ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 2, February 2013

A Review on Concurrency Fault Detection Techniques

C. Revathi, M. Mythily

***Abstract*— The UML is the most common language that is used for system modeling. But, this language has been designed as a general purpose modeling language that might need modeling constructs for the specific real time embedded (RTE) domain. To fill this lack, OMG has standardized a UML addition, called MARTE. Domain specification UML provides a special way called a profile. The UML profile for MARTE is a concentration of UML that provides idea devoted to real time modeling for design and analysis of real time application and platforms. The existing work, can detect only the concurrency problem such as deadlock and starvation. But, the UML / MARTE profile is specifically designed with the genetic algorithms to detect other concurrency issues like data races. This paper reviews the performance of the search space size and concurrency detection techniques.**

***Index Terms*—Genetic Algorithm, Hill Climbing,MARTE Profile,Random Search**

I. Introduction

Concurrency problem should be identified in the design phase of software Enginerring process.It is made progressively difficult in larger and more complex systems. The finding of concurrency issues is based on the design models articulated in UML. Once the UML representation is not sufficient to completely model a system for a particular purpose, the representation is extended by profiles. The adjustment of the MARTE (Modeling and Analysis of Real Time and Embedded Systems) profile [1] addresses domain specific parts of real time concurrent system modeling. The objective of this paper is to detect several types of concurrency errors (such as deadlocks, starvation, and data races, data flow problems) and that can be simply combined into a Model Driven Architecture (MDD) approach, an OMG standard by the UML based MDD [2]. A genetic algorithm (GA) is tailored to identify different types of concurrency issues.

The existing work uses genetic algorithm to delict deadlocks [3] and starvation [4] into a composed form. GA method can also detect the data races .The fitness function exactly designed to detect deadline and starvation, correspondingly. Now, fitness functions needed towards data race detection and to improve the performance comparison.

The next section presents a comparative study to measure the performance and also compares with a hill climbing search and random search.

Manuscript received Feb, 2013.

***C.Revathi**,Department of Computer Science and Engineering, karunya University,Coimbatore,India,*

***M.Mythily**,Asst.Prof,Department of Computer Science and Engineering, Karunya University,Coimbatore,India.*

II. Comparative Study

This section compares the three techniques: Random Search (RS), Hill Climbing (HC) and Genetic Algorithm (GA).These three techniques can be used to detect the concurrency issues like deadlocks and starvation.The three techniques have been compared based on performance,execution time and search space.

A. Random Search (RS)

The search space, a point is randomly selected and checked for a concurrency fault. Deadlock detection of the random search techniques is capable of discovering a fault, but with very less probabilities. RS can be applied to the cruise design model [4]. The cruise model is used at very small search space. The random search is likely to be as useful as the GA. Random search also detects the starvation. Here, the random search techniques can be useful for the two design models ModPhil (Modified Dining Philosophers problem) and ModCruise. The starvation detection rate is very high, because the ModPhil have the highest search space and ModCruise has a smaller search space size. So, the random search cannot detect the concurrency problem in small search space.

Random generation is random and checked for data races in a point in the search space. Complexity is not an issue in random search as information about the landscape of the search space is not used through out the search. Still, random search performs poorly in MEOS [5]. Running a pre-determined space involves running a random search, usually the search space has large number of points. The main disadvantage of RS is that poorly detects the deadlock and also starvation detection rate is very low compared to other two techniques. So, RS can be used only in small search space. Advantages of the RS are good response time and starvation can be detected in any search space .

B. Hill Climbing (HC)

Hill climbing is an optimization technique that works based on randomly chosen candidates. It identifies a set of neighbourhoods depending upon the problem that it represents. A closer move to the neighbour improves the fitness value of the problem solution.

In existing work, HC techniques can be used by the stochastic hill climbing [6]. One random point and mutating the current point generated via neighboring point. It exchanges it and a new adjacent point is generated. The stopping criteria point is nothing better than the current point, then the execution can be stopped. This continuous fills process a maximum number of classifications needed. In [7], HC worked by randomly choosing a candidate solution and separate iteration the element of a set of near neighboring to the existing solution are considered. These techniques have a crucial error with the HC testing is that the Hill pointed out by the Algorithm may be local maxima. This can be for interior then global maxima in the search space.

HC does not perform well in the some of the particular design models. It is powerless to detect a deadlock and variance comparing to which genetic algorithm is mathematically significant. Since its execution depends on the first randomly generated chromosome. The GA was analyzed and generated by the number of sequences, and the HC uses the same number of sequences always be generated.If, the optimal chromosome is very different from the initial hill climbing chromosome. Therefore, HC might tired to allocate the number of sequences. HC total detection rate does not differ much in Phil and tends to be poor overall.

With respect to starvation, all three techniques are also capable of detecting this concurrency problem. Here, the detection rate of the starvation is the better than the two design model i.e. ModPhil and ModCruise [5] [4]. But, the detection rate of date race is very worst performance by HC techniques. Both hill climbing and the genetic algorithm is significantly better in cost reduction [8]. The main advantage of these techniques are execution time is very low. And the disadvantages are the detection rate of the deadlock and data races is very low in the large search space.

Table 1. Comparison of different techniques

Techniques	Fault Types		
	Deadlock	Starvation	Data Races
RS	**	***	*
GA	****	****	****
HC	*	***	*

Fair (*), Good (**), Less Efficient (***), More Efficient (****)

C. Genetic Algorithms (GA)

Genetic Algorithm (GA) is metaheuristic techniques. GA algorithm belongs to the large group of Evolutionary Algorithm (EA) and generate solutions to optimization problems using techniques motivated via natural evolution such as inheritance, mutation ,selection and crossover.

The GA can be detected to the concurrency problem such as deadlock and starvation. GA not only detects the deadlock and starvation, also detect the other concurrency issues like data races. The work by Mahfoud and Goldberg GA is established to produce much better results than the other two techniques [12]. The three techniques proceeds in a different way. The GA examined the number of sequences generated and generated the same number of other two techniques that is HC and RS to make sure a fair comparison [5]. Shousha, Braind and Labiche can be detect the deadlock and starvation [4].These also supports a UML/MARTE profile to detect the starvation and deadlock. GA has a higher detection rate for deadlock and starvation when compared to HC and RS. GA has the better search space solution and fitness value. Comparsion of three search techniques have been depicted in Table 2 and Table 1

Table 2. Comparison of Merits and Demerits

Techniques	Merits	Demerits
RS	Execution time is very less. The detection rate of the starvation is modualt.	Deadlock and starvation detection rate is very poor eventhough it use, of MARTE profile.
GA	Deadlock and starvation detection rate is more efficient. GA also detects the other concurrency issues like data races. The result of the data races detection also more efficient.	Total execution time is high. The cost of reduction also high.
HC	Total execution time is very less. HC can be detecting the issues in large and small search space.	Deadlock and data races detection rate is very poor performance in the search space.

III. Supporting Profiles

A. UML/SPT profile

The UML Profile for Schedulability, Performance and Time (SPT) that have been established to be curable. In [9] describe problems and possible solutions related to the usage of the profile in the representation of Schedulability analysis models for real-time distributed systems.

The work in [10] determines the new Request for Proposals issued through OMG for a new UML Profile named "Modeling and Analysis of Real-Time and Embedded systems". In this[10] explains first a few domain concepts for annotating Nonfunctional Properties (NFPs) and focuses on supporting temporal verification of UML-based models. The importance is given to Schedulability and performance analysis for real-time systems.

The early design process can only identify concurrency problems such as deadlock [3] .But MARTE method based on the analysis of particular models expressed in the UML (Unified Modeling language) that uses an exclusive designed GA to detect deadlocks. All related concurrency information is excerpted from the system UML models that fulfill with the UML Schedulability; Performance and Time profile, a standardized specialization of UML for real-time concurrency system. CFD tool is supporting the methodology. The CFD is decayed into three components: scheduler, GA and RAG evaluator. The SPT sub profiles are RTconcurrency modeling , SAprofile and RTresources modeling. Deadlock detection is performed using RAG evaluator .A chromosome results into starvation if at least two threads waiting on locks. If a cycle is present, CFD tool will delicts the deadlock. If the cycle is not present the deadlock cannot do detect.

B. UML/MARTE profile

In this profile the UML provides a common extension system for customizing UML models for particular domains and platforms. Extension system allows improving standard semantics in exact additive fashion, so that they can't contradict the standard semantics. Profiles are defined using stereotypes, tag definition and constraints .In Table.3.comparsion of stereotypes and tag definitions that are applied to specific model elements such as class, Attributes, operation and activities are shown.

The MARTE profile is a replacement of the Schedulability; Performance and Time (SPT) profile [9]. MARTE is tackled for both the real time and embedded system domain. The profile is roughly divided into two subdivisions: the MARTE design model and the MARTE analysis model.

In existing work, first UML/MARTE based methodology for executable RTE systems modeling and another one, a framework and its underlying model transformations required to execute UML models conforming to the MARTE standard [11]. In this paper proposed by Accord||UML methodology to play the role of that structuring framework. An important aspect of models exploitation is to enable their executability. The Accord framework provides the infrastructure for MARTE model execution. The main disadvantages of Accord execution platform could be parameterized with a subset of MARTE concepts properties but do not support all parameter values, hence this introduces some restrictions on usage of MARTE.

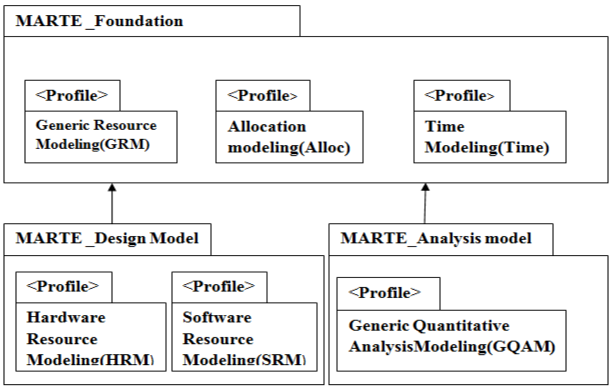

Fig 1. MARTE Profile

In, [4] [3] UML/MARTE profile detect the concurrency problem such as deadlock and starvation. Here, CFD tool can detect the fault.As,illustrated in Fig 1,The profile can be divied to the two subdivision :the MARTE Design model and MARTE analysis model.The design model can be define by models various features of real time and embedded system.The analysis model is used for the purpose of system analysis.These two subdivision are based on the MARTE foundation.the MARTE foundation is defines the time concepts and use of concurrent resources allocation.The concepts are first obtained from the foundation of MARTE, namely, the Generic Resource Molding (GRM) packages.The GRM introduces two stereotypes <<Acquire>>, <<Release>>. The Software Resource Modeling (SRM) is the sub profile of GRM. This presents for designing multitasking application. SRM divided into four packages: SW_Resource Core (resource concepts),SW-Courrrency(concurrent concepts), SW-interaction (communication and synchronization) and SW-Brokering(resource management). Another sub profile is Generic Quantitative Analysis Modeling (GQAM) sub profile.This subprofile defines stereotype <<SaStep>>. Its tag includes priority,interOccTime and execution time.Maximum and Mininum time ranges can be used specified to the execution time.

Table 3. Comparison of SPT and MARTE profile

Parameters	SPT	MARTE
Input [3] [4] [5]	[T , L, a] (T is a thread , L-is a Lock, a is a specific time unit)	[T, L, a] and [T, a]
Fault Type	Deadlock [3]	Deadlock and starvation [4] and other concurrency issues like data races. [5]
Tool	CFD	CFD (concurrency Fault Detector)
Algorithms	GA [3]	GA, HS, RS. [4] [5]. GA is better than the other two techniques.
Stereotype/ tag	<<CRconcurrent>> ,<<SArecource>>, <<SAaction>>, <<GRMacquire>>, <<GRMrelease>>	<<SWconcrrentResource>>, <<Acquire>> <<Release>>, <<SWMutualExclusionResource>>

IV. A UML/MARTE MODEL ANALYSIS METHOD FOR CONCURRENT SYSTEM

The UML/SPT profile [3] proposed by cannot detect the starvation and data races. Here, the CFD (Concurrency Fault Detector) tool can be detects deadlock based on the analysis of design description in UML and SPT.

The UML/MARTE profile can be detecting the other concurrency problems such as deadlock and starvation, data races, data flow. In [5] [4] [3]detects the deadlock , starvation and also data races.The UML/MARTE profile can be used to CFD tool. This tool can be detecting the fault accurately. Here, GA is the better performance to other than two techniques (i.e. HC, RS). Because; the concurrency problem (deadlock, starvation and data races) detection rate is high performance in the search space.

In existing work, [5] the data race can be detected by the UML/MARTE sequence diagram of the shared resources in Therac. Three optimization techniques are capable of detecting data races in both MEOS and DECF (resources), but with extremely different probabilities. CFD is an automated system that identifies concurrency problems than any concurrent function modeled with the UML/MARTE schema. Here, GA is well known to produce much better results than the other two techniques. The main advantages are the detection rate of the deadlock ,starvation and data races..GA is better than all other techniques because the total execution time is less a cost comparison to other two techniques.

It , describe a methodology and tool for detecting deadlocks and starvation based on the analysis of design representation in UML and MARTE profile. A UML/MARTE model analysis method can be detecting the other concurrency problems like data races.Here, also support the CFD tool.CFD is decomposed into three portions as a scheduler,GA and RAG evaluator.

A. CFD scheduler

Concurrency Fault Detector CFD is the tool used as an automated system that identifies concurrency faults in any concurrent modeled application. The CFD is categorized into 3 parts namely Scheduler, Genetic Algorithm, RAG Evaluation. The work of scheduler is when a system is designed; assumptions are made about the architecture it will be run on. And the deployment assumptions are incorporated in CFD in the form of the scheduler. Scheduler currently emulates single processor execution and is POSIX compliant.

A. RAG evaluation(Deadlock Detection)

If a chromosome result in waiting on locks Deadlock and starvation detection is performed using a RAG. If a cycle is found, CFD outputs the details of the chromosome causing it (executing threads and waiting threads for each lock as well as lock access times), the corresponding RAG, and the fitness value. If no deadlock is found, CFD terminates, showing both the fitness value and output details of the highest fitness chromosome found.

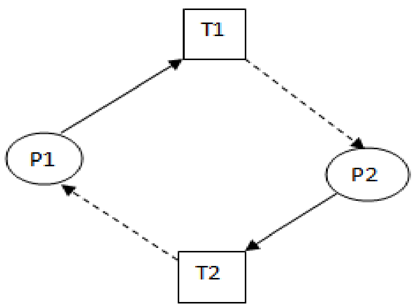

Fig 2. Deadlock

For example Fig 2.if the cycle is present the deadlock can be occur.Here,T1 and T2 represent by the resources and P1 and P2(process).P1 request T1(indicated by the solid arrow)and T1 held by P2(indicated by the dotted arrow),similarly P2 request by T2 ,but T2 currently held by P1. This situation represents the occurrence of deadlock.

B. Genetic Algorithm

Genetic algorithms operate on a set of possible solutions. Because of the random nature of genetic algorithms, solutions found by an algorithm can be good, poor, or infeasible [defective, erroneous], so there should be a way to specify how good that solution is. This is done by assigning a fitness value [or just fitness] to the solution. Chromosomes represent solutions within the genetic algorithm. The two basic components of chromosomes are the coded solution and its fitness value. To use a genetic algorithm, you must represent a solution to your problem as a genome (or chromosome). The genetic algorithm then creates a population of solutions and applies genetic operators such as mutation and crossover to evolve the solutions in order to find the best one(s).

C. Chromosome representation

A chromosome is collected of genes and perfects a solution to the optimization problem.A gene can be described as a three tuples(T,L,a),where T-thread,L-Lock,a-specific time unit when T accesses L.the optimization is done on access times of threads two locks.Under the testing environment time interval should start with start time of system to detect the starvation.But the verification cost more.To avoid this user defined heuristics developed for verification followed by reduced the time interval

D. Crossover Operator

They are three constructive that have to be met for the formation of valid chromosome after performing the crossover operation.The three constraints are :1) All genes within the chromosome are ordered according to increasing thread identifiers,then lock identifiers,then increasing access time.2)Lock access times must fall within the specified time interval or are set to -1.3) Consecutive genes for the same thread and lock identifiers must have access time differences equal to at least the minimum and at most the maximum lock access range of the associated thread and lock, if start and end times are defined as ranges.

After performing the crossover if constraint 3 is not met for any of the constraint genes ,then the second genes access time is replaced with a randomly chosen access time from a set of possible access times.This is repeated until all the genes meet constraint 3.

E. Mutation

The mutation operator operates by altering the access time of the genes .The access times are moved within a specified time intervals with the aim of finding at the optimal lock access time.so that it will not cause starvation .If the chosen value lies outside the time interval the access times is set -1.

V. CONCLUSION

Concurrency abounds in much software system, where systems usually include threads that access shared resources and difficult thread communication. If not handled accurately, such access can deadlock and starvation situation, which might delay system execution. In existing work, HC and RS can be detecting the concurrency problem such as deadlock and starvation. These two algorithm performance is very less efficient. So, in this paper, describe a methodology and tool for detecting deadlocks and starvation based on the analysis of design representation in UML and MARTE profile. A UML/MARTE model analysis method can be detecting the other concurrency problems like data races.MARTE provides a support for modeling the real time embed systems.MARTE focuses on performance and scheduling. GA is better than the other two techniques. Because, the detection rate of the GA performance is more efficient. So, we can conclude that the genetic algorithm is efficiently performed as well or better than the two other alternatives and the difference are driven by the search space size and complexity of the search space.

REFERENCES

[1] OMG: UML Profile for Modeling and Analysis of Real-time and Embedded Systems. 1.0 Beta

[2] Kleppe A., Warmer J. and Bast W., MDA Explained - The Model Driven Architecture: Practice and Promise, Addison-Wesley, 2003.

[3] Shousha, M., Briand, L., and Labiche, Y.: "A UML/SPT model analysis methodology for concurrent systems based on genetic algorithms‖ . *Proc. ACM/IEEE MODELS*, 475-489, (2008).

[4] Shousha, M., Briand, L.C., Labiche, Y., "A UML/MARTE Model Analysis Methodology for Detection of Starvation and Deadlocks in Concurrent Systems," Carleton University, Technical Report SCE-09-01, squall.sce.carleton.ca, 2009.

[5] Shousha, M., Briand, L.C., Labiche, Y., "A UML/MARTE Model Analysis Method for Detection of Data Races in Concurrent Systems," Carleton University, Technical Report SCE-09-01, squall.sce.carleton.ca, 2009.

[6] A. Rosete-Sua´rez, A. Ochoa-Rodrı´guez, and M. Sebag, "Automatic Graph Drawing and Stochastic Hill Climbing," Proc. Genetic and Evolutionary Computation Conf., vol. 2, pp. 1699-1706, 1999.

[7] M. Harman, "The Current State and Future of Search Based Software Engineering," Proc. Int'l Conf Software Eng., pp. 342-357, 2007.

[8] M. Harman and P. McMinn, "A theoretical empirical analysis of evolutionary testing and hill climbing for structural test data generation," in *Proceedings of the 2007 International Symposium on Software Testing and Analysis (ISSTA '07)* London, United King-dom: ACM, 2007.

[9] OMG, UML Profile for Schedulability, Performance and Time Specification, http://www.omg.org/docs/formal/05-01-02.pdf.2005.

[10] D.C. Petriu, "Performance Analysis with the SPT Profile," Model-Driven Eng. for Distributed and Embedded Systems, pp. 205-224,Hermes Science Publishing Ltd., 2005.

[11] C. Mradiha, Y. Tanguy, C. Jouvray, F. Terrier, and S. Gerard, "An Execution Framework for MARTE-Based Models," Proc. 13th IEEE Int'l Conf. Eng. of Complex Computer Systems, pp. 222-227, 2008.

[12] Mahfoud, S. W. and Goldberg, D. E.: "Parallel recombinative simulated annealing: a genetic algorithm". *Parallel Comp*. Vol 21, No. 1, 1-28. (1995).

C.Revathi has completed her B.E degree in Computer Science and Engineering at srinivasan engineering college in the year 2011 and is currently doing M.Tech in the Department of Computer Science and Engineering, Karunya University, Coimbatore, India.

Mrs.M. Mythily has completed her master's degree with a specialization of computer science and engineering at Government college of Technology, Coimbatore.Currently, She is pursuing her Ph.d degree with a specialization of software engineering at Anna University, Coimbatore .At present working as an Assistant Professor at karunya University with seven years of teaching interest .Her area of interest are Software Engineering and Operating System.