij}(t)}{\sum_{j=1}^{N} w_{ij}(t)} + \varepsilon_i(t), j \neq 1$$ ....equation(2)

Whereas $o_j(t)$ is the opinion of neighbours of individual i , N is the no. of neighbour ,

$w_{ij}(t)$ represents the Social Influence and $\epsilon_i(t)$ is a distributed random noise with zero mean.

Table 5. Pseudo Code of HOD (Human Opinion Dynamics)

```
START
Initialize opinion for every dimension d by randomly assigned values.
iter=0;
While (iter < max & error >=min)
 DO
Society. fitness=Evaluate fitness function(society. opinions)
Society. ranking=calculate rank;//it ranks the individuals based on society fitness
 iter=iter+1;
FOR
each individual i and each dimension d
DO
 Calculate u_ij from Eq.......(1)
 Update opinion of individual  i as  defined above
END
```

V. CHALLENGES

Software Industry has faced many challenges of Software crisis due to time, cost and quality. The software to be developed needs to be accurate with expected quality, within the estimated cost given certain time constraint. However, if budget exceeds the estimated budget then project fails horribly. The project cases where the cost exceeds the estimated cost the organisation is left crippled with wastage of all the effort and time put into the project without any business. Since cost proves to be a vital parameter for any project being undertaken, utmost importance is required to estimate cost using precise effort. So, there is a need to have a technique that gives more accurate results in terms of effort.

V. CONCLUSION

There has been a lot of research on the parameter evaluation done in COCOMO for software cost estimation. A number of optimisation techniques like ACO, Firefly, GA, PSO-FLANN, have been used empirically to modify the conventional COCOMO model and its performance in terms of accuracy and prediction and error reduction. In the current work, there has been done a critical analysis of optimization techniques like MOPSO, BAT, Firefly, Bee Colony Optimisation and Human Opinion Dynamics. The paper also has introduced the novel idea of implementing HOD (Human Opinion Dynamics) and BAT for the same. Although these techniques find their applications in the areas of social sciences and global numerical optimisation .The Authors are currently working on the implementation of proposed techniques for software effort estimation.

REFERENCES

[1] G. Sivanageswara Rao Ch V. Phani Krishna, and K. Rajasekhara Rao, "Multi Objective Particle Swarm Optimization for Software Cost Estimation," In *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol I*, pp. 125-132, Year (2014).

[2] Rishemjit Kaur, Ritesh Kumar, Amol P. Bhondekar, and Pawan Kapur, "Human opinion dynamics: An inspiration to solve complex optimization problems,"*Scientific reports* 3 Year (2013).

[3] Magne Jorgensen Martin Shepperd, "A systematic review of software development cost estimation studies," *Software Engineering, IEEE Transactions on* 33, no.1 ,pp 33-53.,Year (2007).

[4] Sweta Kumari, and Shashank Pushkar, "Performance analysis of the software cost estimation methods: a review," *Int. J. Adv. Res. Comput. Sci. Softw. Eng*3 ,pp. 229-238,Year(2013).

[5] Sheta, Alaa F, "Estimation of the COCOMO model parameters using genetic algorithms for NASA software projects," *Journal of Computer Science* 2, no. 2 ,pp. 118-123.Year(2006).

[6] Tirimula Rao Benala, Korada Chinnababu, Rajib Mall, and Satchidananda Dehuri, "A Particle Swarm Optimized Functional Link Artificial Neural Network (PSO-FLANN) in Software Cost Estimation," In *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA)*, pp. 59-66,Year (2013).

[7] Aljahdali, Sultan, and Alaa F. Sheta, "Software effort estimation by tuning COOCMO model parameters using differential evolution," In *Computer Systems and Applications (AICCSA), 2010 IEEE/ACS International Conference on*, pp. 1-6.,Year (2010).

[8] Sumeet Kaur Sehra, Yadwinder Singh Brar, and Navdeep Kaur, "Soft Computing Techniques for Software project Effort Estimation," *International Journal of Advanced Computer and Mathematical Sciences* 2, no. 3 ,pp.162-163,Year(2011).

[9] Alaa F.Sheta, and Sultan Aljahdali, "Software effort estimation inspired by COCOMO and FP models: A fuzzy logic approach," *International Journal of Advanced Computer Science and Applications (IJACSA)* 4, no. 11 ,Year (2013).

[10] Chris F,Kemerer, "An empirical validation of software cost estimation models,"*Communications of the ACM* 30, no. 5 ,pp. 416-429.,Year(1987).

[11] Astha Dhiman, and Chander Diwakar, "Optimization of COCOMO II Effort Estimation using Genetic Algorithm,." *proceedings of AIJRSTEM* ,pp. 208-212,Yea (2013).

[12] Vahid Khatibi Bardsiri, Dayang Norhayati, Abang Jawawi, Siti Zaiton Mohd Hashim, and Elham Khatibi.,"A PSO-based model to increase the accuracy of software development effort estimation." *Software Quality Journal* 21, no. 3 pp.501-526,Year(2013).

[13] Mandeep Kaur, and Sumeet Kaur Sehra, "Particle swarm optimization based effort estimation using Function Point analysis," In *Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on*, pp. 140-145,Year (2014).

[14] Sultan Aljahdali, and Alaa F. Sheta, "Software effort estimation by tuning COOCMO model parameters using differential evolution,." In *Computer Systems and Applications (AICCSA), 2010 IEEE/ACS International Conference on*, pp. 1-6,Year (2010).

[15] Zahra Ashegi Dijazi, Reza Ahmadi, Hojjat Gholizadeh, and F. S. Gharehchopogh, "A Bee Colony Optimization Algorithm Approach for Software Cost Estimation," *International Journal of Computer Applications* 104, no. 12 ,pp. 41-44,Year (2014).

[16] Stein Grimstad, Magne Jørgensen, "A framework for the analysis of software cost estimation accuracy," In *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, pp. 58-65, Year (2006).

[17] Amol P Bhondekar., Rishemjit Kaur, Ritesh Kumar, Renu Vig, and Pawan Kapur, "A novel approach using Dynamic Social Impact Theory for optimization of impedance-Tongue (iTongue)," *Chemometrics and Intelligent Laboratory Systems* 109, no. 1 pp. 65-76Year,(2011)