1-year project report for ARTES project 9811-3.

Identification of Complexity-Reduction Techniques for Optimal Scheduling in Embedded Distributed Real-Time Systems

Jan Jonsson

Department of Computer Engineering
Chalmers University of Technology
SE–412 96 Göteborg, Sweden
janjo@ce.chalmers.se

Mikael Strömberg

Mecel AB
Box 140 44
SE–400 20 Göteborg, Sweden
mikael.stromberg@mecel.se

June 10, 2000

Summary

The application of optimal search strategies to scheduling for distributed real-time systems is, in general, plagued by an inherent computational complexity. This has effectively prevented the integration of search strategies such as branch-and-bound or constraint programming in scheduling frameworks and tools used in practice today. The integration of an optimal scheduling strategy could lead to, *e.g.*, higher application schedulability, better utilization of scarce resources, or higher system reliability, than can be attained for a sub-optimal scheduling technique. Inspired by recent research results, this ARTES project aims at demonstrating that optimal scheduling is, in fact, a viable alternative for many real-time scheduling scenarios. Our approach is based on the hypothesis that, with detailed knowledge about the real-time application and its characteristics, it is possible to make intelligent choices in the configuration of the search algorithm in such a way that the time it takes to generate an optimal assignment of tasks to system resources is reduced to a tractable value.

The project is a cooperation between two ARTES nodes: Department of Computer Engineering, Chalmers, Göteborg and Mecel AB, Göteborg, and one external node: Real-Time Computing Laboratory, University of Michigan, Ann Arbor. One senior researcher and one Licentiate/Ph.D. student from Chalmers will be actively working on the project. In addition, a scientist at Mecel AB will contribute to the project with expertise in design techniques for distributed vehicular real-time applications and by providing examples and case studies.

1 Problem Statement

In recent decades, distributed real-time systems have emerged in many application domains including drive-by-wire automotive systems, intelligent traffic control systems, banking systems, and autonomous robots. Depending on application domain, one may strive for an optimal or a sub-optimal solution to the task assignment and scheduling problem. For embedded, safety-critical real-time systems, where schedules must be generated *a priori*, optimal scheduling is an attractive choice because it leads to (depending on scheduling objective) higher application schedulability, better utilization of scarce resources, or higher reliability, than does a sub-optimal scheduling technique. For example, a high utilization of the system hardware is extremely important in high-volume real-time systems (*e.g.*, vehicles) where the hardware cost is a dominating factor in the total product cost. In fact, one can foresee increasingly higher demands on the computing services that will be offered by future high-volume embedded systems. However, the requirements on such systems often result in conflicting design goals. In order to deliver high performance, it is essential to use as powerful hardware as possible; on the other hand, hardware cost and power dissipation must be kept at a minimum to allow for a large sales market and high mobility. This kind of multifaceted optimization criteria calls for effective scheduling techniques that can keep resource utilization at a maximum while still meeting other optimization criteria. Unfortunately, the state-of-the-art in industry has not reached this level of maturity, but often rely on strategies based primarily on intuition and tradition. For future distributed systems with more tasks and resources to handle, human intuition no longer suffices to find the most cost-effective solution for a given optimization criterion. It is, thus, necessary to use automated tools with intelligent heuristics to find good design solutions.

Among the optimal task assignment and scheduling techniques, the Branch-and-Bound (B&B) and Constraint Programming (CP) strategies have been successfully used within different application fields. Unfortunately, scheduling of real-time applications with the objective of finding an optimal solution is, in the general case, hampered by an inherent computational complexity[1]. This has effectively prevented an adaption of optimal strategies in scheduling frameworks and tools used in practice today. To investigate whether a practical use of optimal search strategies, such as B&B or CP, indeed exists, it is important to assess the complexity of these strategies for real-world applications. Recent research on the B&B algorithm [1, 2, 3, 4] has indicated the viability of finding techniques that can offer significant complexity reduction[2] under the assumption of certain optimization criteria. While these techniques duly account for general application characteristics such as task execution times, task deadlines or the number of processors used in the system, they are not cognizant of constraints that represent other important characteristics of the application and its environment. Examples of such constraints are those that stem from demands on high system reliability (task replication constraints) or a low total system cost (minimize the number of communication links).

2 Main Ideas

Our approach is based on the hypothesis that, with detailed knowledge about the real-time application and its environment, it is possible to make intelligent choices in the configuration of the scheduling algorithm in such a way that the time it takes to find an optimal schedule is reduced to a tractable value. Examples of such configuration choices in the case of B&B are those that determine the order in which search-tree vertices are traversed [1], or those that identify and avoid redundant branches in

[1]In the case of the B&B algorithm, the search for an optimal solution is organized by means of a *search tree*, where the vertices in the tree represents partial and complete solutions to the problem. To measure the computational complexity of the algorithm, we use the *time complexity* of the B&B algorithm, that is, the number of vertices explored during the search for an optimal solution.

[2]Since the addressed search problem is known to be NP-hard, the intent of the project is to find techniques for complexity reduction *on the average*, using heuristics for embedded distributed systems that make optimal assignment techniques practically useful for a large fraction of the applications considered. In the case that the computational complexity for a specific application should be intractable, the designer can always resort to near-optimal heuristics with acceptable complexity. The development of such techniques is also part of the project. In particular, one of our goals is to find rules of thumb that can aid in choosing the correct technique for each specific application.

the search tree [2]. Other configurations deliver tractable performance by making explicit assumptions about the underlying task-scheduling policy [5].

Unfortunately, the techniques proposed to date in the open research literature all suffer from the same major drawback, namely that they assume a general application model with very limited semantics for describing specialized application constraints typically found in embedded computer systems. In such systems, the application and its behavior is well-known which allows for a large degree of fine-tuning as regards the scheduling algorithm used. Furthermore, the design optimization criteria used for such systems differ relatively much from that of general computer systems. Some examples of specific assumptions that can be made for an embedded system are listed below:

System constraints: These constraints are imposed by the selected hardware architecture of the system. How the hardware is composed depends on functional or economical reasons, which means that the system configuration is typically fixed. The configuration includes information on the number of processors and other resources in the system as well as their performance and capacities (for example, worst-case execution time, context-switch cost, or communication network delay).

Task timing constraints: These constraints are the restrictions that control the timing behavior of the tasks and concern both the behavior of a single task (for example, period, deadline and release time) as well as the interaction between tasks (for example, distance, freshness, and harmonicity).

Task execution constraints: These constraints are the restrictions that control the execution behavior of the tasks. Similar to the timing constraints, the execution constraints concern both the behavior of a single task (for example, resource usage, locality or preemption) as well as the interaction between two or tasks (for example, precedence, communication, exclusion, and clustering).

In order to convey the system requirements to the scheduling framework in a semantically-correct manner, it is necessary to use these constraints wisely [6, 7]. Since the scheduling framework is used to validate that the system requirements are actually met by the implementation, the methodology used for constructing constraints must be cognizant of the practical application domain as well as the theoretical scheduling domain. In our view, the constraints must therefore *(a) reflect the intended behaviour of the system*, *(b) be derived in natural way without overconstraining the system*, and *(c) be fully supported by the scheduling framework*.

3 Expected results and impact

The expected research contributions of this project are as follows:

C1. A scheduling framework with support for application constraints typically found in embedded distributed real-time systems. Existing frameworks only handle limited subsets of all existing constraints.

C2. A scheduling framework with support for multi-objective optimization criteria. Current frameworks either only considers a single optimization criteria, or they can use multiple objectives but are difficult to use in practice since they must be "calibrated" for each specific application.

C3. Heuristics for complexity reduction in the scheduling of embedded distributed real-time systems. Current work has focused on finding techniques that are generally applicable but not cognizant of application-specific constraints.

By applying our techniques to applications provided to us from Mecel, the research will demonstrate practical techniques to achieve high resource utilization and meet other design criteria for interesting test cases. Since embedded distributed real-time systems is a rapidly-emerging application domain, the research is expected to generate new design-support tools for these applications. Furthermore, the problem of scheduling tasks on distributed processors is solved predominantly "by hand" in industry today. With the aid of a well-functioning automated B&B or CP framework, the scheduling process will be less error prone and will lead to more cost-effective[3] design solutions.

4 Project plan and results

4.1 Original plan and recent updates

The first phase of the project has been concerned with the analysis of distributed real-time applications and the identification of their constraints and optimization criteria. This investigation has been carried out by the project members at Chalmers in cooperation with application designers at Mecel. Preliminary results from this part of the project was published at the *International Workshop on Real-Time Constraints* in October 1999 [8].

The identified constraints and optimization criteria has been modeled using a CP framework based on the SICStus Prolog [9] interpreter from the Swedish Institute of Computer Science. SICStus Prolog was found to be suitable for our purposes as it includes a constraint solver for finite domains [10] which offers some predefined constraints specific for scheduling problems. The translation of the constraints into the code format internal to the solver is fairly straightforward and several examples of how constraints and applications are modeled can be found in a Chalmers technical report [11]. We have recently proposed a scheduling framework for embedded systems using the SICStus Prolog environment, and submitted a manuscript containing the results to the *IEEE Real-Time Systems Symposium* [12]. In this manuscript, we propose a taxonomy of embedded system constraints, strategies for single- and multi-objective optimization, and a performance evaluation of the framework for a set of realistic embedded applications.

We are currently in the process of integrating our proposed scheduling framework into Mecel's existing development tools. In particular, a user interface to the framework is developed in cooperation with Mecel to enable importing and exporting data between SICStus Prolog and Mecel's **Butler** tool [13]. By means of this user interface, it will be possible to get example data from Mecel as well as vizualize schedules made by our scheduling framework. Our main approach for modeling the real-time applications will follow the guidelines of **BASEMENT** [14], a distributed architecture for real-time control applications resulting from a joint work by Mecel AB and Uppsala University.

The second phase of the project will be concerned with developing techniques within the proposed framework that will aid in reducing the complexity associated with finding optimal schedules. The impact of the new techniques will be evaluated using real applications as well as randomly-generated applications. Initially, an evaluation platform developed at Chalmers [15] was planned to be the experimental resource for this part of the project. However, because of the promising results from the first part of the project, our evaluation platform will instead be based on SICStus Prolog. Some initial

[3]By the term "cost effective", we primarily mean resource effective in terms of, *e.g.*, hardware, power consumption or weight. However, we also use this term refer to shorter turn-around times for correcting erroneous design choices.

observations made in conjunction with the development of our scheduling framework [12] indicate that there is a large potential to find algorithm configurations for the SICStus Prolog constraint solver that will contribute significantly to reducing the search complexity. In fact, we expect to find techniques that can reduce scheduling time by orders of magnitude, similar to what has been demonstrated for the B&B algorithm [1, 2, 3, 4].

4.2 Updated milestones and deliverables

The Ph.D. candidate (Cecilia Ekelin) began working in the project in May 1999. Therefore, the milestones and deliverables apply for the period 990501–040501. This encompasses 2×2 years of research and one year of teaching.

Activity 9905–0004: The first phase of the project. Identification of application constraints and optimization criteria used in the design of distributed embedded real-time applications.

Deliverable 0004: State-of-the-art report on optimal algorithms and their application to industrial real-time scheduling. The implementation of a working scheduling framework based on CP.

Activity 0005–0007: Development of interfaces between the framework and Mecel's **Butler** tool.

Deliverable 0007: Demonstration of the scheduling framework and its interaction with **Butler**.

Activity 0008–0109: The second phase of the project. Development and evaluation of complexity-reduction techniques in the CP framework.

Deliverable 0109: Licentiate thesis. Specification of the activities in the remaining time period.

Activity 0109–0309: Further development and evaluation of complexity-reduction techniques in the CP framework. Development of an "engineering handbook" containing guidelines for the practical application of optimal scheduling methods (B&B and CP) to real-time scheduling.

Deliverable 0309: Integration of developed techniques in academic evaluation tools and industrial design tools. Detailed evaluation of the performance of the developed techniques.

Activity 0309–0405: Writing of Ph.D. thesis.

Deliverable 0405: Ph.D. thesis.

5 Industrial relevance and interaction

Optimal scheduling in distributed real-time systems has mainly been reserved for academic purposes. With the introduction of multiple processors in high-volume embedded systems, the scheduling of scarce computing resources can no longer be allowed to be in the hands of an *ad hoc* designer. The techniques developed in the project are expected to be useful for a quite broad range of embedded applications. The general results of the project are expected to be useful, not only for Mecel AB, but also for other industries concerned with the design of embedded real-time systems, such as Ericsson Microwave Systems or Volvo Technical Development.

There are regular meetings with the industrial partner (Mecel AB) where research problems and industrial needs are discussed. The industrial partner also provides case studies that allows for the evaluation of the tools and methods developed at Chalmers. Mecel has also set aside funding for one Master's Thesis project during the year 2000 or 2001.

6 Cooperations and mobility

Similar to some other ARTES projects this project addresses the issues of effective real-time system design. Therefore, we expect the output from this project to be useful in the other related ARTES projects. In particular, we anticipate a close cooperation and synergy effects with the ARTES project "Flexible Reliable Timing Constraints" which is led by Dr. Gerhard Fohler at Mälardalen University. For this reason, we anticipate frequent visit to Dr. Fohler's research group.

Professor Shin's research group at the University of Michigan is one of the few that has made significant work on the B&B algorithm for real-time scheduling purposes. Within this project the collaboration with Professor Shin has so far resulted in one joint research manuscript [4] which reports several effective methods of using the B&B algorithm for distributed real-time systems scheduling. The proposed methods are shown to contribute to a significant (orders of magnitude) reduction in the average search complexity of the algorithm. In the near future (summer or fall 2000) we plan to invite Professor Shin or some of his colleagues to Chalmers for scientific input in the project. We also expect to let the Ph.D. candidate visit Professor Shin's group (most likely after the candidate has achieved the Licentiate degree).

7 Context

This research will be carried out in the High-Performance Computer Architecture Group at Chalmers which is led by Professor Per Stenström and consists of two senior researchers (Per Stenström and Jan Jonsson) with a very strong scientific background. Per Stenström has conducted research in multiprocessor technology since 1984 and his work within memory technology and performance issues in multiprocessor architecture has been internationally recognized. Jan Jonsson has made significant contributions in the field of design and scheduling of parallel and distributed real-time systems.

Mecel AB has a group consisting of around 10 persons working with the design of distributed real-time systems in vehicular applications. The group is led by Mikael Strömberg.

The Real-Time Computing Laboratory at the University of Michigan is led by Professor Kang G. Shin and consists of seven senior researchers with an outstanding scientific background. Two of the researchers at RTCL (Kang G. Shin and Farnam Jahanian) have conducted internationally recognized research in distributed real-time and fault-tolerant computing systems since 1985. In addition, Kang G. Shin has made significant research contributions in the fields of process control, computer architecture, and computer networking.

8 Publications

The following publications have been produced as part of the described project during the first year.

1. C. Ekelin and J. Jonsson, “Solving Embedded System Scheduling Problems using Constraint Programming,” Submitted on May 8, 2000 to the *IEEE Real-Time Systems Symposium*, Orlando, Florida, USA.
2. C. Ekelin and J. Jonsson, “A Modeling Framework for Constraints in Real-Time SYstems,” Technical Report 00–9, Department of Computer Engineering, Chalmers University of Technology, Sweden, May 2000.
3. J. Jonsson and K. G. Shin, “On the Practical Application of the Branch-and-Bound Algorithm to Real-Time Multiprocessor Scheduling,” Submitted on January 27, 2000 to the *Journal of Parallel and Distributed Computing.*
4. C. Ekelin and J. Jonsson, “Real-Time System Constraints: Where do They Come From and Where do They Go?,” In *Proceedings of the 1st Int’l Workshop on Real-Time Constraints*, Alexandria, Virginia, USA, October 16, 1999, pp. 53–57.

The following publications have not been produced as part of the described project but constitute the foundation for our project plan.

1. J. Jonsson, “Effective Complexity Reduction for Optimal Scheduling of Distributed Real-Time Applications,” In *Proceedings of the 19th International Conference on Distributed Computing Systems*, May/June 1999.
2. J. Jonsson, “The Impact of Application and Architecture Properties on Real-Time Multiprocessor Scheduling,” PhD Thesis, School of Electrical and Computer Engineering, Chalmers University of Technology, September 1997.
3. J. Jonsson and K. G. Shin, “A Parametrized Branch-and-Bound Strategy for Scheduling Precedence-Constrained Tasks on a Multiprocessor System,” In *Proceedings of the 26th International Conference on Parallel Processing*, August 1997.

9 Electronic availability

The current status of the project is continuously updated at the following URL:

`http://www.ce.chalmers.se/~case/hpcag/echidna.html`

References

[1] J. Jonsson and K. G. Shin, "A Parametrized Branch-and-Bound Strategy for Scheduling Precedence-Constrained Tasks on a Multiprocessor System," *Proc. of the Int'l Conf. on Parallel Processing*, Bloomingdale, Illinois, Aug. 11–15, 1997, pp. 158–165.

[2] I. Ahmad and Y.-K. Kwok, "Optimal and Near-Optimal Allocation of Precedence-Constrained Tasks to Parallel Processors: Defying the High Complexity Using Effective Search Techniques," *Proc. of the Int'l Conf. on Parallel Processing*, Minneapolis, Minnesota, Aug. 10–14, 1998, pp. 424–431.

[3] J. Jonsson, "Effective Complexity Reduction for Optimal Scheduling of Distributed Real-Time Applications," *Proc. of the IEEE Int'l Conf. on Distributed Computing Systems*, Austin, Texas, May 31 –June 5, 1999, pp. 360–369.

[4] J. Jonsson and K. G. Shin, "On the Practical Application of the Branch-and-Bound Algorithm to Real-Time Multiprocessor Scheduling," Tech. Rep. 99–11, Dept. of Computer Engineering, Chalmers University of Technology, SE-412 96 Göteborg, Sweden, August 1999, Submitted on January 27, 2000 to the *Journal of Parallel and Distributed Computing*.

[5] D.-T. Peng and K. G. Shin, "Static Allocation of Periodic Tasks with Precedence Constraints in Distributed Real-Time Systems," *Proc. of the IEEE Int'l Conf. on Distributed Computing Systems*, New Port Beach, California, June 1989, pp. 190–198.

[6] M. Saksena, "Real-Time System Design: A Temporal Perspective," *Proc. of IEEE Canadian Conference on Electrical and Computer Engineering*, Waterloo, Canada, May 1998, pp. 405–408.

[7] K. Ramamritham, "Where do Time Constraints Come From and Where do They Go?," *International Journal of Database Management*, vol. 7, no. 2, pp. 4–10, 1996.

[8] C. Ekelin and J. Jonsson, "Real-Time System Constraints: Where do They Come From and Where do They Go?," *Proc. of the Int'l Workshop on Real-Time Constraints*, Alexandria, Virginia, Oct. 16, 1999, pp. 53–57.

[9] Intelligent Systems Laboratory, *SICStus Prolog User's Manual*, Swedish Institute of Computer Science, 1995.

[10] M. Carlsson, G. Ottosson, and B. Carlson, "An Open-Ended Finite Domain Constraint Solver," *Proc. of the Int'l Symposium on Programming Languages: Implementations, Logics, and Programs*, H. Glaser et al., Eds., Southampton, UK, Sept. 3–5, 1997, vol. 1292 of *Lecture Notes in Computer Science*, pp. 191–206, Springer Verlag.

[11] C. Ekelin and J. Jonsson, "A Modeling Framework for Constraints in Real-Time Systems," Tech. Rep. 00–9, Dept. of Computer Engineering, Chalmers University of Technology, SE-412 96 Göteborg, Sweden, May 2000.

[12] C. Ekelin and J. Jonsson, "Solving Embedded System Scheduling Problems using Constraint Programming," Tech. Rep. 00–12, Dept. of Computer Engineering, Chalmers University of Technology, SE-412 96 Göteborg, Sweden, May 2000, Submitted on May 8, 2000 to the *IEEE Real-Time Systems Symposium*.

[13] Mecel AB, "Butler," *http://www.mecel.se/*.

[14] H. A. Hansson, H. W. Lawson, M. Strömberg, and S. Larsson, "BASEMENT: A Distributed Real-Time Architecture for Vehicle Applications," *Real-Time Systems*, vol. 11, no. 3, pp. 223–244, Nov. 1996.

[15] J. Jonsson, "GAST: A Flexible and Extensible Tool for Evaluating Multiprocessor Assignment and Scheduling Techniques," *Proc. of the Int'l Conf. on Parallel Processing*, Minneapolis, Minnesota, Aug. 10–14, 1998, pp. 441–450.

A Short CV for Jan Jonsson

Affiliation

Chalmers University of Technology, Department of Computer Engineering, S–412 96 Göteborg, Sweden.
Phone: +46 (31) 772 5220, Fax: +46 (31) 772 3663; Email: janjo@ce.chalmers.se

Permanent positions and degrees

- Assistant professor (*swe.* universitetslektor) of real-time systems, Chalmers, since June 1999.
- Assistant professor (*swe.* forskarassistent) of real-time systems, Chalmers, since May 1998.
- Assistant professor of data communications, Halmstad University (Sweden), 1997.
- Ph.D. degree in computer engineering, Chalmers (Sweden), in 1997.
- M.Sc. degree in computer science and engineering, Chalmers (Sweden), in 1992.

Visiting positions

- Visiting scientist, Real-Time Computing Laboratory, University of Michigan, Ann Arbor (USA), August–December 1996.

Main research interests

- Real-time scheduling strategies for multiprocessor architectures
- Real-time data communications

Selected professional activities

- Member of the program committe for the *7th IEEE Workshop on Parallel and Distributed Real-Time Systems*, San Juan, Puerto Rico, April 1999.