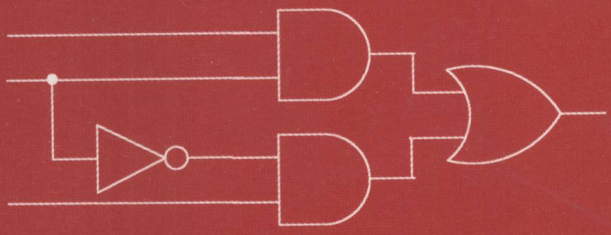
Heinrich Rust

LNCS 3456

Operational Semantics for Timed Systems

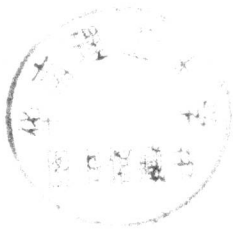
A Non-standard Approach to Uniform Modeling of Timed and Hybrid Systems



Springer

Operational Semantics for Timed Systems

A Non-standard Approach to Uniform Modeling of Timed and Hybrid Systems



Springer

Author

Heinrich Rust
BTU Cottbus, Software-Systemtechnik
Postfach 101344, 03013 Cottbus, Germany
E-mail: heinrich.rust@software-tomography.com

Library of Congress Control Number: 2005923604

CR Subject Classification (1998): D.2, F.1.1, D.3, D.4, F.4

ISSN 0302-9743
ISBN-10 3-540-25576-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-25576-5 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11416166 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3456

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Foreword

This monograph is dedicated to a novel approach for uniform modelling of timed and hybrid systems. Heinrich Rust presents a time model which allows for both the description of discrete time steps and continuous processes with a dense real-number time model. The proposed time model is well suited to express synchronicity of events in a real-number time model as well as strict causality by using uniform discrete time steps. Thus it integrates and reconciles two views of time that are commonly used separately in different application domains. In many discrete systems time is modelled by discrete steps of uniform length, in continuous systems time is seen as a dense flow. The main idea to integrate these different views is a discretization of the dense real-number time structure by using constant infinitesimal time steps within each real-number point in time. The underlying mathematical structure of this time model is based on concepts of Non-standard Analysis as proposed by Abraham Robinson in the 1950s. The discrete modelling, i.e., the description of sequential discrete algorithms at different abstraction levels, is done with Abstract State Machines along the formalisms developed by Yuri Gurevich and temporal logic. These ingredients produce a rich formal basis for describing a large variety of systems with quantitative linear time properties, by seamless integration, refinement and embedding of continuous and discrete models into one uniform semantic framework called "Non-standard Timed Abstract State Machines" (NTASM).

On this theoretically well-founded and elegant basis Heinrich Rust discusses typical problems of time models like "zero time" and "Zeno behaviour," interleaving semantics, time bounds and composition of open systems. The semantic description of two variants of quantitative timed Petri nets, timed automata and hybrid automata with NTASM models shows the generality of the NTASM approach.

This book is an important contribution to the research area of time modelling formalisms. The presentation is well-balanced between theoretical elaboration and a critical discussion of the applicability of the theoretical results by means of appropriate case studies. The new temporal semantics proposed and discussed here can help theoreticians as well as practitioners in gaining better understanding of time models and in building better notations, models and tools for the formal treatment of systems where time matters.

Cottbus, January 2005 Claus Lewerentz

Preface

Time is a fascinating subject. It seems to be quite difficult to come to grips with it. Saint Augustine, in Chapter 14 of Book 11 of the Confessions, said it in this classical way:

> *What, then, is time? If no one asks me, I know what it is. If I wish to explain it to him who asks me, I do not know.*

Making our intuitive understanding of a rich phenomenon explicit, we risk being refuted, by others and by ourselves; and time is an especially rich and irreducible phenomenon. If the subject of time is dealt with in a theological or philosophical context (as Augustine did), this is especially clear, since here time is intimately connected to the concept of existence.

But also in a technical discipline like computer science, time is no simple subject. Here, the question is not what time **is**, but how it should be **modelled** in different situations. Unfortunately, the difference between these questions might seem larger than it turns out to be when we consider specific situations. A model of some phenomenon should abstract from features which are not important in the class of situations considered, while important features should be retained in the model. Thus, dealing with the question of how time should be modelled, we also have to deal with the question of what **are** the important features of time in a class of situations.

A model does not only have to be adequate for the modelled phenomena. If it is to be usable by humans it should also be adequate for their cognitive capabilities. This is sometimes used to justify striving for models that are as simple as possible (while sufficient adequacy with respect to the phenomena is retained). But cognitive simplicity is not an objective trait of a model; with familiarization, a formerly complex model might become simple for somebody working with it. If a model for some phenomenon exists which is very rich in the sense that many other models can be described as special cases of it, then using this model might sometimes be even simpler than using the special cases, and the rich model can serve as an integration platform for ideas which first were used with the more special models. In this way, some unification of concepts might be possible.

This book presents work in which a fairly novel model of quantitative time is tried out, one we hope is both general enough and simple enough to be used as an integration platform for ideas springing from different models of quantitative time. The model of time is discrete, which means that for each

moment there is a well-defined next moment. The model of time is uniform, i.e., the distance between two moments is always the same; and nevertheless it is dense in the real numbers as they are used in classical mathematics, i.e., the resolution induced by the step width is so fine that any real numbered point in (classical) time is approximated by a time point of the model with vanishing error.

After you have read how this model of time is made explicit in this book, you will undoubtedly also see some drawbacks in the approach (several of them are listed in the summary at the end of the book). If you understand this list of drawbacks as a refutation of the approach proposed, then in your eyes I have fallen prey to the problem described above in the citation of Augustine. Let me confess that I myself am not yet completely sure how to interpret the drawbacks. This needs some more investigation.

Credits

A considerable number of people helped during the work which resulted in this book. Claus Lewerentz supported the work from the beginning. My colleagues, especially Dirk Beyer, discussed features of timed systems with me and helped with the presentation, as did in some phases of the work Andreas Prinz and Angelo Gargantini. Egon Börger and Dino Mandrioli gave hints regarding the exposition of ideas and the need to discuss some specific features of the formalism in more depth. The editors at Springer worked hard at correcting my English. And finally, my wife, Korinna Hiersche, made sure that I had time for this work during a parental leave, as did my son Alexander by his arrival.

Cottbus, December 2004 Heinrich Rust

Lecture Notes in Computer Science

For information about Vols. 1–3351

please contact your bookseller or Springer

Vol. 3456: H. Rust, Operational Semantics for Timed Systems. XII, 223 pages. 2005.

Vol. 3455: H. Treharne, S. King, M. Henson, S. Schneider (Eds.), ZB 2005: Formal Specification and Development in Z and B. XV, 493 pages. 2005.

Vol. 3453: L. Zhou, B.C. Ooi, X. Meng (Eds.), Database Systems for Advanced Applications. XXVII, 929 pages. 2005.

Vol. 3452: F. Baader, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XI, 562 pages. 2005. (Subseries LNAI).

Vol. 3450: D. Hutter, M. Ullmann (Eds.), Security in Pervasive Computing. XI, 239 pages. 2005.

Vol. 3449: F. Rothlauf, J. Branke, S. Cagnoni, D.W. Corne, R. Drechsler, Y. Jin, P. Machado, E. Marchiori, J. Romero, G.D. Smith, G. Squillero (Eds.), Applications on Evolutionary Computing. XX, 631 pages. 2005.

Vol. 3448: G.R. Raidl, J. Gottlieb (Eds.), Evolutionary Computation in Combinatorial Optimization. XI, 271 pages. 2005.

Vol. 3447: M. Keijzer, A. Tettamanzi, P. Collet, J.v. Hemert, M. Tomassini (Eds.), Genetic Programming. XIII, 382 pages. 2005.

Vol. 3444: M. Sagiv (Ed.), Programming Languages and Systems. XIII, 439 pages. 2005.

Vol. 3443: R. Bodik (Ed.), Compiler Construction. XI, 305 pages. 2005.

Vol. 3442: M. Cerioli (Ed.), Fundamental Approaches to Software Engineering. XIII, 373 pages. 2005.

Vol. 3441: V. Sassone (Ed.), Foundations of Software Science and Computational Structures. XVIII, 521 pages. 2005.

Vol. 3440: N. Halbwachs, L.D. Zuck (Eds.), Tools and Algorithms for the Construction and Analysis of Systems. XVII, 588 pages. 2005.

Vol. 3439: R.H. Deng, F. Bao, H. Pang, J. Zhou (Eds.), Information Security Practice and Experience. XII, 424 pages. 2005.

Vol. 3436: B. Bouyssounouse, J. Sifakis (Eds.), Embedded Systems Design. XV, 492 pages. 2005.

Vol. 3434: L. Brun, M. Vento (Eds.), Graph-Based Representations in Pattern Recognition. XII, 384 pages. 2005.

Vol. 3433: S. Bhalla (Ed.), Databases in Networked Information Systems. VII, 319 pages. 2005.

Vol. 3432: M. Beigl, P. Lukowicz (Eds.), Systems Aspects in Organic and Pervasive Computing - ARCS 2005. X, 265 pages. 2005.

Vol. 3431: C. Dovrolis (Ed.), Passive and Active Network Measurement. XII, 374 pages. 2005.

Vol. 3427: G. Kotsis, O. Spaniol, Wireless Systems and Mobility in Next Generation Internet. VIII, 249 pages. 2005.

Vol. 3423: J.L. Fiadeiro, P.D. Mosses, F. Orejas (Eds.), Recent Trends in Algebraic Development Techniques. VIII, 271 pages. 2005.

Vol. 3422: R.T. Mittermeir (Ed.), From Computer Literacy to Informatics Fundamentals. X, 203 pages. 2005.

Vol. 3421: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part II. XXXV, 1153 pages. 2005.

Vol. 3420: P. Lorenz, P. Dini (Eds.), Networking - ICN 2005, Part I. XXXV, 933 pages. 2005.

Vol. 3419: B. Faltings, A. Petcu, F. Fages, F. Rossi (Eds.), Constraint Satisfaction and Constraint Logic Programming. X, 217 pages. 2005. (Subseries LNAI).

Vol. 3418: U. Brandes, T. Erlebach (Eds.), Network Analysis. XII, 471 pages. 2005.

Vol. 3416: M. Böhlen, J. Gamper, W. Polasek, M.A. Wimmer (Eds.), E-Government: Towards Electronic Democracy. XIII, 311 pages. 2005. (Subseries LNAI).

Vol. 3415: P. Davidsson, B. Logan, K. Takadama (Eds.), Multi-Agent and Multi-Agent-Based Simulation. X, 265 pages. 2005. (Subseries LNAI).

Vol. 3414: M. Morari, L. Thiele (Eds.), Hybrid Systems: Computation and Control. XII, 684 pages. 2005.

Vol. 3412: X. Franch, D. Port (Eds.), COTS-Based Software Systems. XVI, 312 pages. 2005.

Vol. 3411: S.H. Myaeng, M. Zhou, K.-F. Wong, H.-J. Zhang (Eds.), Information Retrieval Technology. XIII, 337 pages. 2005.

Vol. 3410: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), Evolutionary Multi-Criterion Optimization. XVI, 912 pages. 2005.

Vol. 3409: N. Guelfi, G. Reggio, A. Romanovsky (Eds.), Scientific Engineering of Distributed Java Applications. X, 127 pages. 2005.

Vol. 3408: D.E. Losada, J.M. Fernández-Luna (Eds.), Advances in Information Retrieval. XVII, 572 pages. 2005.

Vol. 3407: Z. Liu, K. Araki (Eds.), Theoretical Aspects of Computing - ICTAC 2004. XIV, 562 pages. 2005.

Vol. 3406: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing. XVII, 829 pages. 2005.

Vol. 3404: V. Diekert, B. Durand (Eds.), STACS 2005. XVI, 706 pages. 2005.

Vol. 3403: B. Ganter, R. Godin (Eds.), Formal Concept Analysis. XI, 419 pages. 2005. (Subseries LNAI).

Vol. 3401: Z. Li, L.G. Vulkov, J. Waśniewski (Eds.), Numerical Analysis and Its Applications. XIII, 630 pages. 2005.

Vol. 3399: Y. Zhang, K. Tanaka, J.X. Yu, S. Wang, M. Li (Eds.), Web Technologies Research and Development - APWeb 2005. XXII, 1082 pages. 2005.

Vol. 3398: D.-K. Baik (Ed.), Systems Modeling and Simulation: Theory and Applications. XIV, 733 pages. 2005. (Subseries LNAI).

Vol. 3397: T.G. Kim (Ed.), Artificial Intelligence and Simulation. XV, 711 pages. 2005. (Subseries LNAI).

Vol. 3396: R.M. van Eijk, M.-P. Huget, F. Dignum (Eds.), Agent Communication. X, 261 pages. 2005. (Subseries LNAI).

Vol. 3395: J. Grabowski, B. Nielsen (Eds.), Formal Approaches to Software Testing. X, 225 pages. 2005.

Vol. 3394: D. Kudenko, D. Kazakov, E. Alonso (Eds.), Adaptive Agents and Multi-Agent Systems III. VIII, 313 pages. 2005. (Subseries LNAI).

Vol. 3393: H.-J. Kreowski, U. Montanari, F. Orejas, G. Rozenberg, G. Taentzer (Eds.), Formal Methods in Software and Systems Modeling. XXVII, 413 pages. 2005.

Vol. 3392: D. Seipel, M. Hanus, U. Geske, O. Bartenstein (Eds.), Applications of Declarative Programming and Knowledge Management. X, 309 pages. 2005. (Subseries LNAI).

Vol. 3391: C. Kim (Ed.), Information Networking. XVII, 936 pages. 2005.

Vol. 3390: R. Choren, A. Garcia, C. Lucena, A. Romanovsky (Eds.), Software Engineering for Multi-Agent Systems III. XII, 291 pages. 2005.

Vol. 3389: P. Van Roy (Ed.), Multiparadigm Programming in Mozart/OZ. XV, 329 pages. 2005.

Vol. 3388: J. Lagergren (Ed.), Comparative Genomics. VII, 133 pages. 2005. (Subseries LNBI).

Vol. 3387: J. Cardoso, A. Sheth (Eds.), Semantic Web Services and Web Process Composition. VIII, 147 pages. 2005.

Vol. 3386: S. Vaudenay (Ed.), Public Key Cryptography - PKC 2005. IX, 436 pages. 2005.

Vol. 3385: R. Cousot (Ed.), Verification, Model Checking, and Abstract Interpretation. XII, 483 pages. 2005.

Vol. 3383: J. Pach (Ed.), Graph Drawing. XII, 536 pages. 2005.

Vol. 3382: J. Odell, P. Giorgini, J.P. Müller (Eds.), Agent-Oriented Software Engineering V. X, 239 pages. 2005.

Vol. 3381: P. Vojtáš, M. Bieliková, B. Charron-Bost, O. Sýkora (Eds.), SOFSEM 2005: Theory and Practice of Computer Science. XV, 448 pages. 2005.

Vol. 3380: C. Priami, Transactions on Computational Systems Biology I. IX, 111 pages. 2005. (Subseries LNBI).

Vol. 3379: M. Hemmje, C. Niederee, T. Risse (Eds.), From Integrated Publication and Information Systems to Information and Knowledge Environments. XXIV, 321 pages. 2005.

Vol. 3378: J. Kilian (Ed.), Theory of Cryptography. XII, 621 pages. 2005.

Vol. 3377: B. Goethals, A. Siebes (Eds.), Knowledge Discovery in Inductive Databases. VII, 190 pages. 2005.

Vol. 3376: A. Menezes (Ed.), Topics in Cryptology – CT-RSA 2005. X, 385 pages. 2005.

Vol. 3375: M.A. Marsan, G. Bianchi, M. Listanti, M. Meo (Eds.), Quality of Service in Multiservice IP Networks. XIII, 656 pages. 2005.

Vol. 3374: D. Weyns, H.V.D. Parunak, F. Michel (Eds.), Environments for Multi-Agent Systems. X, 279 pages. 2005. (Subseries LNAI).

Vol. 3372: C. Bussler, V. Tannen, I. Fundulaki (Eds.), Semantic Web and Databases. X, 227 pages. 2005.

Vol. 3371: M.W. Barley, N. Kasabov (Eds.), Intelligent Agents and Multi-Agent Systems. X, 329 pages. 2005. (Subseries LNAI).

Vol. 3370: A. Konagaya, K. Satou (Eds.), Grid Computing in Life Science. X, 188 pages. 2005. (Subseries LNBI).

Vol. 3369: V.R. Benjamins, P. Casanovas, J. Breuker, A. Gangemi (Eds.), Law and the Semantic Web. XII, 249 pages. 2005. (Subseries LNAI).

Vol. 3368: L. Paletta, J.K. Tsotsos, E. Rome, G.W. Humphreys (Eds.), Attention and Performance in Computational Vision. VIII, 231 pages. 2005.

Vol. 3367: W.S. Ng, B.C. Ooi, A. Ouksel, C. Sartori (Eds.), Databases, Information Systems, and Peer-to-Peer Computing. X, 231 pages. 2005.

Vol. 3366: I. Rahwan, P. Moraitis, C. Reed (Eds.), Argumentation in Multi-Agent Systems. XII, 263 pages. 2005. (Subseries LNAI).

Vol. 3365: G. Mauri, G. Păun, M.J. Pérez-Jiménez, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing. IX, 415 pages. 2005.

Vol. 3363: T. Eiter, L. Libkin (Eds.), Database Theory - ICDT 2005. XI, 413 pages. 2004.

Vol. 3362: G. Barthe, L. Burdy, M. Huisman, J.-L. Lanet, T. Muntean (Eds.), Construction and Analysis of Safe, Secure, and Interoperable Smart Devices. IX, 257 pages. 2005.

Vol. 3361: S. Bengio, H. Bourlard (Eds.), Machine Learning for Multimodal Interaction. XII, 362 pages. 2005.

Vol. 3360: S. Spaccapietra, E. Bertino, S. Jajodia, R. King, D. McLeod, M.E. Orlowska, L. Strous (Eds.), Journal on Data Semantics II. XI, 223 pages. 2005.

Vol. 3359: G. Grieser, Y. Tanaka (Eds.), Intuitive Human Interfaces for Organizing and Accessing Intellectual Assets. XIV, 257 pages. 2005. (Subseries LNAI).

Vol. 3358: J. Cao, L.T. Yang, M. Guo, F. Lau (Eds.), Parallel and Distributed Processing and Applications. XXIV, 1058 pages. 2004.

Vol. 3357: H. Handschuh, M.A. Hasan (Eds.), Selected Areas in Cryptography. XI, 354 pages. 2004.

Vol. 3356: G. Das, V.P. Gulati (Eds.), Intelligent Information Technology. XII, 428 pages. 2004.

Vol. 3355: R. Murray-Smith, R. Shorten (Eds.), Switching and Learning in Feedback Systems. X, 343 pages. 2005.

Vol. 3354: M. Margenstern (Ed.), Machines, Computations, and Universality. VIII, 329 pages. 2005.

Vol. 3353: J. Hromkovič, M. Nagl, B. Westfechtel (Eds.), Graph-Theoretic Concepts in Computer Science. XI, 404 pages. 2004.

Vol. 3352: C. Blundo, S. Cimato (Eds.), Security in Communication Networks. XI, 381 pages. 2005.

Contents

1. Overview

This work introduces a novel approach to modelling timed systems. The main idea consists of a new model of time which is both discrete and dense in the real numbers. This allows to use a discrete base formalism for the description of timed algorithms where system behaviors can be interpreted very straightforwardly in a timed manner without sacrificing that much precision.

Chapter 2 presents the context of our work, which is our understanding of the role of formal methods in the software development process. The main point in that chapter is that "formal methods" does not just mean the use of concepts from mathematics explicitly in software engineering, but the use of such concepts in order to deepen one's understanding of a software engineering problem and its solution, and to express this understanding unambiguously and consistently. From this contextual frame, we derive some consequences for the formalism to be developed.

Part I introduces basic concepts: Our model of time, the discrete base formalism on which we build, and a notation for a temporal logic.

Chapter 3 discusses different models of linear time and introduces a new model which avoids the main problems of classical discrete and continuous models of linear time. The main idea consists in the use of infinitesimals: The flow of time is conceptualized as a sequence of steps of identical infinitesimal length, i.e., we use an infinitesimal discretization of real numbered time.

Chapter 4 presents a short introduction to the number concept we use, which is Nelson's axiomatic approach to infinitesimality.

Chapter 5 presents a variant of abstract state machines (ASMs) as a base formalism for giving the operational semantics of discrete systems. This variant admits two kinds of composition: synchronous and asynchronous. We introduce a semantics for ASMs which is compositional for both kinds of composition, which we call "action semantics". We give reasons for using ASMs as the discrete base formalism.

Chapter 6 describes how we combine ASMs and our model of time in order to describe hybrid systems. Other applications of ASMs in the description of timed and hybrid systems specify the timing independently from the discrete changes – in our approach, the timing is derived from the discrete semantics. An approach using a classical model of time (standard timed ASMs, STASMs) is given first as a comparison; then we use our model of time (non-standard

time ASMs, NTASMs) and show that the infinitesimal discretization might essentially change the semantics of an algorithm. We describe what simulation means for two algorithms given as a STASM and an NTASM, and we introduce a concept which relates the STASM interpretation of an algorithm and the NTASM interpretation of an algorithm: an algorithm is "well-behaved" if and only if each run of its STASM interpretation can be mimicked by a run of its NTASM interpretation.

Chapter 7 introduces a notation for a temporal logic which allows us to specify many properties of NTASM systems succinctly. It uses ideas from the Duration Calculus, an interval-based temporal logic, and transfers them to the infinitesimally discretized time domain.

Part II introduces basic modelling strategies for timed systems – interleaving and synchronous composition, deadlines and openness – and it describes how different magnitudes of hyperreals can be used.

Many modelling formalisms used for describing timed systems support either interleaving or synchronous composition. Our formalism supports both. Chapters 8 and 9 describe how interleaving composition and synchronous composition of timed systems are expressed without formal overheads. We point out how the typical problems of synchronous formalisms, those regarding causality and micro-steps, appear in our framework, and we discuss some specific modelling problems of interleaving and synchronous systems of NTASMs.

The concepts of deadlines, urgency and openness pose special problems in the NTASM framework, which are discussed in Chaps. 10 and 11.

Chapter 12 presents a first application: We model hardware on the gate level with timing-enhanced ASMs. We illustrate illustrated how different magnitudes of the hyperreals can be used to express in the model the fact that some delays are considered to be negligible with respect to others, but if the system is considered using a finer timescale some previously neglected delays can become considerable.

Part III describes some applications of our approach.

Chapter 13 presents an NTASM model of Fischer's real-time based synchronization protocol, and a purely discrete correctness proof made possible by our model of time.

Chapters 14 and 15 present meta-models, i.e., STASM and NTASM semantics of other modelling formalisms, in order to make it plausible that our model can express other formalisms with minimal formal overheads. Chapter 14 investigates two forms of quantitatively timed Petri nets, making explicit their differences in an operational way by supporting both variants in a common formalism. Chapter 15 discusses timed automata. In both chapters, we illustrate how the concept of well-behavedness can be lifted from the base formalism to the expressed formalism.

Chapter 16 presents a larger case study, which is inspired by real-world requirements. We describe the control program for a flexible and timing-

enhanced production cell. It will become clear that the necessary timing properties are very simply expressed in our formalism, and the flexibility with respect to the abstraction level chosen comes in handy when common properties of different variants of the system are described.

Part IV presents a summary of our work.

The appendix collects the definition of some often used notation.