FP-International Journal of Computer Science Research (IJCSR)
Volume 2, Issue 2, Pages 109-117, June 2015

Human Opinion Dynamics for Software Cost Estimation

Ruchi Puri
Research Scholar
Department of Computer Science
Chandigarh University
Gharuan, India
ruchi2192jk@gmail.com

Iqbaldeep kaur
Assosciate professor
Department of Computer Science
Chandigarh University
Gharuan, India
Iqbaldeepkaur.cu@gmail.com

ABSTRACT
Human Opinion Dynamics is a novel approach to solve complex optimization problems. In this paper we propose and implement Human Opinion Dynamics for tuning the parameters of COCOMO model for Software Cost Estimation. The input is coding size or lines of code and the output is effort in Person-Months. Mean Absolute Relative Error and Prediction are the two objectives considered for the fine tuning of parameters. The dataset considered is COCOMO. The current paper demonstrates that use of human opinion dynamics illustrated promising results. It has been observed that when compared with standard COCOMO it gives better results.

Keywords
Human Opinion Dynamics, COCOMO, MARE, Social influence, Update Rule

1. INTRODUCTION
When the information available is not complete or imperfect with less computation capacity and there is a need to find out the best solution from a large set of sample solutions then meta-heuristic or partial search algorithm is used to provide a solution to an optimization problem in Computer Science or mathematical optimization. There are various meta-heuristics available like Multi-objective Particle Swarm Optimization [1], Genetic algorithm,Ant colony optimization[2],Bacteria Foraging Optimization, Genetic Algorithm, BAT Algorithm, Memetic Algorithm, Firefly Algorithm etc. Extensive research has been done on these over large no. of applications. There is no exhaustive research using human opinion dynamics. Human opinion dynamics finds its applications in the areas of social physics. However recently a researcher throws light on the use of human opinion dynamics for solving complex mathematical optimization problems applied on some benchmark mathematical functions and compared the results with PSO [3]. As Human being is the highest creature so the algorithm inspired by human creative problem solving process can be useful and provide better results. It is used as dynamic social impact theory as an optimized for the optimization of an impedance-tongue and results are compared with Genetic Algorithm and PSO [7].

2. HUMAN OPINION DYNAMICS

It is a meta-heuristic technique to solve complex optimization problem based upon human creative problem solving process. Understanding the concept of collective decision making, the study of opinion dynamics and opinion formation is important. It has been one of the most significant areas in social physics. Human Interactions give RISE to different kind of opinions in a society [3]. In social network the formation of different kinds of opinions is an evolutionary process. There are several models describing human interaction networks like cultural dynamics, opinion dynamics, crowd behaviour, human dynamics etc utilised for search strategies and complex mathematical optimization problems. The process of collective intelligence from the tendencies of social influence with effects of individualisation escapade for developing search strategies[3]. The algorithm formed based upon the opinion formation structure of individuals. The algorithm is governed by four basic essential elements: Social Structure, Opinion Space, Social influence, Updating Rule.

2.1. Social Structure
The Interaction between individuals, group of individuals, frequency of interaction and the way they interacts comes under social structure. There are number of models like Cellular automata, Small world, Random graphs etc have been proposed and simulated in social physics [1].The models are explained below:

2.1.1 Cellular Automata Model
This model was first considered by Von Neumann studied in 1950 as a model used in biological systems. It works according to some set of rules based on the states of neighbouring cells. It is affected by the presence of corresponding or adjacent neighbour. It is a discrete

model used in complex science and biological processes. An example of cellular automata model is given below in table 1 which is having a system of cell objects exhibiting the following characteristics as follows:

- All the cells live on Grid
- Each cell has a state which might have two possibilities 0 or 1 or referred as yes or no.
- Neighbourhood of cell can be defined in plenty of ways but most probably the list of adjacent cells.
- The figure represents grid of "cells" each "yes" or "no" and the coloured part presents the neighbourhood of cells.

Table 1.Example of cellular automata model

Yes	No	Yes	No	No	Yes	No
No	Yes	Yes	Yes	No	No	Yes
No	Yes	No	No	Yes	Yes	No
Yes	Yes	No	Yes	No	No	Yes
No	No	Yes	No	Yes	Yes	No
Yes	Yes	No	Yes	No	No	Yes
No	No	Yes	No	Yes	Yes	No
No	Yes	Yes	Yes	No	No	Yes

2.1.1.1 Random Graph

Random graph can be well defined as a random process of probability distribution over graphs. The application area of random graph in the areas where complex networks need to be designed.

2.1.1.2 Small World

It is the type of network where most nodes are not influenced by the neighbours but finding ways to reach each other by small number of steps.

2.2. Social Influence

It is the influence of individuals on each other and they act according to others actions or suggestions. "Equation (1) describes social influence u_{ij} (t) of individual j on individual i is given by

$$u_{ij}(t) = \frac{SR_j(t)}{d_{ij}(t)} \qquad (1)$$

Where d_{ij}(t) is the Euclidean distance between individuals .Social Ranking (SR) is based upon the fitness value of individuals.

2.3. Updating Rule

This rule is used to update the position of individuals in the search space. As it is dynamic in nature so change of position according to the best fitness value needs to be updated. In context to optimization problems it determines the new updated position of individuals. Equation (2) demonstrates the formula for updating rule

$$\Delta O_i = \frac{\sum_{j=1}^{N}(O_j(t) - O_i(t))u_{ij}(t)}{\sum_{j=1}^{N} u_{ij}(t)} + \epsilon_i(t), j \neq 1 \; (2)$$

Where ΔO_i represents updating rule,o_j(t) is the opinion of number of individuals,u_{ij}(t) represents Social influence,$\epsilon_i(t)$ is a normally distributed random noise with mean zero and N is the number of neighbours.

2.4. Opinion Space

There are two types of opinions of the individuals continuous or discrete. Continuous is the one which takes real values. Discrete takes values in the given range [0,1] or [1,-1].

3. HUMAN OPINION DYNAMICS IN SOFTWARE COST ESTIMATION

Cost Estimation is an important activity and can be done throughout the entire life cycle of the software product to be developed. It is the process of calculation of effort used for the development of project. Time and budget are the two important factors in software project management. The main focus is on time and budget in software project development [4]. There are various models used for the effort calculation in software cost estimation. One of the most widely used algorithmic model is COCOMO .The parameters of COCOMO tuned with the help of meta-heuristic techniques. We are using Human Opinion Dynamics to optimize the parameters of COCOMO.

3.1. COCOMO

COCOMO model is developed by Boehm and have been widely used for the calculation of effort. Effort calculated by COCOMO model is measured in terms of size and constant value parameters a, b, c. We use

Intermediate COCOMO II model in which effort can be calculated using equation (3). "Equation (3) gives the formula for effort".

$$Effort = a * (size)^b * EAF + c \qquad (3)$$

where size is the size of project measured in LOC (lines of code) or KLOC.EAF are effort multipliers. The value of parameters a=3 b=1.2 .As these values are fixed for COCOMO model but these parameters vary from organisation to organisation depends on various factors like environmental factors. So, there is a need to tune the value of parameters to obtain better result in terms of accuracy and less error.

3.2. Fitness function

Fitness function is that function which is used to evaluate that which opinion is performing best and gives best results.

Each objective has some weight which is used to combine the two objectives into single objective. The weights assigned must be equal to one.

$$W1+W2=1 \qquad (4)$$

Hence, we are using the fitness function where we need to minimize the error and maximize the prediction. Most Real World problems involve optimization of two or more objectives [1].A multi objective optimization function involves minimization of one and maximization of other. The fitness function used in our equation is based upon two objectives MARE i.e. mean absolute relative error and prediction.

"Equation (4) defines the fitness function we used in our approach."

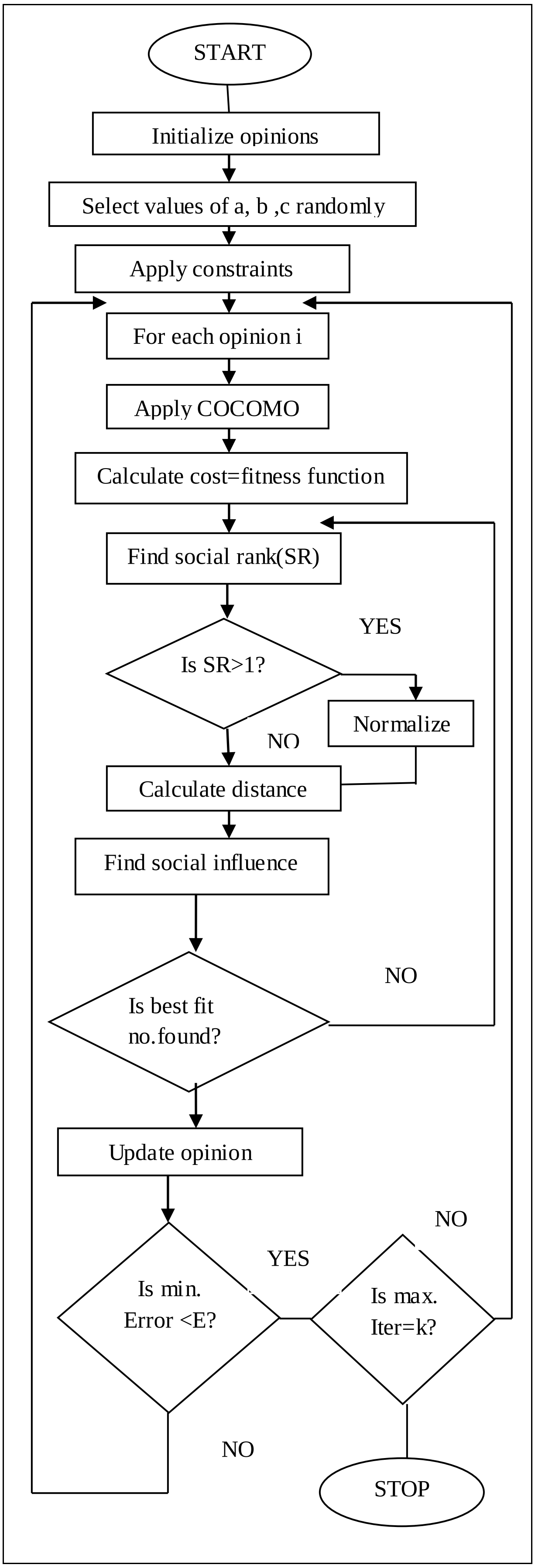
$$function = w1(MARE^b) + w2(1 - pred.) \quad (5)$$

Where MARE is mean absolute relative error and Prediction (n) is the project having n% error."Equation (6) gives the formula to calculate mean absolute error"

$$\%MARE = \sum\left[\frac{abs.(mes.effort - est.effort)}{(mes.effort}\right]/n(6)$$

Where "n" is the total no. of projects, abs. is the absolute effort. Mes. Effort is the actual measured effort and est. Effort is the effort calculated by using cost model COCOMO II and proposed technique.

4. PROPOSED METHODOLOGY



- Step 1: START
- Step 2: Initialize the opinions from 1 to 30
- Step 3:Assign any random value to a, b ,c
- Step 4:Apply constraints
- Step 5:For every value of opinion i.
- Step6:Apply COCOMO formula and calculate effort with the randomly assigned values of parameters a, b, c.
- Step 7:For these values of a, b, c calculate cost ie. equal to fitness function
- Step 8:Find Social Rank for the present value or opinion
- Step 9: If Social rank is greater than 1 than calculate distance ie. Euclidean distance of two individuals.
- Step 10:Find social influence of the opinion which is based upon social rank and Euclidean distance
- Step 11:If best fit number is found ie, the opinion which fits aptly in the fitness function then go to Step 12 otherwise go to step 7
- Step 12: Update the value of Opinion

And if minimum error <E and maximum iteration =k then stop, otherwise go to Step 5 and perform the optimization process again.

5. PERFORMANCE RESULTS AND IMPLEMENTATIONS

This section describes the experimentation part. For testing the effectiveness of proposed models, we tested it on COCOMO dataset. Two datasets of 20 projects and 21 projects are considered. Tuned value of parameters obtained by implementing the above methodology.

Experiment 1.Total 20 projects are considered from COCOMO dataset .Total no. of iterations performed =100, no. of opinions =30. The optimized values of a, b , c obtained are a=4.2 , b=1 , c= 1.3 .

1) The table given below is showing the values that are already available in dataset from COCOMO dataset.

Table2. Experimental results for the comparison of effort

Meas	COCOMO Effo	HOD Effort
240	347.2294197	224.744
33	55.15808369	43.052
8	12.18428797	10.292
79	115.5563114	84.38
9	11.53574316	12.9044
7.3	7.475113353	7.688
5.9	6.196705766	6.26756
47	70.20610386	47.252
8	10.44854672	8.8556
8	8.648095925	8.78
6	5.548713573	4.265
45	66.75800317	50.297
36	49.08832967	35.408
41	71.19532527	64.136
14	25.52408244	24.299
20	16.51001299	15.8864
70	100.7635967	75.434
50	73.60772685	51.62
38	48.82798088	42.653
15	18.54325035	15.08

2) Table 3 given below represents the actual effort and compared to effort calculated by COCOMO and proposed method.

Table3. Results for comparison of error

COCOMO Error	HOD error
0.446789249	0.063566667
0.671457082	0.304606061
0.523035997	0.2865
0.462738119	0.068101266
0.28174924	0.433822222
0.023988131	0.053150685
0.050289113	0.062298305
0.493746891	0.005361702
0.306068341	0.10695
0.081011991	0.0975
0.075214405	0.289166667
0.483511182	0.117711111
0.363564713	0.016444444
0.736471348	0.564292683
0.823148745	0.735642857
0.17449935	0.20568
0.439479953	0.077628571
0.472154537	0.0324
0.284946865	0.122447368
0.23621669	0.005333333

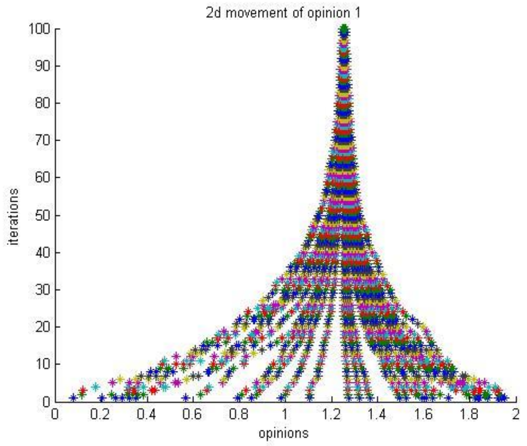


Figure.1.Represents 2d movement of convergence of opinions for opinion 1

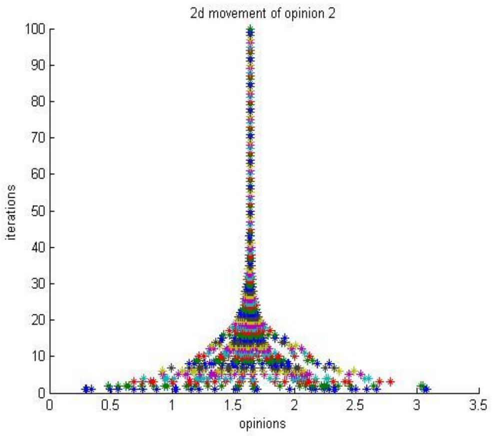


Figure 2.Represents the plot of convergence of opinions for opinion 2

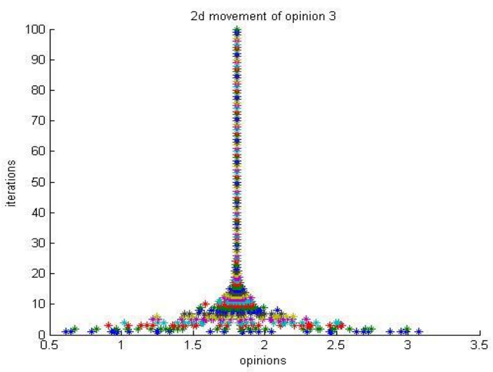


Figure3. Represents the 2d movement of convergence of opinion 3

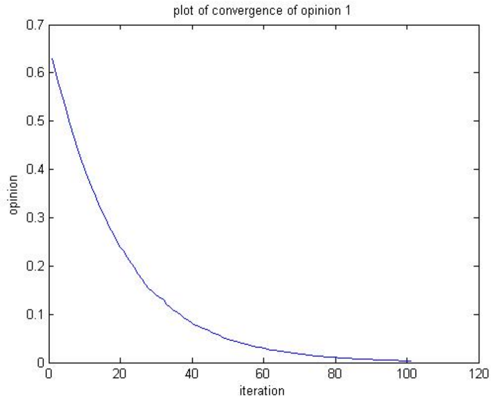


Figure 4. Represents the plot of convergence of opinion 1 when 100 iterations are performed.

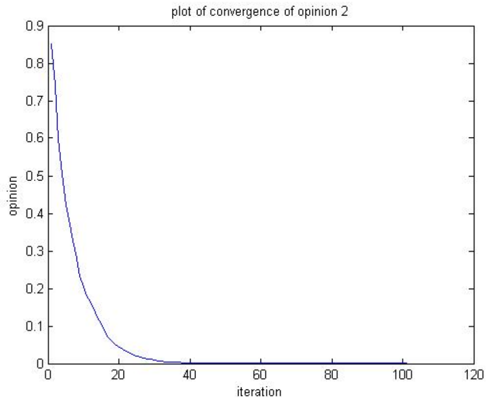


Figure 5. Represents the plot of convergence for opinion 2

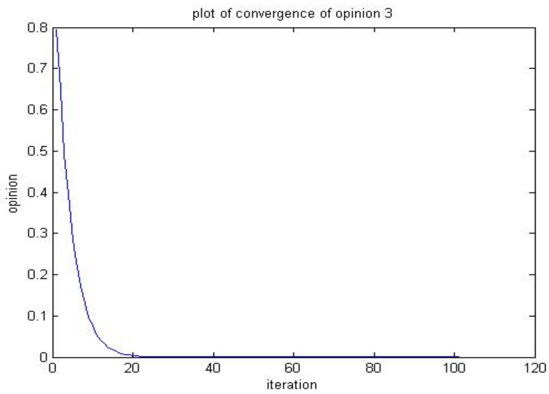


Figure 6.Represents the plot for convergence of opinion 3

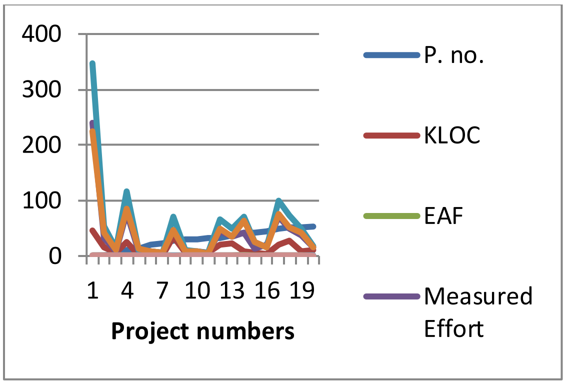


Figure 7. Represents the dataset values which consists of project number. lines of code, effort multipliers (EAF) and measured effort.

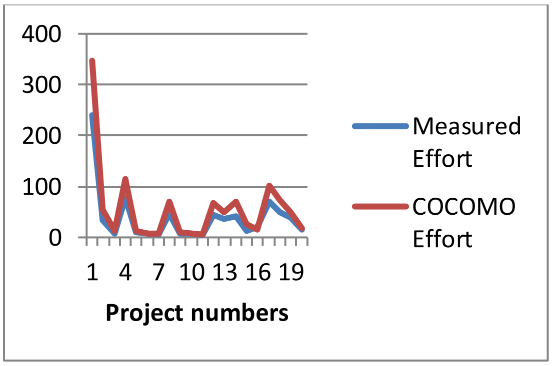


Figure 8. Represents the comparison of COCOMO effort with measured effort

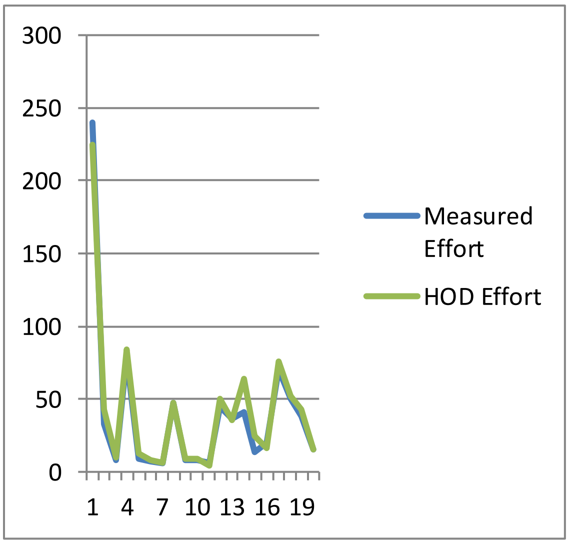


Figure 9. Represents the actual measured effort and effort calculated by human opinion dynamics

Effort calculated by Human Opinion Dynamics is more close to the actual measured effort. Thus our proposed approach gives better results

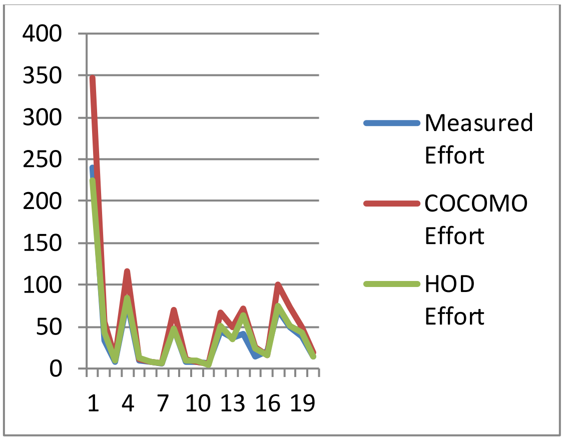


Figure 10.Respresents the comparison between actual effort, COCOMO effort and effort calculated by Human Opinion Dynamics

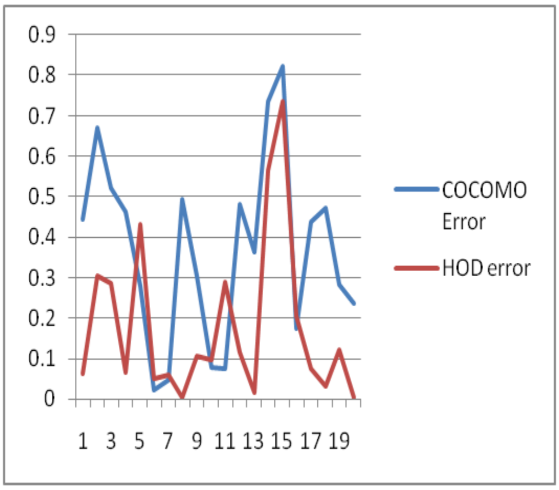


Figure 11. Represents the mean absolute relative error between COCOMO and human opinion dynamics, thus the error of COCOMO is more than the error of human opinion dynamics.

COCOMO error is more than the error calculated by human opinion dynamics.

FP-International Journal of Computer Science Research (IJCSR)
Volume 2, Issue 2, Pages 109-117, June 2015

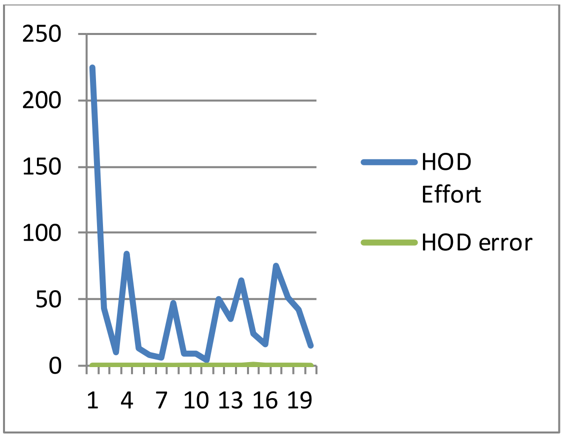


Figure 12.Represents calculated effort and error of human opinion dynamics

Experiment 2. Total 21 projects are considered from COCOMO dataset for testing the model. The tuned value of parameters obtained a=3.9, b=1.1,c=5.8.

Table 4. Represent the results of comparison of effort calculated by COCOMO, measured and human opinion dynamics.

Table 4. Experimental results of comparison of effort

Meas	COCOMO Effo	HOD Effort
240	347.2294197	302.010963
33	55.15808369	48.5426138
43	35.15169074	33.98172143
8	12.18428797	7.257456815
107	933.2187223	884.803766
423	424.6827895	387.1125386
321	229.1000679	217.2700783
218	259.9765682	244.6808678
201	255.9654486	226.1020171
79	115.5563114	103.5240969
73	65.69984976	70.72363706
61	55.75777155	57.46190997
40	40.5200193	40.4160499
9	11.53574316	8.264152613
539	474.8809821	395.0870279
453	464.8472928	379.52697
523	460.1465625	409.9776829
387	362.3003293	314.0069941
88	90.05438122	87.769738
98	183.0457279	178.3232425
7.3	7.475113353	3.205747324

Table 5 .Represents the error values of COCOMO error and human opinion dynamics error.

Table 5. Comparison of error

COCOMO Error	HOD error
0.446789249	0.063566667
0.671457082	0.304606061
0.523035997	0.2865
0.462738119	0.068101266
0.28174924	0.433822222
0.023988131	0.053150685
0.050289113	0.062298305
0.493746891	0.005361702
0.306068341	0.10695
0.081011991	0.0975
0.075214405	0.289166667
0.483511182	0.117711111
0.363564713	0.016444444
0.736471348	0.564292683
0.823148745	0.735642857
0.17449935	0.20568
0.439479953	0.077628571
0.472154537	0.0324
0.284946865	0.122447368
0.23621669	0.005333333

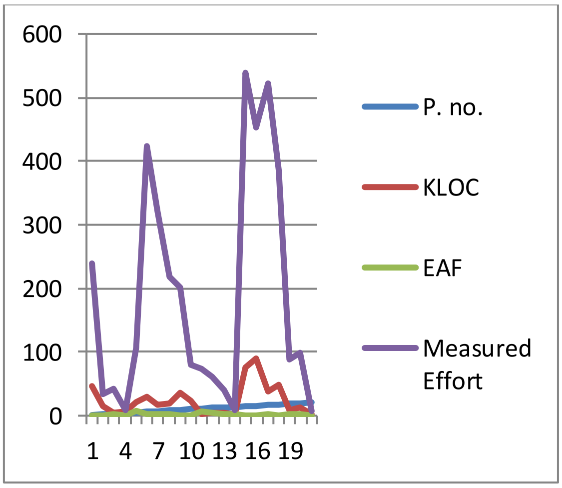


Figure 13. Shows the dataset 2 of 21 projects

The dataset consists of Project numbers .KLOC (kilo lines of code) ,EAF, Measured effort

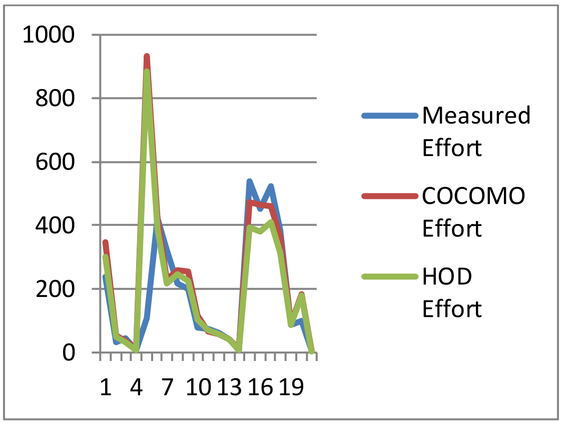


Figure 14 .Represents the effort calculated by COCOMO and when compared to actual effort then it can be seen that the estimated effort is more than the actual effort.

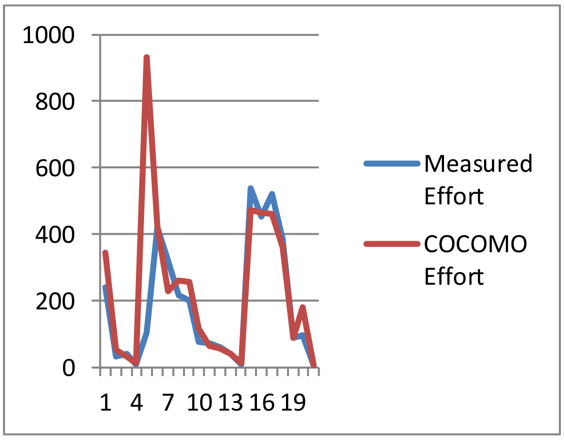


Figure 15. Gives the comparison between COCOMO effort and measured effort

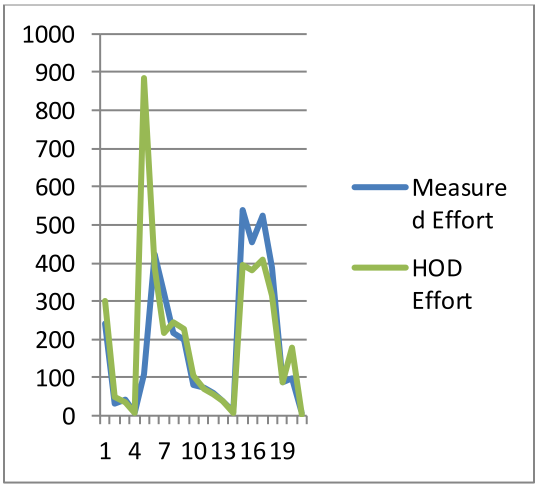


Figure 16. represents the comparison of actual effort with HOD effort

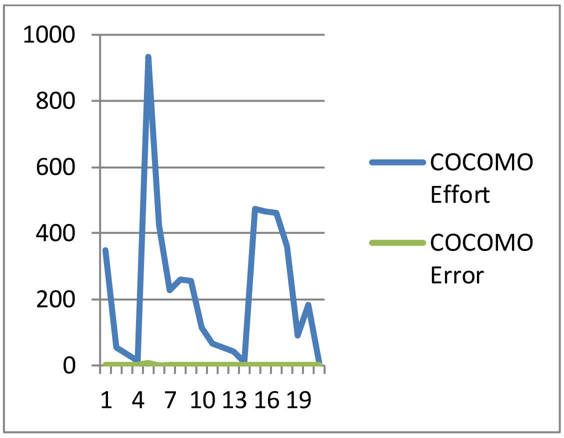


Figure 17.Represents COCOMO effort and COCOMO error

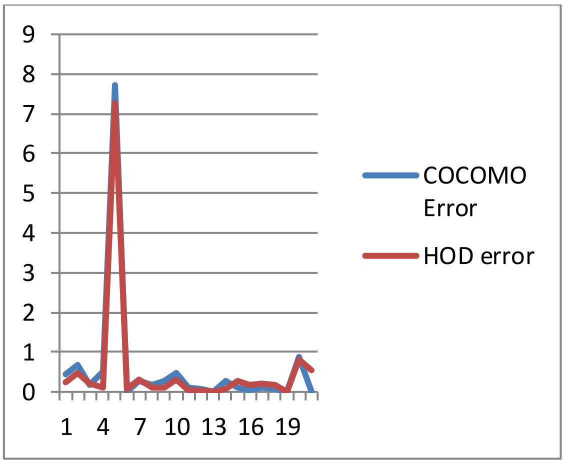


Figure18. Represents the comparison between COCOMO error and HOD error

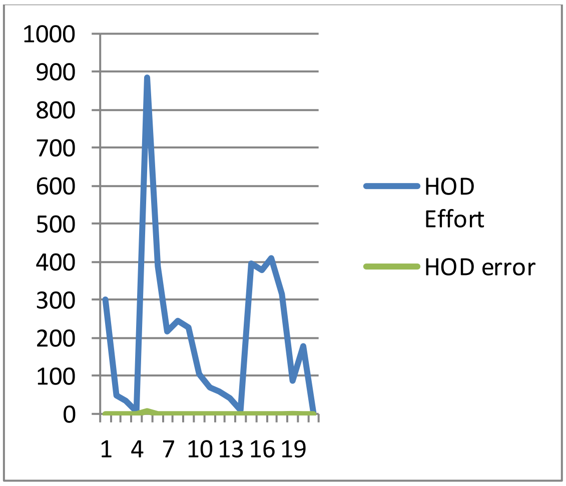


Figure 19. Represents HOD error and HOD effort

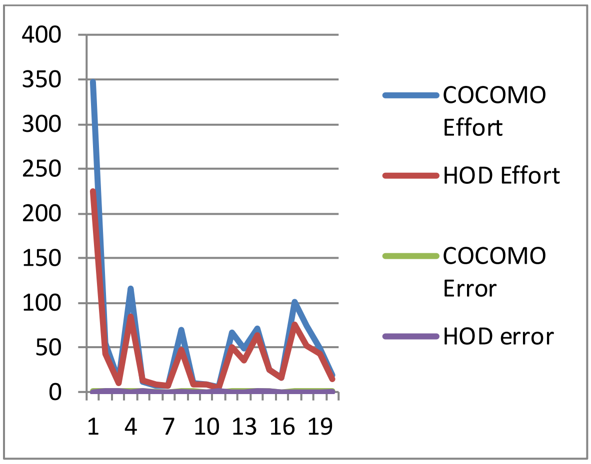


Figure 20.Represents the comparison between COCOMO effort, HOD effort and actual measured effort error.

6. CONCLUSION AND FUTURE SCOPE

The Software Cost Estimation problem was dealt with in this paper which is a very important problem in the SDLC cycle as it influences the decision making process. A novel meta-heuristic approach known as Human Opinion Dynamics based Optimisation to estimate the Software Cost using intermediate COCOMO-II model. The software costs are predicted and the results are found to be quite better than that of the normal COCOMO-II model. The results when compared among HOD and COCOMO infer that the HOD's performance is quite better than the COCOMO's results both in terms of convergence and accuracy.

In future, other meta-heuristics algorithms can be applied and the results can be compared with our proposed methodology. The advanced COCOMO models can also be utilised and the approach can be utilised to develop an online system which would automatically predict the software cost serving as an automated feedback system for the business analyst.

7. ACKNOWLEDGEMENT

I would like to express my sincere and deep sense of reverence for **Dr. Amit Verma** ,Head of Department, Computer Science and Engineering, Chandigarh University, Gharuan for his guidance and continuous encouragement

8. REFERENCES

[1] Rao, G. Sivanageswara, 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. ,Springer International Publishing, Year 2014.

[2] Maleki, Isa, Ali Ghaffari, and Mohammad Masdari. ,"A new approach for software cost estimation with hybrid genetic algorithm and ant colony optimization," *International Journal of Innovation and Applied Studies* 5, no. 1,pp. 72-81,Year (2014).

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

[4] Bardsiri, Vahid Khatibi, 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).

[5] Benala, Tirimula Rao, 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. Springer Berlin Heidelberg, Year (2013).

[6] Sheta, Alaa F., 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, pp no. 11,Year (2013).

[7] Bhondekar, Amol P., 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: 65-76,Year (2011).