Indian Journal of Science and Technology, *Vol 9(44), DOI: 10.17485/ijst/2016/v9i44/105159, November 2016* *ISSN (Print) : 0974-6846* *ISSN (Online) : 0974-5645*

Assessment of Ant Colony using Component based Software Engineering Metrics

Chander Diwaker[1*] and Pradeep Tomar[2]

[1]Department of CSE, U.I.E.T., Kurukshetra University, Kurukshetra - 136119, Haryana, India; chander_cd@rediffmail.com

[2]School of ICT, Gautam Buddha University, Greater Noida - 201312, Uttar Pradesh, India; parry.tomar@gmail.com

Abstract

Objectives: Conventionally, to solve a problem it is necessary to design a new system every time and that it is difficult to reuse existing system for the designing of new system. This paper estimates usability of reusable components and interaction between components for integration of system. **Methods:** To handle this kind of problem Component Based System (CBS) has been considered. The reusability concept is the main factor of Component Based Software Engineering (CBSE) for components. More usage of reusable components provides better results in terms of estimation of reliability and efficiency of Component Based System (CBS). Ant Colony Optimization (ACO) is one among soft computing technique which may be used to estimate the path followed by components and interaction of components. In this paper ACO is used as methodology to find out more reusable components to increase the reliability or efficiency of the system. MATLAB is used for implementation of ACO. Performance metrics like Component efficiency, component density and component dependency are used for the analysis of proposed work. **Findings**: The results shows that in proposed mechanism the component efficiency is high, component density is also high as compare to existing CBS. **Applications:** This paper may helpful for the analysis of soft computing techniques by the use of CBSE performance metrics.

Keywords: ACO, Component Based Software Engineering (CBSE), Component Based System (CBS), Reusability, Soft Computing and MATLAB

1. Introduction

CBSE is a novel movement in software engineering. CBSE metrics are useful in software development to create high quality software products within time by reusing the particular components. Many parameters like component consistency, quality, productivity and program design may be assessed using CBSE metrics. Reusability plays significant role in CBSE which results in fast development, easy in improvement, low cost and improved reliability of the system. The measurement of behavior of reusable components is an important task as well as complex task. There is need to develop new methods to check the compatibility of components for a particular environment. Components based software reliability depends on system architecture. The path usage by different components is an essential parameter to predict the reliability of CBS. CBSE provides methods, models and guidelines for the developers of component-based systems. Component based development involves a system that includes of significant use of various components. A path is a distance between components from one location to other location. The optimal path is the best computed path that helps in increasing reliability. There are different paths available for components interaction. To find optimal path i.e. most used path, difference soft computing techniques may be applied. ACO is one of them. In this paper in section 2 ACO with working has been discussed, in section 3 literature review of number of research paper related to proposed work has been presented, in section 4

Assessment of ACO has been presented, in section 5 results and analysis has been presented.

*Author for correspondence

2. Ant Colony Optimization (ACO)

ACO involves the behavioral nature of ant species. These ants leave pheromone in the path. The path is considered as favorable path that will be followed by other associated ants of the colony[1]. ACO working may be applied to various problems to solve optimization problems. Movement of ants in open space for searching food is shown in Figure 1. The main components of ACO include ants, pheromones and food/target.

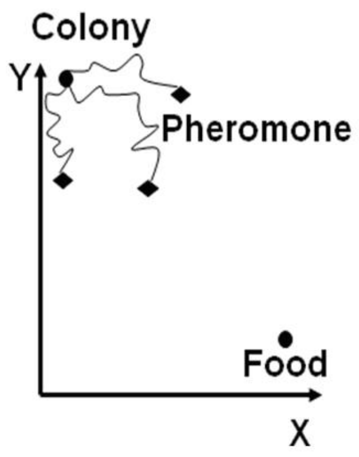

Figure 1. Various positions of Ants in ACO.

2.1 Application of ACO

ACO may be applied to solve various following problems:

- NP Hard problem
- Scheduling Problems
- Routing problems
- Assignment problems
- Vehicle routing problems
- Real World problems
- Multi objective Applications
- Stochastic Environment
- Quadratic Assignment Problem
- Traveling Salesman Problem
- Network Model Problem

2.2 Limitation of PSO

The limitations of PSO are as follows:

- Less accuracy due to different direction and different motion of particles.
- PSO may not successfully work where contact scattering of particles motion is requires.
- PSO may cause problem in non-coordinate environment.

2.3 Benefits of ACO

The advantages of ACO are as follows:

- Natural parallelism.
- Quick finding of good quality result.
- ACO may be applied where many components having random behavior.

2.4 Working of ACO

The working principle of ACO has been discussed with help of DFD as follows:

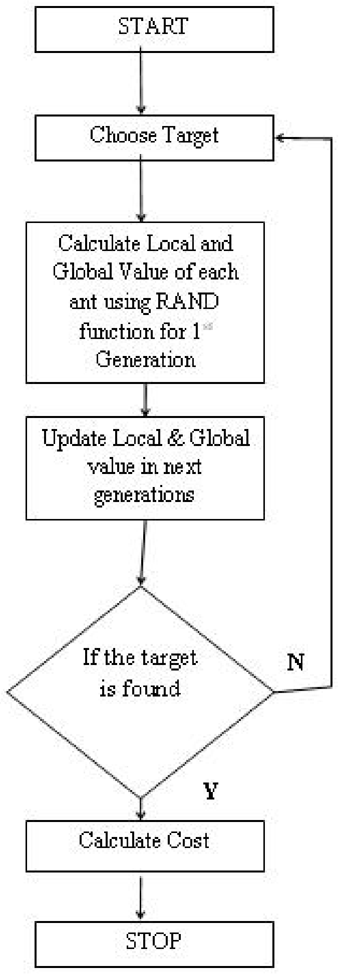
START
Choose Target
Calculate Local and Global Value of each ant using RAND function for 1st Generation
Update Local & Global value in next generations
If the target is found
N
Y
Calculate Cost
STOP

Figure 2. DFD for working of ACO.

In Figure 2, the first step involves, initialize numbers of Ants with selecting a random target. In next step, 1st generation is started and when ant moves from starting position to target position, during this period ants compute individual local value and a common global value.

In next generation, update ant's local value and global value. Repeat these steps, till the target is achieved. When the target is achieved, compute cost of path. On the basis of above calculation, select optimal path.

3. Related Work

In this section, a literature review of number of research papers that covers ACO, PSO, and CBSE with their pros and cons is presented.

In[2] proposed Differential Search Algorithm (DSA) to resolve sharing system dependability optimization problem. DSA was compared with various evolution techniques like PSO, Differential Evolutionary Algorithm (DEA), GA, ACO and Gravitational Search Algorithm (GSA). The outcome shows better performance, quality solution and efficiency.

In[3] proposed SDLC model to reduce software development cost in with the help of component based software engineering methodology.

In[4] proposed a scheme for flexibility of effective migration by dynamically adjusting migration period and migration cost. The scheme included two different algorithms in which the first dynamically modify the migration period based on resolution value of each sub-colony and the second assume both period and cost by the response from other sub-colonies. These algorithms provided better outcomes. The future work may be carried on by finding some target.

In[5] proposed algorithm based on ACO which produced set of most favorable paths and ranking the paths. This scheme produced test data series inside the province that was used as input of produced paths. This scheme covered complete software with lowest amount of redundancy.

In[6] proposed search engine executing the boundary approach depends on ACO named ACOR for persisted problems. The paper describes ACOR for constrained statistical optimization troubles and finding the brunt of parameter on the performance of the proposed algorithm. The modified ACOR was compared with standard ACO for persisted problems.

In[7] compared ACO and Bee Colony Optimization (BCO) with their acceptable variants. The basic principle, applications and the features of ACO and PSO were also discussed.

In[8] proposed feature selection technique on the basis of Ant Colony Optimization. In this paper multiple iterations are applied to extract best feature suite and reduce redundancy among similar features.

In[9] Ant Colony Optimization (ACO) and Chaos Optimization Algorithm (COA) are applied on real datasets of NASA to test the reliability of a system.

In[10] discussed cluster based selection process and cluster analysis that helps in component selection. The selection of components was analyzed for different requirement.

In[11] discussed a structural model for component presentation named as suffix tree that form a tree model. This model consists of various levels. The efficiency is based on the pattern requirement.

In[12] proposed a model for evaluating CBS reliability using path usage method. The path usage method consists of ACO technique. The path usage is shown with help of heuristic component dependency graphs. The model is based on ACOREL algorithm. The proposed scheme depends on CBS attributes.

In[13] developed a software reuse warehouse with data mining hierarchical clustering technique. It was assessed for reusability forecast of object oriented software systems. This technique showed better precision value to identify object supported reusable unit from the existing repository of software elements.

In[14] discussed reusability issues like reusable components and procedure of component recovery. The component recovery mechanism using GA and AC were also discussed.

In[15] focused on the relative examining of Swarm Intelligence: ACO and PSO. A relative analysis is carried out with fitness sharing to check the performance of these techniques for future progressive algorithms.

4. Assessment of ACO

In ACO, pheromone plays important role in finding optimal path. In CBS, finding optimal path between components is a challenging task. Therefore, assess ACO with CSE metrics like efficiency, component dependency and component density.

Performance metrics: Component Dependency, Component Density, Efficiency

Tool used: MATLAB2014 is used for analysis purpose. It is open source software which can be run on any platform.

5. Results and Analysis

The following performance metrics are used for the assessment of ACO.

5.1 Efficiency of Component

The efficiency defines the percentage of the ration of components required to total number of components. It is observed from Table 1, Table 2 and Figure 3 that efficiency has improved by using ACO for optimization of retrieved component with target search.

Table 1. Efficiency value with ACO

Sr. No.	Search Value	Efficiency
1	4	74%
2	7	61%
3	8	59%
4	3	21%
5	10	14%

Table 2. Efficiency value without ACO

Sr. No.	Search Value	Efficiency
1	4	60%
2	7	55%
3	8	14%
4	3	59%
5	10	3%

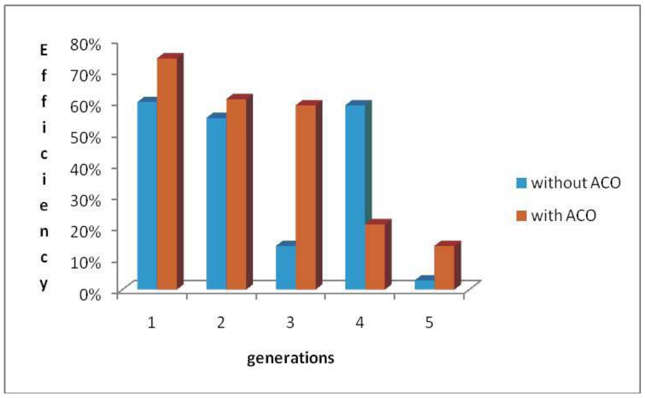

Figure 3. Efficiency v/s generations.

5.2 Component Density

It is the ration of definite number of interface to the accessible number of interfaces.

Table 3.Component density value

Sr. No.	Without ACO	With ACO
1	0.2347	0.4063
2	0.2525	0.4527
3	0.2653	0.4803
4	0.2515	0.5483
5	0.2643	0.5207

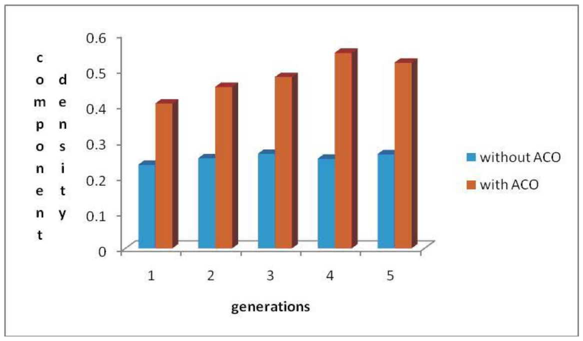

Figure 4. Component density v/s generations.

As shown in Table 3 and Figure 4, the componnent density is reduced as interfaces between components is low.

5.3 Component Dependency

The dependency presents the dependability of one component interaction on another component interaction. The component may a function or module which is required at the time of component/system integration. Dependency between components is high in case of ACO. Table 4 and Figure 5 show the resultant values.

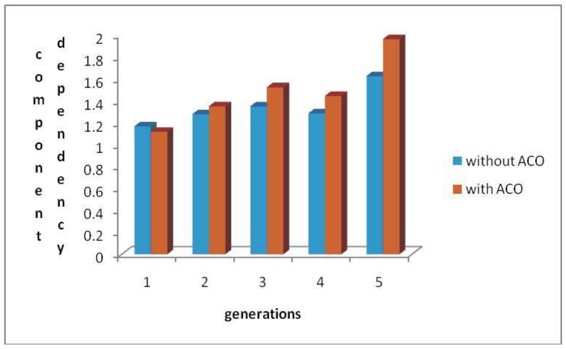

Figure 5. Component dependency v/s generations.

Table 4. Components dependency values

Sr. No.	Without ACO	With ACO
1	1.1667	1.1176
2	1.2778	1.35
3	1.35	1.5238
4	1.2857	1.4444
5	1.625	1.963

6. Conclusion

In this paper, a soft computing technique: ACO is used to find the optimal path between components of CBS. The working of ACO with the help of flow chart is also discussed. The results show that efficiency of interaction between components increases when applying ACO. In future, a new reliability framework may be proposed and analyzed using real data sets.

7. References

1. Katiyar S, Nasiruddin II, Abdul Ansari Q. Ant colony optimization: A tutorial review. MR International Journal of Engineering and Technology. 2015; 7(2):35-41.
2. Ray S, Bhattacharya A, Bhattacharjee S. Optimal placement of switches in a radial distribution network for reliability improvement. International Journal of Electrical Power and Energy Systems. 2016 Mar; 76:53–68.
3. Jain P. Towards the adoption of modern software development approach: component based software engineering. Indian Journal of Science and Technology. 2016 Aug; 9(32):1-5.
4. Hong TP, Huang LI, Lin WY, Liu YY, Chakraborty G. Dynamic migration in multiple ant colonies. Proceedings of 2nd International Conference on Cybernetics; Gdynia. 2015. p. 146–50.
5. Biswas S, Kaiser MS, Mamun SA. Applying ant colony optimization in software testing to generate prioritized optimal path and test data. Proceeding of 2nd International Conference on Electrical Engineering and Information Communication Technology; Dhaka. 2015. p. 1-6.
6. Ibnez ML, Stutzle T, Dorigo M. Ant colony optimization: A component-wise overview. IRIDIA Technical Report Series. 2015. p. 1-41.
7. Kaur I, Kaur S. Software component retrieval using GA and ACO. International Journal of Advanced Research in Computer Science and Software Engineering. 2015 Jul; 5(7):212-5.
8. Sabeena S, Sarojini B. Optimal feature subset selection using ant colony optimization. Indian Journal of Science and Technology. 2015 Dec; 8(35):1-5.
9. Dizaji ZA, Gharehchopogh FS. A hybrid of ant colony optimization and chaos optimization algorithms approach for software cost estimation. Indian Journal of Science and Technology. 2015 Jan; 8(2):128-33.
10. Madaan N, Kaur J. A survey on selection techniques of component based software. International Journal of Advanced Computation Technology. 2014; 4(13):1245-50.
11. Sandhu K, Gaba T. A novel technique for components retrieval from repositories. International Journal of Advanced Computer Technology. 2014; 3(6):912-20.
12. Tyagi K, Sharma A. A heuristic model for estimating component-based software system reliability using ant colony optimization. World Applied Sciences Journal. 2014; 31(11):1983-91.
13. Niranjan P, Shireesha P, Reddy MV. Development of reuse repository and software component performance analysis. International Journal of Application or Innovation in Engineering and Management. 2013 Jun; 2(6):473-7.
14. Chikhalikar A, Darade A. Swarm intelligence techniques: Comparative study of ACO and BCO. Journal of Computer Society of India, Mumbai. 2013 Apr:1-9.
15. Selvi V, Umarani R. Comparative analysis of ant colony and particle swarm optimization techniques. International Journal of Computer Applications. 2010; 5(4):1-6.