Optimizing Resource Usage in Component-Based Real-Time Systems - Appendix

Johan Fredriksson, Kristian Sandström, and Mikael Åkerholm

Mälardalen Real-Time Research Centre,
Department of Computer Science and Engineering,
Mälardalen University, Box 883, Västerås, Sweden,
http://www.mrtc.mdh.se
johan.fredriksson@mdh.se

Abstract. This report presents data regarding simulation setup, the genetic algorithms setup and a short description to the use of the genetic algorithms. This report is written in the context of the paper *Optimizing Resource Usage in Component-Based Real-Time Systems*.

Appendix

This technical report is written in the context of [1], and should not be considered as a stand alone report.

Simulation set up

This section describes the simulation method and set up. For each simulation the genetic algorithm (GA) assigns components to tasks and evaluates the allocation, and incrementally finds new allocations. The evaluation is performed in a number steps:

1. System data - components and transactions with deadlines are created. There exists at least one solution for all data that are passed on to the GA.
2. Initial Population - The GA creates a random population that makes up a set of allocations. One population comprises several chromosomes, and each chromosome represents an allocation.
3. Apply Fitness function - The fitness function calculates how fit a chromosome is. The higher fitness value, the more likely is the chromosome to be passed on to the next generation.
4. Create New population - The GA combines different chromosomes, and performs mutations by reassigning one or several components.
5. Repeat from step 3, each iteration is referred to as a generation.

The system data is produced by creating a random schedulable task set, on which all components are randomly allocated. The component properties are deduced from the task they are allocated. Transactions are deduced the same way from the task set. In this way it is always at least one solution for each system. However, it is not sure that all systems are solvable with a one-to-one allocation. The components and component transactions are used as input to the framework. Hereafter, systems that are referred to as generated systems are generated to form input to the framework. Systems that come out of the framework are referred to as allocated systems. The simulation parameters are set up as follows:

- The number of components of a system is randomly selected from a number of predefined sets. The numbers of components in the systems are ranging in twenty steps from 40 to 400, with a main point on 120 components.
- The period times for the components are randomly selected from a predefined set of different periods between 10 and 100 ms.
- The worst case execution time (WCET) is specified as a percentage of the period time and chosen from a predefined set. The WCETs together with the periods in the system constitutes the system load.
- The transaction size is the size of the generated transactions in percentage of the number of components in the system. The transaction size is randomly chosen from a predefined set. The longer the transactions, the more constraints, regarding schedulability, on how components may be allocated.
- The transaction deadline laxity is the percentage of the lowest possible transaction deadline for the generated system. The transaction deadline laxity is evenly distributed among all generated systems and is always greater or equal to one, to guarantee that the generated system is possible to map. The higher the laxity, the less constrained transaction deadlines.

One component can be involved in more than one transaction, resulting in more constraints in terms of timing. The probability that a component is participating in two transactions is set to 50% for all systems.

To get as realistic systems to simulate as possible, the values used to generate systems are gathered from some of our industrial partners. The industrial partners chosen are active within the vehicular embedded system segment. The task switch time used for the system is 22 μs, and the tcb size is 300 bytes. The task switch time and tcb size are representative of commercial RTOS tcb sizes and context switch times for common CPUs. The simulations are performed for four different utilization levels, 30%, 50%, 70% and 90%. For each level of utilization 1000 different systems are generated with the parameters presented above.

In this appendix we show the specific data used for generating systems to the simulations. The software is implemented using java, and for the basic structures and operations of the genetic algorithm the open gnu library Gajit, written by Mathew Faupel. The GA was setup with an initial population of 300 individuals, and every simulation was run for 500 generations. The simulations were run on a 1.8 GHz Pentium 4m processor with 768 MB of RAM. The mean time for each simulation is 133 seconds. The parameters used for the GA in the experiments are shown in table 1.

Param on component level		WCET in % of period	Dist. %
Number components	Dist. %	2	45
40	1,25	4	50
50	6,25	8	5
60	10	Stack size	Dist. %
70	6,25	256	10
80	2,75	512	25
100	7,5	1024	25
120	13	2048	35
140	7,5	4096	10
150	5	Param on System level	
160	2,5	ctr. size % of num. comp.	Dist. %
180	8	10	10
200	5,25	13	25
210	5	17	25
240	9	21	25
250	1,25	25	15
280	5	Laxity % of ctr.dl	Dist. %
300	2	110	33
320	1	130	33
350	1,25	150	33
400	0,25	Utilization %	Dist. %
Isolation %	Dist. %	30	25
0	20	50	25
10	30	70	25
20	30	90	25
30	20	GA parameters	
Period time (s)	Dist. %	GA property	Value
10000	20	Population	300
25000	20	Generations	500
50000	40	Elite rate	5%
100000	20	Cull rate	40%
		Mutation rate	1%

Table 1. Data used for generating systems, and GA parameter

References

1. Fredriksson, J., Sandström, K., Åkerholm, M.: Optimizing resource usage in component-based real-time systems. In: Proceeding of CBSE8 International Symposium on Component-based Software Engineering, St.Louis, US, IEEE (2005)