

Real Time Systems'2002

Session 1 : Plates-formes et validation,

Présidente : F. SIMONOT-LION, ENSEM, Vandœuvre

Bossa: a Language-based Approach for the Design of Real Time Schedulers

Luciano Porto Barreto & Gilles Muller

Abstract

Over the years, OS developers and researchers have been continuously implementing new scheduling policies to address the requirements of real-time applications. Such schedulers are especially designed to ensure timing guarantees that traditional schedulers are unable to provide.

However, writing schedulers is a difficult task. It requires understanding the operation of multiple low-level kernel mechanisms. To complicate matters, schedulers are highly optimised components. Such optimisations hinder code maintenance and testing, and make development prone to errors.

In this paper we present the design of Bossa, a language and its run-time system that are specifically targeted toward the development of scheduling policies. The Bossa language provides high-level abstractions that are specific to the domain of scheduling. These abstractions permit the separation of the scheduling policy from the rest of the kernel, easing the development task.

Additionally, a Bossa policy can be verified so as to ensure important safety properties.

We illustrate our approach by presenting the Bossa implementation of an EDF process scheduler and a fixed-priority scheduler that controls other schedulers. We also describe the implementation of the Bossa run-time system, which permits schedulers to be loaded into the Linux kernel.

1.Introduction

2.An analysis of schedule development

2.1. Writing process schedulers

2.2. Writing schedulers that control other schedulers

3.The Bossa language

3.1. Writing a process scheduler

3.2. Writing a virtual scheduler

4.Assessment

5.Implementation

6.Conclusion

Plate-forme d'expérimentation pour l'ordonnancement des applications temps réel à contraintes strictes

L. David, E. Grolleau & S. Constantin

Résumé

Nous étudions à partir d'une plate-forme de test l'apport des techniques d'ordonnancement hors-ligne. Le procédé étudié pour la plate-forme est à fortes contraintes de temps, puisqu'il s'agit de maintenir un pendule en position verticale inversée. Pour cette étude, nous utilisons le noyau temps réel VxWorks(TM).

L'étude est menée de la spécification/conception du système de commande du chariot, jusqu'à la phase d'ordonnancement de l'application à l'aide d'une méthode d'ordonnancement hors-ligne et sa mise en œuvre.

Nous proposons une solution permettant de diminuer la durée du quantum de temps au-delà des possibilités usuelles des systèmes d'exploitation temps réel afin d'ordonner finement le système de tâches et ainsi diminuer les temps de réponse du système.

Mots clés : Système temps réel, ordonnancement en ligne et hors ligne, VxWorks, SA-RT, DARTS, granularité des ordonnanceurs.

1.Introduction

2.Plate-forme de test: le pendule inversé

2.1. Description de la plate-forme

2.2 L'application de contrôle du procédé

3.L'ordonnancement des tâches

3.1. Détermination des paramètres temporels

3.2. Détermination d'ordonnancement

4.Conclusion

ARTISST: an Extensible Framework for the Simulation of Real-Time Systems

David Decotigny & Isabelle Puaut

Abstract

This paper presents ARTISST (ARTISST is a Real-Time System Simulation Tool), an event-driven simulation framework for real-time systems. Contrary to most existing real-time systems simulators, it allows to simulate complex systems made of tasks performing arbitrary computations and exhibiting a complex and realistic pattern for their arrival law, synchronization relations, and execution time. The simulator actually focuses on a time-accurate simulation, for it allows, among other things, to take the operating system (including its scheduler) costs into account. Furthermore, thanks to its modular and extensible architecture, the simulator is not dedicated to a particular Operating System API, but is fully customisable instead.

Keywords: Real-time, Operating System, Scheduler, Discrete events simulator, extensibility.

1.Introduction

1.1. Context

1.2. Motivation for the design of a real-time simulation framework

1.3. Artisst characteristics

1.4. Paper organization

2.Task Model and Task Execution

2.1. Basic Task Model

2.2. Basic Task Execution Status

2.3. Customization and extensibility

2.4. Task execution

3.Overall software architecture

3.1. Artisst messages and modules

3.2. Events and real-time in Artisst

3.3. Global event management engine

3.4. Default modules available

3.5. Extensibility

4.The system simulator module

4.1. Overview

4.2. System simulator core

4.3. Simulated system

4.4. Generated events

4.5. System cost simulation

4.6. Distributed systems simulation

5.Related works

5.1. General-purpose discrete event simulators

5.2. Computing systems simulators

6.Conclusion and perspectives

References

UML et le paradigme synchrone : application à la conception de contrôleurs embarqués

Marie-Agnès Peraldi-Frati, Charles André & Jean-Paul Rigault

Résumé

Nous proposons une approche qui combine les concepts des langages synchrones et d'UML pour la mise en œuvre de systèmes temps réel. Nous prenons un exemple de l'industrie automobile (un contrôleur de siège) pour illustrer notre approche. L'UML n'étant pas capable de prendre en compte certaines spécificités du synchrone, nous introduisons des modèles dynamiques et des stéréotypes mieux adaptés à une programmation réactive. La méthode proposée permet une conception à base d'objets qui prend en compte la sémantique des modèles synchrones et facilite la vérification formelle de propriétés. La réalisation des contrôleurs peut être totalement ou partiellement synchrone.

Mots clefs : UML, Programmation synchrone, Contrôleur embarqué, Application automobile.

1.Introduction

1.1. Domaine d'application

1.2. Problèmes de conception de contrôleurs

1.3. L'approche proposée

1.4. Plan de la présentation

2.UML, le temps réel et l'approche synchrone

2.1. UML et le temps réel

2.2. Le paradigme synchrone

2.3. UML et le paradigme synchrone

3.UML synchrone

3.1. L'approche

3.2. Mise en œuvre

4.Étude de cas : contrôle d'un siège automobile

4.1. Le siège

4.2. La plate-forme

4.3. Les aspects fonctionnels

4.4. Les aspects structurels

4.5. Les aspects comportementaux

4.6. Validation

5. Conclusion

6. Bibliographie

D'un formalisme à l'autre. Entre TIOSM et SDL

Laurent Kaiser & Françoise Simonot-Lion

Résumé

Dans cet article, nous proposons une méthode pour traduire formellement un système décrit dans le formalisme d'une sous-classe des automates temporisés, appelée TIOSM (*Timed Input Output State Machine*) dans un modèle écrit en SDL (*Specification and Description Language*). Les deux formalismes sont présentés ainsi que les règles de traduction. La méthode est illustrée sur un exemple simple. Nous montrons par ailleurs en quoi le modèle est dégradé dans cette traduction en raison d'une représentation du temps différente pour chaque formalisme.

Mots-clés : validation, vérification formelle, automates temporisés, techniques de description formelle

ABSTRACT

In this paper, we propose a method for the formal translation of a system described thanks to the TIOSM formalism (Timed Input Output State Machine), which is a specialization of Timed Automata into a model that respects the SDL syntax (Specification and Description Language).

Both TIOSM and SDL formalism are presented. Then the translation rules are detailed. The translation process is illustrated on a simple example. Furthermore, we show how the obtained SDL model is degraded by the translation because of the different representation of time provided by each formalism.

Keywords : Validation, Formal Verification, Timed Automata, Formal Description Techniques

1. Introduction

2. TIOSM

3. SDL

4. Construction du modèle SDL au niveau système

5. Construction du modèle SDL au niveau bloc

6. Construction du modèle SDL au niveau processus

7. Conclusion et perspectives

Bibliographie

Analyse des approches d'extension de SDL pour la spécification de systèmes et réseaux temps réel

C. Teyssié & Z. Mammeri

Abstract

SDL (*Specification and Description Language*) est une méthode permettant de modéliser de nombreux aspects des systèmes et réseaux temps réel. Mais SDL ne permet pas de spécifier les aspects non fonctionnels d'un système comme le temps ou les ressources système. Dans cet article, notre but est de cibler les points faibles de SDL dans la spécification de systèmes temps réel, de passer en revue plusieurs des approches mises en œuvre pour l'étendre et d'analyser leurs différents avantages et inconvénients.

Mots clés: SDL, Spécification, Temps Réel, Extensions.

1. Introduction

2. Rappel sur les principes de base de SDL

3. Limitation de SDL pour la spécification de systèmes temps réel

4. Adaptations de SDL pour la prise en compte des contraintes de temps

4.1. Méthodes « sans modification de SDL »

4.2. Méthodes « avec modification de SDL »

5. Comparaison des méthodes : SDL*/PMSC et Queueing SDL

6. Conclusion

Bibliographie

Session 2 : Ordonnancement, Présidente : A. Choquet-Geniet, ENSMA, Chasseneuil

Contraintes de précédences et ordonnancement mono-processeur

M. Richard, P. Richard, E. Grolleau & F. Cottet

Résumé

Nous étudions l'ordonnancement mono-processeur de tâches temps réel soumises à des contraintes de précédence. Le modèle mathématique de tâches périodiques est étendu pour prendre en compte un protocole de synchronisation (comme les sémaphores). Lorsque un ordonnanceur à priorités fixes est utilisé, des anomalies d'ordonnancement peuvent survenir. Dans le cas de graphes de tâches réentrant, nous proposons un algorithme de calcul d'une borne supérieure du pire temps de réponse des tâches périodiques.

Mots clés : temps réel, ordonnancement à priorités fixes, mono-processeur, contraintes de précédence.

- 1.Introduction et problématique
- 1.1. Anomalies d'ordonnancement
- 2.Test général d'ordonnançabilité
- 2.1. Les étapes du test
- 2.2. Transformation d'un graphe en chaînes
- 2.3. Forme canonique des priorités et temps de réponse
- 2.4. Regroupement des tâches en segment par rapport à $p_i(j)$
- 2.5. Période d'activité et temps de réponse
- 2.6. Calcul du pire temps de réponse de $t_i(j)$
- 3.Exemple [HKL94]
- 4.Conclusion
- Références

Analysis and Experimentation of Delay-EDD and Jitter-EDD for Guaranteeing Real-Time Communication

D. Bouzid & Z. Mammeri

Abstract

Delay-EDD is a service discipline that can guarantee end-to-end delay for real-time communication. Jitter-EDD is an extension of Delay-EDD in order to guarantee Jitter constraint. The two most important elements that influence the performances are connection admission control (CAC) test and assignment of local delay in each switch .for both disciplines. In this paper we analyze and simulate CAC test and delay assignment functions of Delay-EDD and Jitter-EDD disciplines J'or periodic messages in ATM network.

Keywords: Packet scheduling, Service discipline, Delay-EDD, Jitter-EDD, ÀTM.

- 1.Introduction
- 2.Delay-EDD and Jitter-EDD service disciplines
- 2.1. Delay-EDD service discipline
- 2.2. Jitter-EDD service discipline
- 3.Delay-EDD and Jitter EDD service disciplines for periodic messages
- 4.Analysis of Delay-EDD and Jitter-EDD disciplines for periodic messages
- 4.1. Discussions
- 4.2. Comparison of Liu algorithms with those of Ferrari and Verma
- 5.Simulation od Delay-EDD and Jitter-EDD behaviours
- 6.Conclusion
- Bibliography

Ordonnancement temps réel d'applications comportant des tâches à durée variables

Stéphane Paillet & Annie Choquet-Geniet

Résumé

Nous proposons une méthodologie d'ordonnancement hors ligne d'applications temps réel comportant des tâches à durées variables, les variations provenant de la présence d'instructions conditionnelles dans le corps des tâches. Après avoir adapté le modèle temporel de tâches à ce contexte, nous modélisons ces applications à l'aide de réseaux de Petri autonomes fonctionnant sous la règle de tir maximal et munis d'ensemble terminaux. Nous définissons deux concepts d'ordonnançabilité :

l'ordonnançabilité locale et l'ordonnançabilité globale et nous définissons la notion d'arbre d'ordonnançabilité. Puis, nous montrons comment obtenir les arbres d'ordonnancement à partir des graphes d'ordonnançabilité globale.

Mots clés : Durée variable, instruction conditionnelle, ordonnançabilité locale, ordonnançabilité globale, arbre d'ordonnancement, réseau de Petri, ordonnancement hors ligne.

- 1.Introduction
- 2.Le modèle de tâche étendu
- 2.1. Le modèle temporel
- 2.2. Arbre d'ordonnancement
- 3.Modélisation de l'application
- 3.1. Principe de la modélisation
- 3.2. Règle de fonctionnement et tâche oisive
- 3.3. Prise en compte du modèle étendu
- 4.Analyse hors-ligne
- 4.1. Le graphe des marquages
- 4.2. Extraction d'arborescence
- 5.Conclusion
- 6.Bibliographie

Real-Time scheduling for systems with precedence, periodicity and latency constraints

Liliana Cucu, Rémy Kocif & Yves Sorel

Abstract

First we present the main results concerning, in the one hand systems with periodicity constraints and deadlines, and in the other hand systems with precedence constraints and deadlines, in both cases for one computing resource. Then, we give a model

based on graphs in order to state clearly the problem for scheduling systems with precedence, periodicity and latency constraints. In order to solve this problem we give a non pre-emptive, off-line scheduling algorithm which uses in turn an algorithm of latency marking. We demonstrate the optimality of the scheduling algorithm, and after proving the equivalence of the notions of latency constraint and deadline, we extend this latter algorithm for scheduling real-time systems with precedence, periodicity constraints and deadlines for one computing resource.

Keywords: algorithm, scheduling, optimality, real-time, periodicity, latency, precedence, deadline.

1. Introduction
2. Notations and results
3. The model and the problem to solve
4. Algorithm
 - 4.1. Algorithm of latency marking
 - 4.2. Scheduling algorithm
 - 4.3. Consequences of solving our problem
5. Conclusion and further research

Real-Time Scheduling on Multiprocessors

Joël Goossens, Sanjoy Baruah & Shelby Funk

Abstract

We have recently been studying the scheduling of real-time systems upon uniform and identical multiprocessor platforms. In particular, we have been exploring the use of EDF-scheduling in such systems; this paper summarizes some of our recent findings in this field.

It has previously been shown that EDF is not optimal for multiprocessors in the sense that it is upon uniprocessors. Nevertheless, EDF remains a good algorithm to use in multiprocessor systems - we formally justify this statement, by obtaining conditions which permit us to determine whether a task system can be scheduled to meet all deadlines using EDF upon a given uniform multiprocessor platform, provided we know it to be feasible upon some other uniform multiprocessor platform.

We apply this result to obtain a test for determining whether a periodic task system can be scheduled using EDF upon any given uniform multiprocessor platform. For the problem of scheduling periodic task systems upon identical multiprocessors, we propose a new priority-driven scheduling algorithm that is provably superior to EDF for this purpose.

keywords: Multiprocessors scheduling; periodic tasks; Earliest Deadline First.

1. Introduction
 2. Definitions and assumptions
 - 2.1. Processor model
 - 2.2. Job model
 3. Theoretical foundations
 - 3.1. EDF-scheduling of periodic tasks
 4. Priority-driven algorithms
 - 4.1. EDF-scheduling of periodic task systems
 - 4.2. Priority-driven scheduling of periodic task systems
 5. Conclusion
- References

Session 3 : Communications, Présidente : Pascale Minet, INRIA, Le Chesnay

Ordonnancement FIFO et garantie déterministe de qualité de service en environnement distribué

Laurent George, Steven Martin & Pascale Minet

Résumé

Nous nous intéressons au problème de garantie de qualité de service dans un système distribué implémentant des services différenciés selon l'approche Diffserv. Nous étudions plus particulièrement la classe EF, classe la plus prioritaire. Nous considérons un ensemble de trafics sporadiques de la classe EF suivant une ligne de diffusion dans un arbre de diffusion. Nous souhaitons offrir une garantie déterministe de qualité de service de bout en bout à tout trafic de cette classe. Cette garantie est obtenue en déterminant le temps de réponse de bout en bout de tout trafic de cette classe. Ce temps de réponse doit être inférieur à l'échéance de bout en bout du trafic. Chaque nœud utilise un ordonnancement FIFO au sein de la classe EF, cette classe étant traitée prioritairement à toute autre classe. Le traitement des messages est non préemptif. La solution est basée sur une approche par trajectoire, qui permet d'obtenir un temps de réponse de bout en bout moins pessimiste que d'autres approches existantes. Les résultats théoriques ainsi obtenus sont confirmés par un outil de simulation que nous avons réalisé.

Un contrôle d'admission est alors proposé pour la classe EF en environnement distribué.

Mots-clefs : Qualité de service, Diffserv, classe EF, ordonnancement FIFO, garantie déterministe, contrôle d'admission, temps de réponse de bout en bout, échéance.

1. Introduction
2. Problématique

- 2.1. État de l'art
- 2.2. Notations
- 3. Résultats théoriques
 - 3.1. Résultats connus en monoprocesseur
 - 3.2. Nouveaux résultats en présence d'une seule classe
 - 3.3. Extension au cas de plusieurs classes
- 4. Résultats expérimentaux
 - 4.1. Outil utilisé
 - 4.2. Exemple
 - 4.3. Résultats obtenus
- 5. Contrôle d'admission
- 6. Conclusion
- Références

Flexible Scheduling on Controller Area Network

Paulo Pedreiras & Luís Almeida

Abstract

The Controller Area Network (CAN) protocol was originally developed aiming at automotive applications. However, it rapidly expanded to other real-time application fields, mainly due to its prioritised medium-access control (MAC), high responsiveness, high data efficiency for short data transfers and low cost. Nevertheless, the fact that its MAC relies on fixed priorities imposes a compromise that, in particular situations, is undesired. For example, CAN does not support fairness among streams of messages that require similar quality-of-service and also, it does not allow very high bandwidth utilization levels with guaranteed timeliness for all message streams. In this paper we briefly present the FTT-CAN protocol (Flexible Time-Triggered communication on CAN) and show how this protocol can be used to efficiently support any type of message scheduling over CAN. Particularly, the paper illustrates this flexibility by focusing on the implementation of Earliest Deadline First (EDF) message scheduling on CAN. A comparison between EDF and Rate Monotonic (RM) scheduling in this environment is performed, namely concerning processing and communication overhead and achievable bus utilization. Schedulability analysis for both RM and EDF is presented, as well as a comparison with other proposals for EDF scheduling on CAN.

Keywords: Real-time Communication, Real-time Systems, Communication protocols, Distributed Computer Control Systems, Fieldbus, Scheduling Algorithms.

- 1. Introduction
- 2. FTT-CAN brief presentation
 - 2.1. Synchronous Messaging System
 - 2.2. Asynchronous Messaging System
- 3. RM and EDF scheduling analysis on FTT-CAN
 - 3.1. Rate monotonic scheduling analysis on FTT-CAN
 - 3.2. Earliest Deadline First scheduling on FTT-CAN
- 4. Scheduling and overhead results
 - 4.1. Level of schedulability for RM and EDF on FTT-CAN
 - 4.2. Overhead on the Master node
- 5. Comparing EDF on FTT-CAN with other proposals for EDF on CAN
- 6. Conclusion
- References

Validation temporelle d'applications temps réel distribués à contraintes strictes

Gaëlle Largeteau & Dominique Geniet

Résumé

Ce papier décrit une technique de validation temporelle d'applications temps réel distribuées à contraintes strictes, par le biais de langages rationnels. Dans cette étude, nous tenons compte des impératifs matériels qu'impose la machine cible. En effet, certaines caractéristiques temporelles des tâches sont dépendantes du matériel, ce qui influe sur le langage modèle. Nous faisons l'hypothèse qu'il n'y a pas de migration des tâches sur les différents sites du système et nous n'étudions pas le problème du placement. Nous appliquons enfin cette méthode à un protocole temps réel: CAN.

Abstract :

This paper deals with temporal validation of distributed hard real time systems. We consider here the target machine physical properties. Indeed, several temporal characteristics of some tasks depend on hardware properties, and then influence the model. We suppose that there is no task migration, and we did not study the tasks allocation problem. We valid this method to the protocol CAN.

Mots clés : validation, temps réel, modèle synchrone, langages rationnels, systèmes distribué, réseau.

Key words : validation, real time, synchronous model, rational language, distributed systems, network

Introduction

- 1. Validation de systèmes centralisés
 - 1.1. Modélisation temporelle de tâches
 - 1.2. Modèle temporel d'une application temps réel
 - 1.3. La validation temporelle (G00)
- 2. Validation de systèmes distribués

- 2.1. Prise en compte de la multi-cadence des sites
- 2.2. Intégration de protocoles de communications
3. Étude de cas : validation d'une application basée sur CAN
- 3.1. Le protocole CAN sans gestion d'erreur
- 3.2. Intégration de la synchronisation des drivers dans la modélisation
- 3.3. Gestion d'erreur dans CAN
- 3.4. Le protocole CAN avec gestion des états dégradés

4. Conclusion

Bibliographie

Évaluation de performances d'Ethernet commuté pour des applications temps réel

Anis Koubâa, Ye-Qiong Song

Résumé

Ethernet commuté représente de plus en plus d'intérêt pour des applications temps réel. À côté des propositions industrielles, il manque une évaluation quantitative de ce type de réseau. Dans ce papier, nous proposons une évaluation de performances des réseaux Ethernet commuté en tenant compte de la notion de priorité. Nous adoptons deux approches différentes pour l'évaluation du temps de réponse d'un commutateur Ethernet : l'approche probabiliste (file d'attente M/G/1) pour évaluer la garantie statistique de la qualité de service pour des applications avec contraintes temps réel souples et l'approche déterministe (analyse d'ordonnancement) pour étudier la garantie au pire cas de la qualité de service pour des applications avec contraintes temps réel strictes. Les résultats obtenus fournissent les éléments de base pour le calcul du temps de réponse de bout en bout.

Mots Clés : Ethernet commuté, Qualité de Service, Temps réel, Évaluation de performances

Abstract

Nowadays, switched Ethernet is considered to be of great interest for real time applications. All the special-purpose industrial solutions lack of some general quantitative evaluation of the real time handling capability of switched Ethernet. This paper gives a contribution for the priority-based evaluation of the real time capability of switched

Ethernet technology and its ability to support time constrained applications. We have developed two different approaches to evaluate the Ethernet switch performance, we evaluate the average response time as well as its distribution for Soft Real Time (SRT) applications with a probabilistic approach and the worst-case response time for Hard Real Time (HRT) applications with a deterministic approach. The obtained analytic results could be used for further analysis of the end-to-end delay for larger and more complex networks.

Key words : Switched Ethernet, QoS, Real-time, Performance Evaluation

1. Introduction
2. Modélisation et évaluation de performances
- 2.1. Approche probabiliste pour évaluer la QoS SRT
- 2.2. Approche déterministe pour évaluer la QoS HRT
3. Étude de cas
- 3.1. Description de l'application
- 3.2. Évaluation du pire temps de réponse
- 3.3. Évaluation du temps d'attente moyen
4. Conclusion et perspectives

Bibliographie

Délai de propagation dans les réseaux de capteurs sans fil auto-organisés

Jean-Dominique Decotignie

Résumé

Les réseaux de capteurs sans fil auto-organisés (*self organised wireless sensor networks*) présentent des propriétés intéressantes du point de vue de la fiabilité et de la consommation. L'étude ici présentée s'intéresse au temps de propagation de l'information des capteurs aux consommateurs de cette information. Il apparaît qu'en comparaison avec les réseaux classiques où tous les capteurs sont en visibilité directe des consommateurs, ce délai est dans un certain nombre de cas inférieur.

Mots clés : réseaux sans fil, réseaux de capteurs, contraintes temporelles.

1. Introduction
2. Les réseaux de capteurs sans fil auto-organisés
3. Modèle de calcul
4. Délai en cas de sauts multiples
- 4.1. Principes
- 4.2. Transmission séquentielle
- 4.3. Transmission en parallèle
5. Cas particuliers
6. Discussion
7. Conclusion

Références

Index des Auteurs