Real-Time Systems, 12, 329–345 (1997)

Report Dagstuhl Seminar on Time Services Schloß Dagstuhl, March 11. – March 15. 1996

DANNY DOLEV dolev@cs.huji.ac.il
Hebrew University, Computer Science Dept., Givat Ram, 91904 Jerusalem, Israel

RÜDIGER REISCHUK reischuk@informatik.mu-luebeck.de
Med. Universität zu Lübeck, Institut für Theoretische Informatik, Wallstrasse 40, D-23560 Lübek, Germany

FRED B. SCHNEIDER fbs@cs.cornell.edu
Cornell University, Dept. of Computer Science, Upson Hall, Ithaca, NY 14853-7510

H. RAYMOND STRONG strong@almaden.ibm.com
IBM Almaden Research, Dept. K53/802, 650 Harry Road, San Jose, CA 95120-6099

Keywords: clock synchronization, network time protocol, time service, fault tolerance

1. Introduction

This seminar brought together a significant fraction of the world's researchers and engineers of network and distributed time services. Until now, there had been little communication between those who implemented time services for computer networks and those exploring new algorithms and analysis techniqes for distributed time services. Our Dagstuhl seminar housed —for the first time— representatives from each of the major approaches to clock synchronization and each of the major providers of real-time synchronization services.

The formal presentations covered new clock synchronization algorithms, methods and actual analysis of these algorithms, and the details of fielded implementations. Participants learned not only the details of the current state of the Internet time protocol (NTP), but also about the history and origins of the algorithms in that implementation.

Some participants could see the practical impacts their fundamental research was having. Others learned about what problems are real and what problems could be safely put aside as academic curiosities. And, implementors learned of new algorithms ripe for trial.

The informal discussions and interactions were at least as important. The meeting forged a community of scientists and engineers with like interests where none had been. Workers in time services now realize the importance of their efforts and have a set of peers with whom to discuss their work. Clock synchronization and time services are a critical part of the infrastructure for any distributed network. By creating a community of workers in this area, the Dagstuhl seminar has itself created an infrastructure to facilitate further advances in the subject.

Schloß Dagstuhl provided an excellent atmosphere for this meeting. We like to thank the staff and the Dagstuhl foundation who made this exchange of research possible.

2. Contents

Introduction 329

Final Seminar Programme 331

Abstracts of Presentation:

David Mills: *Precision Network Time Synchronization* 332

Keith Marzullo: *Synchronizing Clocks and Reading Sensors* 332

John Rushby: *Formal Verification of Clock Synchronization Algorithms* 333

Ulrich Schmid: *Interval-based Clock Synchronization* 334

Klaus Schossmaier: *UTCSU - An ASIC for supporting clock synchronization for distributed Real-Time Systems* 334

Wolfgang Halang: *Time Services Based on Radio Transmitted Official UTC* 335

Christof Fetzer: *Fail-Aware Clock Synchronization* 335

Paulo Verissimo: *Cesium Spray: a Precise and Accurate Time Services on Large-Scaled Distributed Systems* 336

Shlomi Dolev: *Self-Stabilizing Clock Synchronization Algorithms* 336

Wolfgang Reisig: *A Logic for "As Soon As Possible"* 337

Augusto Ciuffoletti: *Self-Stabilization Issues in Clock Synchronization Algorithms* 337

Sergio Rajsbaum:*Extending Causal Order with Real-Time Specifications and Applications to Clock Synchronization* 338

Richard Hofmann: *A Fault-Tolerant Clock Synchronization Method with Correctness Proof and a Posteriori Correction* 338

Timothy Mann: *Marching to Many Distant Drummers* 339

Boaz Patt-Shamir: *Complexity of Clock Synchronization* 339

Ray Strong: *Reading Clocks by Eavesdropping* 340

M. Azadmanesh: *Exploiting Omissive Faults in Synchronous Approximate Agreement* 340

Roger Kieckhafer: *Hybrid Fault Models for Byzantine-safe Systems: Have they reached the Limit?* 341

Rüdiger Reischuk: *Observable Clock Synchronization* 341

Participant List 342

3. Final Programme

Monday, 11.

9.15 - 10.15 David Mills:
10.30 - 11.45 David Mills:
15.30 - 16.30 Keith Marzull
16.45 - 17.45 John Rushby:

Tuesday, 12.

9.00 - 10.00 Ulrich Schmid
10.20 - 11.10 Klaus Schossn *for distributed*
11.15 - 12.05 Wolfgang Hal
15.30 - 16.30 Christof Fetze
16.45 - 17.45 Paulo Verissin *on Large-Scal*

Wednesday, 1

9.00 - 10.00 Shlomi Dolev:
10.30 - 11.15 Wolfgang Reis
11.25 - 12.00 Augusto Ciuff *Algorithms*

Thursday, 14.

9.00 - 10.00 Sergio Rajsba *and Applicatio*
10.20 - 11.20 Richard Hofm *Correctness P*
11.25 - 12.10 Timothy Mann
16.00 - 17.00 Boaz Patt-Sha
17.15 - 17.45 Ray Strong: *R*

Friday, 15. Ma

9.15 - 10.00 M. Azadmanes *Agreement*
10.30 - 11.15 Roger Kieckha *Have they reac*
11.20 - 12.05 Rüdiger Reisch
12.15 end of the conf

.... 329
.... 331
.... 332
ors 332
on Algorithms 333
.... 334
ck synchronization
.... 334
itted Official UTC......... 335
.... 335
ime Services
.... 336
rithms 336
.... 337
hronization
.... 337
pecifications and
.... 338
Method with
.... 338
.... 339
.... 339
.... 340
Approximate
.... 340
stems:
.... 341
.... 341
.... 342

3. Final Programme

	Monday, 11. March 1996
9.15 - 10.15	David Mills: *Precision Network Time Synchronization*
10.30 - 11.45	David Mills: *Precision Network Time Synchronization II*
15.30 - 16.30	Keith Marzullo: *Synchronizing Clocks and Reading Sensors*
16.45 - 17.45	John Rushby: *Formal Verification of Clock Synchronization Algorithms*
	Tuesday, 12. March 1996
9.00 - 10.00	Ulrich Schmid: *Interval-based Clock Synchronization*
10.20 - 11.10	Klaus Schossmaier: *UTCSU - An ASIC for supporting clock synchronization for distributed Real-Time Systems*
11.15 - 12.05	Wolfgang Halang: *Time Services Based on Radio Transmitted Official UTC*
15.30 - 16.30	Christof Fetzer: *Fail-Aware Clock Synchronization*
16.45 - 17.45	Paulo Verissimo: *Cesium Spray: a Precise and Accurate Time Services on Large-Scaled Distributed Systems*
	Wednesday, 13. March 1996
9.00 - 10.00	Shlomi Dolev: *Self-Stabilizing Clock Synchronization Algorithms*
10.30 - 11.15	Wolfgang Reisig: *A Logic for "As Soon As Possible"*
11.25 - 12.00	Augusto Ciuffoletti: *Self-Stabilization Issues in Clock Synchronization Algorithms*
	Thursday, 14. March 1996
9.00 - 10.00	Sergio Rajsbaum: *Extending Causal Order with Real-Time Specifications and Applications to Clock Synchronization*
10.20 - 11.20	Richard Hofmann: *A Fault-Tolerant Clock Synchronization Method with Correctness Proof and a Posteriori Correction*
11.25 - 12.10	Timothy Mann: *Marching to Many Distant Drummers*
16.00 - 17.00	Boaz Patt-Shamir: *Complexity of Clock Synchronization*
17.15 - 17.45	Ray Strong: *Reading Clocks by Eavesdropping*
	Friday, 15. March 1996
9.15 - 10.00	M. Azadmanesh: *Exploiting Omissive Faults in Synchronous Approximate Agreement*
10.30 - 11.15	Roger Kieckhafer: *Hybrid Fault Models for Byzantine-safe Systems: Have they reached the Limit?*
11.20 - 12.05	Rüdiger Reischuk: *Observable Clock Synchronization*
12.15	end of the conference

4. Abstracts of Presentation

Computer Network Time Synchronization

David L. Mills
University of Delaware

This presentation consists of two parts, the first on the architecture, protocol and algorithms of the Network Time Protocol (NTP), which is used to synchronize computer clocks in the Internet, and the second on planned enhancements of NTP to support an autonomous configuration and cryptographic authentication service. The talk covers the architecture model, consisting of the algorithms to select the best sample from a series of samples for each of several servers, then determine the best subset of servers using an intersection algorithm modified from one due to Marzullo. Additional algorithms are used to delete statistical outlyers and improve accuracy by a weighted average of the survivors.

A key to the performance of NTP is the local clock model, which is based on a type-II, adaptive parameter, hybrid phase/frequency-lock loop. The parameters of the model and the phase-frequency mode in which it operates are automatically determined by the prevailing network delays, dispersions and measured local clock stability.

The talk continues to a discussion of enhancements to the NTP model, both in progress and proposed. These include autonomous configuration features which survey the environment and construct a hierarchical spanning tree based on a metric including delay and dispersion, subject to constraints in node degree and total distance. Other enhancements include provision of cryptographic authentication schemes using hybrid public and private key cryptosystems. Factors such as state space required and cryptosystem computation burden are discussed, along with specific proposed systems that combine the best features of both.

Clock Synchronization in a Hard Real-Time System

Keith Marzullo
University of California at San Diego

We study the problem of synchronizing the processor clocks of a set of processors attached by a broadcast-based local area network. We assume that this distributed system is to be used for building a fault-tolerant hard real-time application. So, we choose a failure model that is weak enough for most real-time systems platforms and choose a design goal that the clock synchronization protocol have an extremely small impact on the schedulability of both the processors and the shared network.

We review two obvious candidiates for such a protocol, namely Cristian's probabilistic clock synchronization protocol and Verissimo's *a posteriori* clock synchronization. Both protocols can synchronize quite tightly but for various reasons are not appopriate for our setting. We then present a new protocol, which we call silent clock synchronization, that combines the attractions of both while avoiding their problems. We then present an experimental setup and discuss how we measured the failure rates and the system parameters needed to build our protocol, and discuss the tightness that we attain.

Joint work with Matthew Clegg.

Formal Verificati
Under By

SRI I

Their intricacy and criticali
nized formal verification. I wi
von Henke) of the Interactive
ing this exercise, we found th
lemmas, to be flawed in the c
abled us to detect and correc
provide simpler and more uni

However, the main benefit c
to support exploration of vari
of mechanized deduction allc
much the same way that comp
aerofoil designs. I will descr
one of the assumptions from t
to extend the algorithm to a m
of simple faults to be tolerate

I will also describe similar
Schneider's general treatmen
finement by Paul Miner of N.

References

1. John Rushby and Friedrich vo
actions on Software Engineeri
2. John Rushby. *A formally verif*
teenth ACM Symposium on Pr
1994. Association for Comput
3. Natarajan Shankar. *Mechanic*
synchronization. In J. Vytopil,
571 of Lecture Notes in Con
Springer-Verlag.
4. Paul S. Miner. *Verification of*
NASA Langley Research Cen
5. Paul S. Miner, Shyamsundar
development of a fault-tolera
Systems, pages 128–137, Dan

onization

:hitecture, protocol and algo-
synchronize computer clocks
TP to support an autonomous
talk covers the architecture
from a series of samples for
servers using an intersection
algorithms are used to delete
ige of the survivors.
, which is based on a type-II,
rameters of the model and the
determined by the prevailing
y.
NTP model, both in progress
tures which survey the envi-
a metric including delay and
stance. Other enhancements
ing hybrid public and private
d cryptosystem computation
hat combine the best features

·Time System

go

of a set of processors attached
is distributed system is to be
o, we choose a failure model
nd choose a design goal that
npact on the schedulability of

mely Cristian's probabilistic
clock synchronization. Both
ns are not appopriate for our
ilent clock synchronization,
oblems. We then present an
tes and the system parameters
e attain.

Formal Verification of Clock Synchronization Algorithms Under Byzantine and Hybrid Fault Models

John Rushby
SRI International Menlo Park California

Their intricacy and criticality makes clock synchronization attractive targets for mechanized formal verification. I will describe my verification (undertaken jointly with Friedrich von Henke) of the Interactive Convergence Algorithm of Lamport and Melliar-Smith. During this exercise, we found the proof of the main induction, and four of the five supporting lemmas, to be flawed in the original treatment. Our mechanized verification not only enabled us to detect and correct these flaws, but also to remove the approximations and to provide simpler and more uniform proofs than the originals.

However, the main benefit of the formal specification and verification has been its ability to support exploration of variations on the original algorithm. The calculational character of mechanized deduction allows alternative designs and requirements to be examined in much the same way that computational fluid dynamics supports the exploration of different aerofoil designs. I will describe how, using these capabilities, we were able to eliminate one of the assumptions from the original treatment, to sharpen the statements of others, and to extend the algorithm to a more complex "hybrid" fault model that allows larger numbers of simple faults to be tolerated, without sacrificing Byzantine fault tolerance.

I will also describe similar lessons drawn from my colleague Shankar's verification of Schneider's general treatment of clock synchronization algorithms, and its subsequent refinement by Paul Miner of NASA.

References

1. John Rushby and Friedrich von Henke. *Formal verification of algorithms for critical systems.* IEEE Transactions on Software Engineering, 19(1):13–23, Jan. 1993.
2. John Rushby. *A formally verified algorithm for clock synchronization under a hybrid fault model.* In Thirteenth ACM Symposium on Principles of Distributed Computing, pages 304–313, Los Angeles, CA, August 1994. Association for Computing Machinery.
3. Natarajan Shankar. *Mechanical verification of a generalized protocol for Byzantine fault-tolerant clock synchronization.* In J. Vytopil, editor, Formal Techniques in Real-Time and Fault-Tolerant Systems, volume 571 of Lecture Notes in Computer Science, pages 217–236, Nijmegen, The Netherlands, January 1992. Springer-Verlag.
4. Paul S. Miner. *Verification of fault-tolerant clock synchronization systems.* NASA Technical Paper 3349, NASA Langley Research Center, Hampton, VA, Nov. 1993.
5. Paul S. Miner, Shyamsundar Pullela, and Steven D. Johnson. *Interaction of formal design systems in the development of a fault-tolerant clock synchronization circuit.* In 13th Symposium on Reliable Distributed Systems, pages 128–137, Dana Point, CA, October 1994. IEEE Computer Society.

Interval-based Clock Synchronization

Ulrich Schmid
Technical University of Vienna

We present some results of our research in our SynUTC-project (supported by the Austrian Science Foundation under grant P10244-ÖMA), namely, description and analysis of a novel algorithm suitable for fault-tolerant external clock synchronization. Unlike usual internal synchronization approaches, our interval-based *orthogonal accuracy algorithm* OA provides approximately synchronized clocks maintaining precision and accuracy w.r.t. external time (*universal time coordinated* UTC) simultaneously. This is accomplished by means of a time representation relying on intervals that capture UTC, providing accuracy information encoded in interval lengths. Our analysis, which considers important practical issues like non-zero clock granularity, utilizes a novel, interval-based framework for providing worst-case precision and accuracy bounds subject to a fairly realistic system model.

Keywords: external clock synchronization; fault-tolerant distributed real-time systems; universal time coordinated (UTC); convergence functions; precision analysis; accuracy intervals; clock granularity; Marzullo's function.

UTCSU - An ASIC to support Clock Synchronization for Distributed Real-Time Systems

Klaus Schossmaier
Technical University of Vienna

Targeting a 1 μs precision/accuracy for a time-service in the real-time system area, we need to support the clock synchronization algorithm with adequate hardware. In our project SynUTC we are developing a custom VLSI chip with the following features:

- A rate and state adjustable local clock in the NTP-time format. We propose a new kind of clock, composed of an oscillator and an adder, which adds a particular amount at each tick. This allows us to carry out continuous amortization, to deal with leap-seconds and the use a non-binary input frequency.
- Maintenance of an asymmetrical accuracy interval with proper deterioration dynamics in order to capture UTC. Again, two adder-based clocks are in use with certain supplements.
- Timestamping events for packet transmission/arrival to reduce the clock reading error, for 1PPS from GPS-receivers for injection of external time and for application purpose.
- Generating events with the help of duty-timers to support the clock synchronization protocol for a full message exchange.
- For test and debugging purpose we implemented checksums, signatures, block-sums and snapshots.

The ASIC is manufactured by a $0.7 \mu m$ standard cell CMOS technology and consumes about $100 mm^2$.

Time Services Ba

Having severe deficiencies v
tween sensing and actuation, c
ation are inappropriate to a sur
can be eliminated with availabl
timers handling these alarm jo
real-time systems to achieve pr
occurrences, precisely timed s
dling. No complicated clock
exact — and legal — Univers
GPS, the global navigation an
making use of an application
solution to the timing and clock
feasible.

Fail-A

Univ

Internal clock synchronizatic
any two correct clocks is bound
unbounded message transmissi
nization is impossible to imple
by proposing a new kind of clc
systems with local hardware clc
that new service is derived usin
to transform the specification o
becomes implementable in asy
clock synchronization, we show
leader election problem. The sp
(1) each time server maintains
time servers p and q, if the two
deviation between p's and q's
and (3) a time server which ca
manner and which is timely (i.
synchronization indicator to tru

We show how fail-aware cloc
asynchronous systems. These s
means that a process or messa
process scheduling delays, and
drift rate within some given bou
synchronization is based on the

Time Services Based on Radio Transmitted Official UTC

Wolfgang A. Halang
FernUniversität Hagen

Having severe deficiencies with respect to timing, predictable behavior, and delays between sensing and actuation, contemporary computer systems employed in real-time operation are inappropriate to a surprisingly high extent. It is pointed out how these deficiencies can be eliminated with available technology. The concept of alarm jobs and high-precision timers handling these alarm jobs is presented. They are to serve in embedded, distributed real-time systems to achieve precise time information, accurate time stamping of interrupt occurrences, precisely timed sensing and actuation operations, and comfortable time handling. No complicated clock synchronization by software is needed anymore, since the exact — and legal — Universal Time Co-ordinated (UTC) is received via satellite from GPS, the global navigation and positioning system. A hardware realization is described making use of an application specific integrated circuit. It is shown that the presented solution to the timing and clock synchronization problems is technically and economically feasible.

Fail-Aware Clock Synchronization

Christof Fetzer
University of California at San Diego

Internal clock synchronization requires that at any point in time the deviation between any two correct clocks is bounded by an a priori given constant. Due to network partitions, unbounded message transmission and process scheduling delays, internal clock synchronization is impossible to implement in asynchronous systems. We address this problem by proposing a new kind of clock synchronization that is implementable in asynchronous systems with local hardware clocks: *fail-aware clock synchronization.* The specification of that new service is derived using the general concept of *fail-awareness*: this concept allows to transform the specification of a synchronous service into a fail-aware service such that it becomes implementable in asynchronous systems [1]. To illustrate the usage of fail-aware clock synchronization, we show how it can be used to solve the highly-available, fail-aware leader election problem. The specification of fail-aware clock synchronization requires that (1) each time server maintains a *synchronization indicator*, (2) for any time t and any two time servers p and q, if the two synchronization indicators of p and q are true at t, then the deviation between p's and q's clock at time t is bounded by an a priori given constant δ, and (3) a time server which can communicate with a majority of time servers in a timely manner and which is timely (i.e. it does not suffer performance failures), has to set its synchronization indicator to true.

We show how fail-aware clock synchronization can efficiently be implemented in *timed asynchronous systems*. These systems are characterized by (1) a precise notion of what it means that a process or message to be *timely*, (2) unbounded message transmission and process scheduling delays, and (3) each process has access to a local hardware clock with a drift rate within some given bounds. The proposed protocol to implement fail-aware clock synchronization is based on the protocols described in [2]. Each process p synchronizes its clock in a round based fashion. Only when p succeeds to read the synchronized clocks (i.e. their synchronization indicators are true) of a majority of processes within some given reading error at the end of a round, p can keep its synchronization indicator true for the next round. Otherwise, p's synchronization indicator is (automatically) switched to false [1].

References

1. Fetzer, C. and Cristian, F., "Fail-Awareness in Timed Asynchronous Systems", Proceedings of the 15th ACM Symposium on Principles of Distributed Computing, May 1996, Philadelphia, pp. 314–321a.
2. Cristian, F. and Fetzer, C., "Fault-Tolerant Internal Clock Synchronization", Proceedings of the Thirteenth Symposium on Reliable Distributed Systems, Oct 1994, Dana Point, Ca., pp. 22–31.

CesiumSpray: a Precise and Accurate Global Clock Service for Large-scale Systems

P. Veríssimo
Univ. Lisboa

In large-scale systems, such as Internet-based distributed systems, classical clock-synchronization solutions become impractical or poorly performing, due to the number of nodes and/or the distance. We present a global time service for world-wide systems, based on an innovative clock synchronization scheme, named CesiumSpray. The service exhibits high precision and accuracy; it is virtually indefinitely scalable; and it is fault-tolerant. It is deterministic for real-time machinery in the local area, which makes it particularly well-suited for, though not limited to, large-scale real-time systems.

The clock synchronization scheme is a hybrid of external and internal synchronization. The root of the hierarchy are the GPS satellites, which "spray" their reference time over a set of nodes provided with GPS receivers, one per local network, where the second level of the hierarchy performs internal synchronization, further "spraying" the external time inside the local network. The algorithm of the second level is inspired on the high precision a posteriori agreement synchronization algorithm, modified to follow an external clock, and able to use simple group communication and membership facilities.

Joint work with L. Rodrigues, Tech. Un. Lisboa, and A. Casimiro, Univ. Lisboa.

Self-Stabilizing Synchronization Algorithms

Shlomi Dolev
Ben-Gurion University

Three self-stabilizing clock synchronization algorithms will be presented. The first algorithm is self-stabilizing clock synchronization algorithm that uses bounded clock values. The algorithm guarantees convergence to a state where all clock values are identical, and are subsequently maintained to be identical. It is assumed that during the convergence period no new fault occurs.

The second algorithm is wait
algorithm is more robust than
the convergence period as well
repeatedly stop executing their
The last algorithm consider
severe (and realistic) fault mod
stabilizes to synchronize the sys
that experience Byzantine faul
if it is controlled by an adversa
The talk summarizes joint wo
Welch.

A tempor

Time assumptions can be u
redundant. Nevertheless, the
made explicit, is frequently ea
A temporal logic is suggeste
A local progress operator desc
any interleaved clock synchro
soon as possible" with given r
arguments into parallel contex

Self-Stabiliza

We present an external c
stabilization aspects.
Traditionally, self-stabilizat
clocks, as in the talk given b
external clock synchronizatio
a periodic synchronization of
that the periodic behavior bec
initial state.
The talk outlines the techniq
properties using temporal logi

References

1. Augusto Ciuffoletti, "Using sir
International Conference on Di

s to read the synchronized clocks
y of processes within some given
iization indicator true for the next
iatically) switched to false [1].

ous Systems", Proceedings of the 15th
996, Philadelphia, pp. 314–321a.
iization", Proceedings of the Thirteenth
t, Ca., pp. 22–31.

bal Clock Service for

ited systems, classical clock-
forming, due to the number of
for world-wide systems, based
umSpray. The service exhibits
alable; and it is fault-tolerant.
a, which makes it particularly
ystems.
and internal synchronization.
ay" their reference time over a
ork, where the second level of
aying" the external time inside
pired on the high precision a
follow an external clock, and
cilities.

imiro, Univ. Lisboa.

lgorithms

ll be presented. The first al-
it uses bounded clock values.
k values are identical, and are
iring the convergence period

The second algorithm is wait-free self-stabilizing clock synchronization algorithm. This algorithm is more robust than the first algorithm since it is able to cope with faults during the convergence period as well. The algorithm guarantees convergence even if processors repeatedly stop executing their program and restart.

The last algorithm consider the most sever faults. The algorithm copes with a more severe (and realistic) fault model than the traditional Byzantine fault model. This algorithm stabilizes to synchronize the system clocks in the presence of Byzantine faults — A processor that experience Byzantine fault can exhibit arbitrary "malicious", "two faced", behavior as if it is controlled by an adversary.

The talk summarizes joint works with Anish Arora, Mohamed Gouda, and Jennifer L. Welch.

A temporal logic for "as soon as possible"

Wolfgang Reisig
Humboldt Universität Berlin

Time assumptions can be used to make causal assumptions (implemented by signals) redundant. Nevertheless, the untimed version of an algorithm, with causal dependencies made explicit, is frequently easier to understand and to prove.

A temporal logic is suggested that book-keeps the local resources needed for an action. A local progress operator describes on action's occurrence "as soon as possible", avoiding any interleaved clock synchronization. Its transitive composition describes progress "as soon as possible" with given resources. Powerful composition rules allow to embed local arguments into parallel context.

Self-Stabilization Issues in Clock Synchronization

Augusto Ciuffoletti
Universit' di Pisa

We present an external clock synchronization algorithm that exhibits some self-stabilization aspects.

Traditionally, self-stabilization is bound to internal clock synchronization of logical clocks, as in the talk given by S. Dolev during this seminar. In our case we address an external clock synchronization algorithm, that is based on a diffusion scheme that induces a periodic synchronization of all the nodes. The scheme is self-stabilizing, in the sense that the periodic behavior becomes regular after a bound amount of time and regardless the initial state.

The talk outlines the technique used to carry out the proof, that consists in proving liveness properties using temporal logic, and ordinary algebra to prove real-time related properties.

References

1. Augusto Ciuffoletti, "Using simple diffusion to synchronize the clocks in a distributed system", Proc. 14th International Conference on Distributed Computing Systems, Poznan (Poland), June 1994, pp. 484–491

Extending Lamport's Partial Order with Real-time Specifications and Applications to Optimal Clock Synchronization

Sergio Rajsbaum
Instituto de Matematicas-UNAM, D.F. 04510, Mexico

We consider the problem of clock synchronization in a system with crash faults, with uncertain message delays and bounded clock drifts. To study what is the best possible precision, per execution rather than only in worst cases, we propose a paradigm that extends Lamport's partial order with real-time specifications. Typically, these are specifications related to bounds on message transmission delays and on clock drifts, but can model also other real-time specifications.

We present a characterization theorem for the tightest achievable estimate of the readings of a remote clock in any given execution of the system. Using this theorem, we obtain the first optimal on-line distributed algorithms for external clock synchronization. The general algorithm has unbounded space overhead, which is unavoidable, as we show. For systems with drift free clocks, we present a simple and efficient algorithm.

Joint work with Boaz Patt-Shamir (appeared in Proc. 26th Symp. on Theory of Computing, May 1994, and expanded in Patt-Shamir's Ph.D. thesis at MIT, 1995).

A Fault-tolerant Clock Synchronization Method with Correctness Proof and a Posteriori Correction

Richard Hofmann
University of Erlangen — IMMD VII

A clock synchronization method is presented that augments a universal distributed monitor system with an appropriate facility for time stamping. In order to find out what properties a clock for this purpose must have, the basic problems of monitoring and evaluation of parallel and distributed computer systems are figured out. It turns out that causality of events and time are important properties when analyzing such systems.

There are events that are causally related and other ones that are independent from each other. For the latter ones it is not necessary to know there sequence because it is irrelevant. On the other hand, causally related events must be reflected in the correct sequence. As there is a minimum time between the causing event and the dependent event in a causal relationship, this minimum time interval is the measure for constructing clocks for monitor systems. Currently this time distance lies in the vicinity of half a microsecond.

In order to correctly reflect the sequence of causally related events all clocks used for time stamping in a parallel and distributed system must not deviate more than 1/4 of a microsecond. This can only be achieved by a hardware synchronization method. The method presented in the talk operates on two levels of information that are transferred over the same physical channel, RS 485 serial ports on the master and the slaves, and suitable cabling between them.

The lower level of the synchronization mechanism consists of a PLL scheme that guarantees the same rate for all slave clocks. More precisely, the PLL scheme was optimized for lowest time difference between any two slave clocks with respect to oscillator thermal noise

l-time Specifications and
ıchronization

4510, Mexico

system with crash faults, with
study what is the best possible
propose a paradigm that extends
pically, these are specifications
clock drifts, but can model also

ievable estimate of the readings
ing this theorem, we obtain the
k synchronization. The general
lable, as we show. For systems
orithm.

ymp. on Theory of Computing,
IIT, 1995).

thod with Correctness
ection

VII

a universal distributed monitor
der to find out what properties
monitoring and evaluation of
It turns out that causality of
ch systems.
hat are independent from each
quence because it is irrelevant.
d in the correct sequence. As
e dependent event in a causal
onstructing clocks for monitor
ıalf a microsecond.
ted events all clocks used for
ot deviate more than 1/4 of a
ynchronization method. The
ıation that are transferred over
er and the slaves, and suitable

of a PLL scheme that guaran-
LL scheme was optimized for
ect to oscillator thermal noise

and noise induced by the cabling. Mathematical worst-case analysis has shown error margins of about 5 nanoseconds; our measurements showed an error of less than 2 nanoseconds. Besides providing this precision, the basic synchronization level also contains provisions for fault-tolerance by monitoring the status of the PLL. This status is documented as an attribute to every event.

Based on the services of the basic synchronization level, the upper level guarantees a synchronous start of all clocks, a supervising facility for the synchronization mechanism, and a synchronous stop of all clocks and measurement activities. During a measurement session, so-called *sync-tokens* are issued by the master of the synchronization scheme after regular time intervals (*resynchronization intervals*). Every time such a sync-token is received, an event is generated and stored in the same way as an event from the monitored system.

The regular time intervals allow to check the time stamps of those events generated by the sync-token — these events must have integer multiples of the length of the resynchronization interval. If this is not the case, corrections can be made according to the difference encountered at such events.

The method presented was incorporated into ZM4, a universal distributed hardware monitor system for parallel and distributed systems. Practical experience over a few years has shown, that the most important feature is the ability to prove the correctness of timestamps, eliminating the necessity to take erroneous timestamps into consideration when unexpected behavior is found in the measured event traces.

Marching to Many Distant Drummers

Timothy Mann
DEC Systems Research Center

I haven't worked on time services recently, but I've prepared a talk on some unpublished joint work I did with Leslie Lamport in 1990. My background also includes a time service implementation for SRC's Topaz distributed operating system, based on an earlier algorithm of Lamport's.

In the 1990 work, we address the problem of determining the time in a network where a node may obtain information indirectly from primary time sources via intermediate nodes. Our key idea is to transmit and store each time datum as a pair, consisting of a time interval and a "failure predicate", a boolean expression that indicates precisely which combinations of node failures could invalidate the interval. We describe some techniques based on this idea, but not a complete system design or implementation.

The Complexity of Clock Synchronization

Boaz Patt-Shamir
Northeastern University, Boston

Based on our theory of clock synchronization [PR-94], we prove that any clock synchronization algorithm which is both

- General (i.e., works for all systems), and

- Optimal (i.e, outputs the tightest bounds at any given instant)

is inherently inefficient. Specifically, we define a computational model (a variant of the branching program model), and show that in that model, any optimal synchronization protocol must have unbounded space complexity. The result is proven by considering a set of executions in a very small system (four processors, two of which with drift-free clocks).

Joint work with Sergio Rajsbaum.

Reading Clocks by Eavesdropping

Ray Strong
IBM Almaden Research

The most straightforward way for a client to read a server's clock involves round trip communication between client and server. If there is a requirement that each client read its server's clock at least r times per hour, then the straightforward solution requires at least r message transmissions per hour from each client. We describe a simple protocol that allows one client to read its server's clock by eavesdropping on communication between the server and another client. To be sure that the protocol works on a given network, we do not have to measure accurately such quantities as maximum transmission delay, minimum transmission delay, or tightness of broadcast receipt. We have only to establish that, relative to our method of reading clocks, there is some upper bound on the rate of observable clock drift.

Joint work with Danny Dolev, Rüdiger Reischuk, and Ed Wimmers.

Exploiting Omissive Faults in Approximate Agreement

M.H. Azadmanesh
University of Nebraska at Omaha

The existing voting algorithms, including the Mean of Subsequenced Reduced (MSR) voting algorithms, based on the Thambidurai and Park fault model assume the worst case behavior of malicious faults. Hence, omissive faults are either assumed not to occur, or a predefined value is substituted for the missing values which has the effect of transforming omissive errors into more severe fault modes such as symmetric or asymmetric. As a result, existing voting algorithms can not exploit omissive faults.

This research has further partitioned the symmetric and asymmetric faults into disjoint transmissive and omissive faults. Thus, a new fault model is formed which consists of five modes of failure: benign, transmissive symmetric, omissive symmetric, transmissive asymmetric, and strictly omissive asymmetric. Based on this model, a new family of voting algorithms, called Omission-MSR (OMSR), is introduced. By exploiting omissive faults, OMSR algorithms allow non-faulty processes to locally discard self-evident errors. Furthermore, OMSR voting algorithms are shown to be significantly more fault tolerant than existing voting algorithms.

Hybrid Fau
H

Roger M. K

In recent years considerabl
safe systems. In particular,
been used to show that the fa
Agreement algorithms are co
Byzantine fault model (Byz-
mode Omissive/Transmissive
of Approximate Agreement
algorithms when analyzed ur
This presentation asks wh
gorithms translates into sign
fault-modes have little impa
of new hybrid fault models is
Stochastic Petri-Net (GSPN)
fault-tolerant real-time syste
values show that the new agre
reliability significantly, given
show that incorrect "tuning"
some cases the relationships
intuitive. Hence, extreme car
in real systems.

Obse

We develop a relativistic th
depend on a Newtonian frame
on the problem of estimating
rapport to capture the situatio
purposes.
With a single property, calle
tion flow required for obtainin
concepts with analogs based o
the right quantities.

Joint work with Danny Dolev

stant)

tional model (a variant of the any optimal synchronization is proven by considering a set which with drift-free clocks).

pping

er's clock involves round trip ement that each client read its vard solution requires at least scribe a simple protocol that g on communication between rks on a given network, we do transmission delay, minimum only to establish that, relative n the rate of observable clock

mmers.

iate Agreement

a

bsequenced Reduced (MSR) model assume the worst case er assumed not to occur, or a nas the effect of transforming ic or asymmetric. As a result,

ymmetric faults into disjoint is formed which consists of sive symmetric, transmissive his model, a new family of iced. By exploiting omissive y discard self-evident errors. nificantly more fault tolerant

Hybrid Fault Models for Byzantine-Safe Systems: Have we Reached the Limit ?

Roger M. Kieckhafer, University of Nebraska - Lincoln

In recent years considerable attention has been paid to Hybrid Fault models for Byzantine-safe systems. In particular, Thambidurai and Park's 3-mode fault model (TPH-3) has been used to show that the fault-tolerance of both Byzantine Agreement and Approximate Agreement algorithms are considerably better than predicted by the traditional single-mode Byzantine fault model (Byz-1). We have recently expanded the TPH-3 fault model into a 5-mode Omissive/Transmissive fault model (OTH-5). We have also developed a new family of Approximate Agreement algorithms which are more fault-tolerant than any previous algorithms when analyzed under the OTH-5 fault model.

This presentation asks whether the improved fault-tolerance of the new agreement algorithms translates into significant improvements in Dependability, or whether the new fault-modes have little impact. The motivation is to determine whether the development of new hybrid fault models is worth the effort relative to practical benefits. A Generalized Stochastic Petri-Net (GSPN) Reliability model is presented for a representative distributed fault-tolerant real-time system. GSPN Modeling results over a wide range of parameter values show that the new agreement algorithms and fault model can indeed improve system reliability significantly, given certain values for system parameters. However, results also show that incorrect "tuning" of an agreement algorithm can actually reduce reliability. In some cases the relationships between parameter values and Reliability are quite counter-intuitive. Hence, extreme care must be taken when trying to exploit the OTH-5 fault model in real systems.

Observable Clock Synchronization

Rüdiger Reischuk
Med. Universität zu Lübeck

We develop a relativistic theory of observable clock synchronization that does not use or depend on a Newtonian framework or real time. Within the context of this theory, we focus on the problem of estimating the time on a remote clock. We generalize the concept of *rapport* to capture the situation when such an estimate is sufficient for clock synchronization purposes.

With a single property, called the *Observable Drift Property*, we characterize the information flow required for obtaining rapport. Finally, we compare our relativized and observable concepts with analogs based on the notion of real time in order to show that we are studying the right quantities.

Joint work with Danny Dolev and Ray Strong.

5. Participant List

M.H. Azadmanesh
University of Nebraska at Omaha
Center for Management of Information Technology
308 CBA Building
Omaha NE 68182-0459, USA
email: azad@ahvaz.unomaha.edu
tel: +1-402-554-39 76
url: http://ahvaz.unomaha.edu

Augusto Ciuffoletti
Universit' di Pisa
Dipartimento di Informatica
Corso Italia 40
I-56125 Pisa, Italy
email: augusto@di.unipi.it
tel: +39-50-887-265

Shlomi Dolev
Ben Gurion University
Dept. of Computer Science and Mathematics
Beer Sheva 84105, Israel
email: dolev@CS.bgu.ac.il
url: http://www.cs.bgu.sc.il/~dolev

Christof Fetzer
University of California at San Diego
Department of Computer Science
9500 Gilman Drive
La Jolla CA 92093-0114, USA
email: cfetzer@cs.ucsd.edu
tel: +1-619-597-20 18
url: http://www.cse.ucsd.edu/users/cfetzer

Wolfgang A. Halang
FernUniversität Hagen
FB Elektrotechnik
Informationstechnik / Realzeitsysteme
D-58084 Hagen, Germany
email: wolfgang.halang@fernuni-hagen.de
tel: +49-2331-987 - 372
url: http://www.fernuni-hagen.de/IT

Jaap-Henk Hoepman
Centre for Mathematics and Computer Science
Kruislaan 413
NL-1098 SJ Amsterdam, The Netherlands
email: jhh@cwi.nl
tel: +31-20-592-40 76
url: http://www/cwi.nl/~jhh

Richard Hofmann

Friedrich Alexander Universität
Lehrstuhl Informatik VII
Martensstr. 3
91058 Erlangen, Germany
email: rhofmann@informatik.
tel: +49-9131-85-70 26

Andreas Jakoby
Med. Universität zu Lübeck
Naturwissenschaftliche Fakultät
Institut für Theoretische Informat
Wallstraße 40
D-23560 Lübeck, Germany
email: jakoby@informatik.mu
tel: +49-451/7030-417
url: http://www.itheoi.mu-l

Roger Kieckhafer
University of Nebraska - Lincoln
Dept. of Computer Science
Lincoln NE 68588-0115, USA
email: rogerk@cse.unl.edu
tel: +1-402-472-24 01

Timothy Mann
DEC SRC Systems Research Cen
130 Lytton Ave.
Palo Alto CA 94301-1044, USA
email: mann@src.dec.com
tel: +1-415-853-22 24

Keith Marzullo
University of California at San Di
Department of Computer Science
9500 Gilman Drive
La Jolla CA 92093-0114, USA
email: marzullo@cs.ucsd.edu
tel: +1-619-534-37 29

David L. Mills
University of Delaware
Dept. of Electrical Engineering
Newark DE 19716, USA
email: mills@udel.edu
tel: +1-302-831-8247
url: http://www/eecis.edel.

Boaz Patt-Shamir
Northeastern University
College of Computer Science
161-CN
Boston MA 02115, USA
email: boaz@ccs.neu.edu

Friedrich Alexander Universität
Lehrstuhl Informatik VII
Martensstr. 3
91058 Erlangen, Germany
email: rhofmann@informatik.uni-erlangen.de
tel: +49-9131-85-70 26

Andreas Jakoby
Med. Universität zu Lübeck
Naturwissenschaftliche Fakultät
Institut für Theoretische Informatik
Wallstraße 40
D-23560 Lübeck, Germany
email: jakoby@informatik.mu-luebeck.de
tel: +49-451/7030-417
url: http://www.itheoi.mu-luebeck.de/pages/jakoby

Roger Kieckhafer
University of Nebraska - Lincoln
Dept. of Computer Science
Lincoln NE 68588-0115, USA
email: rogerk@cse.unl.edu
tel: +1-402-472-24 01

Timothy Mann
DEC SRC Systems Research Center
130 Lytton Ave.
Palo Alto CA 94301-1044, USA
email: mann@src.dec.com
tel: +1-415-853-22 24

Keith Marzullo
University of California at San Diego
Department of Computer Science and Engineering 0114
9500 Gilman Drive
La Jolla CA 92093-0114, USA
email: marzullo@cs.ucsd.edu
tel: +1-619-534-37 29

David L. Mills
University of Delaware
Dept. of Electrical Engineering
Newark DE 19716, USA
email: mills@udel.edu
tel: +1-302-831-8247
url: http://www/eecis.edel.edu/~mills

Boaz Patt-Shamir
Northeastern University
College of Computer Science
161-CN
Boston MA 02115, USA
email: boaz@ccs.neu.edu

tel: +1-617-373-20 78

Sergio Rajsbaum
UNAM - Instituto de Matematicas
Ciudad Universitaria
Mexico D.F. 04510, Mexico
email: rajsbaum@servidor.unam.mx
tel: +52-5-622 45 20

Rüdiger Reischuk
Med. Universität zu Lübeck
Naturwissenschaftliche Fakultät
Institut für Theoretische Informatik
Wallstraße 40
D-23560 Lübeck, Germany
email: reischuk@informatik.mu-luebeck.de
tel: +49-451-7030-416
url: http://www.itheoi.mu-luebeck.de/pages/reischuk

Wolfgang Reisig
Humboldt Universität Berlin
Institut für Informatik
Unter den Linden 6
D-10099 Berlin, Germany
email: reisig@informatik.hu-berlin.de
tel: +49-30-20 18 12 20

John Rushby
SRI International Computer Science Laboratory
333 Ravenswood Ave.
Menlo Park CA 94025, USA
email: rushby@csl.sri.com
tel: +1-415-859-54 56
url: http://www/csl/sri/com/fm.html

Ulrich Schmid
Technische Universität Wien
Institut für Automation
Treitlstraße 3
A-1040 Wien, Austria
email: s@auto.tuwien.ac.at
tel: +43-1-58801-8189

Fred B. Schneider
Cornell University
Department of Computer Science
Upson Hall
Ithaca NY 14853-7510, USA
email: fbs@cs.cornell.edu

Klaus Schossmaier
Technische Universität Wien
Institut für Automation
Treitlstraße 3

A-1040 Wien, Austria
email: kmschoss@auto.tuwien
tel: +43-1-58801-8189
url: http://www.auto.tuwien

H. Raymond Strong
IBM Almaden Research
Dept. K53/802
650 Harry Road
San Jose CA 95120-6099, USA
email: strong@almaden.ibm.c
tel: +1-408-927 - 1758

Paulo Verissimo
University of Lisboa
Faculty of Sciences
Campo Grande C5
P-1700 Lisboa, Portugal
email: pjv@di.fc.ul.pt
tel: +351-1-750 01 03
url: http://pandora.inesc.p

Paul Vitanyi
CWI - Mathematisch Centrum
Kruislaan 413
NL-1098 SJ Amsterdam, The Net
email: paulv@cwi.nl
tel: +31-205-924124
url: http://www.cwi.nl/~pau

A-1040 Wien, Austria
email: kmschoss@auto.tuwien.ac.at
tel: +43-1-58801-8189
url: http://www.auto.tuwien.ac.at/~kmschoss

H. Raymond Strong
IBM Almaden Research
Dept. K53/802
650 Harry Road
San Jose CA 95120-6099, USA
email: strong@almaden.ibm.com
tel: +1-408-927 - 1758

Paulo Verissimo
University of Lisboa
Faculty of Sciences
Campo Grande C5
P-1700 Lisboa, Portugal
email: pjv@di.fc.ul.pt
tel: +351-1-750 01 03
url: http://pandora.inesc.pt/

Paul Vitanyi
CWI - Mathematisch Centrum
Kruislaan 413
NL-1098 SJ Amsterdam, The Netherlands
email: paulv@cwi.nl
tel: +31-205-924124
url: http://www.cwi.nl/~paulv

Real-Time Systems, 12, 347–351 (1997)

Contributing Authors

António Casimiro Costa graduated with a degree in Electrical and Computer Engineering in 1991 from the Instituto Superior Técnico (IST) at the Technical University of Lisboa, and obtained his M.Sc. degree from the same University in 1995.

From 1993–1995 he was a teaching assistant at the Department of Digital Systems and Computer Engineering of the IST. Currently he teaches at the Department of Informatics, Faculty of Sciences of the University of Lisboa.

Since 1991 he has been a researcher at INESC, within the Distributed Systems and Industrial Automation Group. He also participated in the international project, Dinas-DQS. His current research interests include SCADA systems and the study of industry-oriented distributed and real-time systems.

Danny Dolev is Professor of Computer Science at the Hebrew University of Jerusalem. He received his Ph.D. from the Weizmann Institute of Science, Israel in 1979. He spent a few years at Stanford University and at IBM carrying out research in various theoretical aspects of Computer Science. His main areas of research since 1982 have been distributed systems and fault tolerance. Professor Dolev is the head of the Transis project in the Hebrew University (Url:http:\\www.cs.huji.ac.il\~dolev). He has published more than 100 research papers.

Martin Horauer received the diploma degree in Electrical Engineering (specialization in industrial electronics and control systems) from the Vienna University of Technology in 1995. Afterwards he joined project SynUTC and is now working towards his Ph.D. at the department of Computer Technology. His interests include computer networking, telecommunication and hardware design, in particular the development of ASICs.

Dietmar Loy received his diploma (1992) and Dr.techn. (1996) degree from the University of Technology (TU) in Vienna. He worked in the industry (Motorola, Austin, Texas) and as a university assistant at the Institute of Computer Technology at TU, Vienna. His research interests are VLSI design with emphasis on Built-In Test and Self-Test structures in communication systems. He is responsible for the ESPRIT projects EUROPRACTICE and EUROFORM at TU, Vienna.

Rüdiger Reischuk earned his Ph.D. in Mathematics at the University of Bielefeld, Germany in 1980. After an assistant professorship in Bielefeld and research at the IBM Research Laboratory, San Jose from 1981–1982 he received the Habilitation in Bielefeld in 1983. From 1985–1994 he was Professor for Theoretical Computer Science at the Technical University Darmstadt; since then he has been a professor at the University Lübeck. His main research interests are complexity theory, efficient algorithms, and parallel and distributed systems. He is the author of the German Monograph, "Einführung in die Komplexitötstheorie."

Reischuk has made several visits to the IBM Watson Research Center Yorktown Heights, International Computer Science Institute Berkeley, and IBM Almaden Research Center San Jose. He is Chairman of the "Theoretical Computer Science Group" of the German Association for Computer Science (1991–1996) and a member of the Editorial Board of the "Electronic Colloquium on Computational Complexity."

Luís Rodrigues graduated
(1991), and a Ph.D. (1996)
puter Engineering from the I
de Lisboa.

He is Assistant Professor
Computer Engineering Dep
where he has taught since
a researcher in the Distribute
Automation Group at INESC
in several national and inter
DELTA-4, Est=EDmulo, BR

His current research intere
and real-time distributed syst
and communication, and rep
in these areas.

Fred B. Schneider has been
nell University's Computer Sc
1978. His research concerr
particularly distributed and
mission-critical applications.
mal methods, as well as protc
tectures for this setting.

Schneider is managing edit
puting, and a member of the ed
Computing Surveys, IEEE Tr
Engineering, High Integrity
Processing Letters, and Annals
text on logic and discrete math
be in print by the year's end.
Professor-at-Large at the Univ

ved the diploma degree in Elec-
:cialization in industrial elec-
:ems) from the Vienna Univer-
1995. Afterwards he joined
now working towards his Ph.D.
mputer Technology. His inter-
:tworking, telecommunication
particular the development of

ed his Ph.D. in Mathematics
feld, Germany in 1980. After
in Bielefeld and research at
ratory, San Jose from 1981–
ilitation in Bielefeld in 1983.
is Professor for Theoretical
Technical University Darm-
been a professor at the Uni-
research interests are com-
algorithms, and parallel and
is the author of the German
in die Komplexitötstheorie."
:h Center Yorktown Heights,
Almaden Research Center
:nce Group" of the German
er of the Editorial Board of

Luís Rodrigues graduated (1986), has a Master's (1991), and a Ph.D. (1996) in Electrical and Computer Engineering from the Instituto Superior Técnico de Lisboa.

He is Assistant Professor of the Electrotechnic and Computer Engineering Department of this Institute, where he has taught since 1989. He has also been a researcher in the Distributed Systems and Industrial Automation Group at INESC since 1986, participating in several national and international projects such as DELTA-4, Est=EDmulo, BROADCAST, and GODC.

His current research interests include fault-tolerant and real-time distributed systems, group membership and communication, and replicated data management. He has more than 20 publications in these areas.

Ulrich Schmid received his diploma (1985) and Dr.techn. degree (1986) from the Technical University of Vienna (computer science and mathematics). Currently, he is Associate Professor at the Department of Automation at TU, Vienna. His current research interests focus on both theoretical and technical computer science topics in the areas of distributed real-time systems, computer communications and operating systems. Dr. Schmid is a member of SIAM, EATCS, and IEEE.

Fred B. Schneider has been on the faculty of Cornell University's Computer Science Department since 1978. His research concerns concurrent systems, particularly distributed and fault-tolerant ones for mission-critical applications. He has worked on formal methods, as well as protocols and system architectures for this setting.

Schneider is managing editor of Distributed Computing, and a member of the editorial boards of: ACM Computing Surveys, IEEE Transactions on Software Engineering, High Integrity Systems, Information Processing Letters, and Annals of Software Engineering. He is co-author of an introductory text on logic and discrete mathematics; his monograph on concurrent programming should be in print by the year's end. Schneider is a Fellow of the ACM and AAAS. He is also a Professor-at-Large at the University of Tromso (Norway).

Klaus Schossmaier received an Engineering degree in Computer Science from the Vienna University of Technology, Austria, in 1991, and a M.S. degree in Computer Science from the University of Massachusetts at Amherst in 1994.

His research interests lie in the areas of real-time computing and distributed systems, with emphasis on clock synchronization and real-time communication.

He is currently a Ph.D. candidate at the Vienna University of Technology, working in project SynUTC at the Department of Automation. Schossmaier is a student member of the ACM and was a recipient of the Fulbright Scholarship in 1992 and 1993.

H. Raymond Strong is Manager of Distributed Computing and Information Systems at the IBM Almaden Research Center in San Jose, California. He received a Ph.D. in Mathematics from the University of Washington in 1967. Before coming to the Computer Science Department at IBM Research San Jose, he was Manager of Theory of Programming at the IBM Watson Research Center in Yorktown Heights. He has worked in the areas of Program Schemata, Access Methods, Machine Architectures, Communication and Synchronization, and Reliable Parallel and Distributed Systems. His current research interests include systems management, time services, fault-tolerance, parallel and distributed algorithms and systems, and rapid prototyping of distributed protocols.

He is the author of several p
on distributed systems, rea
Baltzer Telecommunication
CEC for ESPRIT proposal e

:eived an Engineering degree om the Vienna University of n 1991, and a M.S. degree rom the University of Mas- ı 1994.

; lie in the areas of real-time ed systems, with emphasis on nd real-time communication. . candidate at the Vienna Uni- vorking in project SynUTC at mation. Schossmaier is a stu- M and was a recipient of the ı 1992 and 1993.

Paulo Veríssimo received M.Sc. and Ph.D. degrees in Electrotechnical and Computer Engineering from the IST - Technical University of Lisboa. He is currently a professor in the Department of Informatics, University of Lisboa Faculty of Sciences, where he leads the Navigators group, which has been in several research and technology transfer projects in distributed systems and group-oriented technologies, namely the DELTA-4 and BROADCAST projects.

His current research interests include: real-time and dependability in distributed systems and computer networks; group-oriented architectures and protocols; support for CSCW and distributed computer control. He is the author of several papers in conferences and journals, and co-author of two books, on distributed systems, real-time and fault-tolerance. He is an associate editor for the Baltzer Telecommunications Systems Journal. Veríssimo has also been a consultant to the CEC for ESPRIT proposal evaluation and project reviewing.