DELIVERABLE SUMMARY SHEET

Project Number:	IST-2001-33100
Project Acronym:	PROFUNDIS
Title :	Proofs of Functionality for Mobile Distributed Systems

Deliverable N. :	1
Due Date :	
Delivery Date :	

Partners Owning:	Dipartimento di Informatica Univ. Pisa, Italy
Partners contributed :	Uppsala, FFCT Lisbon, INRIA, Pisa
Made available to:	

1 Summary

This deliverable includes the scientific contributions of the first year for WP1. The goal of this workpackage is to develop a comprehensive automata-like model that supports effective techniques to specify and verify properties of network applications. The deliverable consists of a short presentation of the contributions and of several appendices with the contributed papers. The scientific contributions can be summarized as follows.

Theoretical results have been established about HD automata modeled as coalgebras. Also coalgebraic models (not necessarily finite state) of mobile calculi have been defined, equipped with operations of parallel composition and restriction. Symbolic verification techniques have been developed and applied to security protocols. In particular, the control reachability problem has been studied. The theoretical results have provided the firm foundations needed for the experimental development, and they have driven the design and the prototype implementation. Tool prototypes have been developed for HD automata minimization wrt. bisimilarity. Verification toolkits exploiting symbolic techniques have been designed, implemented and tested. Also, the distributed infrastructure of the Profundis Verification Environment has been designed and a preliminary implementation has been developed.

2 Contributions

We present the contributions according to the tasks they belong to.

Task 1.1: Automata with operations and substitutions

Automata and Coalgebras Pisa has developed a formulation of HD automata as coalgebras on a category of *named sets* and *named functions.* Set elements are equipped with names which are defined up to specific groups of name permutations called *symmetries* [FMP02] (see Appendix 1.1.1). This work exibits two main contributions. First, it provides a coalgebraic definition, which implies the existence of representatives, minimal up to bisimilarity. In previous versions of HD automata, not equipped with symmetries, bisimilarity was characterized in terms of spans of open maps, but no minimal realization was guaranteed to exist. Second, this work formally specifies a declarative procedure to perform effective finite state verification via semantic minimization. Indeed, a toolkit performing state minimization of labelled transition systems for name passing calculi has been implemented [FMRT02]. The software architecture of the toolkit is derived directly from the co-algebraic formulation of the partition-refinement minimization algorithm. The direct correspondence between the semantical structures and the implementation structures facilitates the proof of correctness of the implementation.

Automata, Coalgebras and Operations Instead than coalgebras on a category of named sets, HD automata can be defined as coalgebras on a category of algebras (bialgebras). The construction automatically guarantees that bisimilarity is a congruence wrt. the operations of the algebras. However, this view of HD automata gives up finiteness in all except the most trivial cases. In previous work, an algebra was defined, whose carrier consisted of π-calculus agents equipped with name permutations only. In [BM02], the algebra has been further extended with the operations of parallel composition and restriction. Bisimilarity there corresponds to early observational equivalence, which in fact is a congruence with respect to parallel composition and restriction, but not with respect to prefix. In the same paper a rather general theorem is proved, which gives sufficient conditions for performing the bialgebraic construction in the presence of structural axioms.

Automata, Coalgebras and Spatial Logics Lisbon [Mon02] has developed a notion of automata in which the set of states has been endowed with a structure intended to describe the spatial organization of states in a broad sense. The innovative feature of the model is the treatment of space in coalgebraic terms, the main reason being that for the applications we have in mind we need to observe the structure of given states rather then construct new ones. Indeed, we are looking for general models for spatial logic, where typically we wish to state that if a state has a certain structure, then it satisfies some properties. An additional advantage is that we have a uniform treatment of space and time, since the dynamics of transition systems is naturally described in coalgebraic terms.

Task 1.2: Proof Techniques

Symbolic Verification Techniques INRIA [ALVar] (see Appendix 1.2.1) has developed a *symbolic* reduction system for cryptographic protocols where properties such as secrecy or authenticity are specified by inserting *logical assertions* in the processes. The symbolic reduction system provides a flexible decision procedure for *finite* processes and a reference for sound implementations (the corresponding work on the prototype implementation is commented in the next section). The symbolic reduction system can be regarded as a variant of syntactic unification which is compatible with certain set-membership constraints. For a large class of cryptographic protocols (e.g. the so called ping-pong protocols) a *dag* implementation of the symbolic reduction system leads to an algorithm running in NPTIME, thus matching the lower bound of the problem.

On a related line of work, Pisa [BB02a] (see Appendix 1.2.2) has developed a symbolic operational semantics for the spi-calculus that relies on unification and provides finite and effective models of cryptographic protocols. A method to carry out trace analysis directly on the symbolic model has been also introduced and proved complete (under certain conditions on the cryptographic primitives). A tool named STA (Symbolic Trace Analyzer) [BB02b] has been developed, which implements symbolic execution.

Reachability and Security Protocols INRIA [AM02] (see Appendix 1.2.3) has addressed the issue of decidability of the control reachability problem for various fragments of the asynchronous π-calculus in terms of *name generation*, *name mobility*, and *unbounded control*. It has been proved that the combination of name generation with either name mobility or unbounded control leads to an undecidable fragment. On the other hand, name generation with *unique receiver* and *bounded* input (a condition weaker than bounded control) is decidable by reduction to the coverability problem for Petri Nets. The control reachability problem has been also studied in the context of the Dolev-Yao model of cryptographic protocols [AC02] (see Appendix 1.2.4). A main result is the characterization of a new decidable class of cryptographic protocols with a complexity ranging from simple to double exponential.

Task 1.3. Prototype and case studies

The Profundis WEB The distinguished and innovative feature of the Profundis Verification environment, called *Profundis WEB*, is the idea of viewing the environment as a distributed infrastructure exploited as a *service distributor*. By service we do not mean a monolithic stand-alone verification toolkit, but rather a component available over the WEB that others might use to develop additional services. In the Profundis WEB each verification toolkit has an interface which is network accessible through standard network protocols and which describes the interaction capabilities of the verification toolkit. Hence, verification sessions over the Profundis WEB are developed by combining and integrating together the services available over the WEB. As a consequence, the Profundis WEB is highly portable (it may adapt to a variety of infrastructures) and supports interoperability and dynamic reconfiguration. Moreover, it supports the dynamic integration of several verification techniques. The design of the architecture of the Profundis WEB has been the result of an active collaboration among the research teams in Profundis, with the leading role played by Pisa and Uppsala. To illustrate the effectiveness of the approach a prototype implementation of the Profundis WEB has been developed by Pisa [FGM$^+$02] (see Appendix 1.3.1). This programming experiment has given us the opportunity of testing effectively advantages and weakness of the proposed approach at the very beginning of the project.

Verification Toolkits Several verification toolkits have been implemented in the first year of the project. INRIA [Van02b, Van02a] (see Appendix 1.3.2) has developed a verifier for cryptographic protocols called TRUST. The TRUST toolkit relies on an exact symbolic reduction method, combined with several techniques aiming at reducing the number of interleaving that have to be considered. Authentication and secrecy properties are specified in a very natural way, and whenever an error is found an intruder attacking the protocol is given.

Pisa [FMRT02, RT02] (see Appendix 1.3.3 for [FMRT02]) has developed a toolkit performing state minimization of HD automata. The software architecture of the toolkit is directly suggested by the abstract, declarative, coalgebraic formulation of the partition-refinement minimization algorithm given in [FMP02]. The direct correspondence between the semantical structures and the implementation structures has facilitated the proof of correctness of the implementation. The usefulness of the minimization toolkit has been shown in practice by performing finite state verification of π-calculus specifications.

During previous research efforts, Pisa has also developed a model checker called HAL to verify the satisfiability of properties of π-calculus specifications expressed by a suitable modal logic. The construction of the model checker takes direct advantage of the finite representation of π-calculus in terms of HD-automata. In particular, the construction of the formula expressing properties of specifications is driven by the finite state representation of the system (i.e. the π-calculus process) to be verified. The version of HD-automata considered in this work is without symmetries, and thus it cannot be minimized directly. However under certain conditions, and at the expense of a possibly large increase of the number of states, HD automata can be translated into ordinary automata still preserving bisimilarity, and minimized as such. The paper [FGMP02] (see Appendix 1.3.4), contains refinements and new technical developments of the earlier works which justify the inclusion in the results of Profundis.

Both the minimization toolkit and the model checker have been integrated in the Profundis WEB [FGM$^+$02].

Finally, a tool named STA (Symbolic Trace Analyzer) [BB02b] has been developed in collaboration between Pisa (Profundis) and Firenze (Mikado), which implements symbolic execution of cryptographic protocols. A successful attack is reported in the form of an execution trace that violates the specified property. STA is written in ML. Currently, shared-key, public-key cryptography and hashing are supported.

References

[AC02] R. Amadio and W. Charatonik. On name generation and set-based analysis in Dolev-Yao model (extended abstract). In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

[ALVar] R. Amadio, D. Lugiez, and V. Vanackere. On the symbolic reduction of processes with cryptographic functions. *Theoretical Computer Science*, To appear.

[AM02] R. Amadio and C. Meyssonnier. On decidability of the control reachability problem in the asynchronous pi-calculus. *Nordic Journal of Computing*, 9(2):70–101, 2002.

[BB02a] M. Boreale and M. Buscemi. A framework for the analysis of security protocols. In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

[BB02b] M. Boreale and M. Buscemi. *STA, a Tool for the Analysis of Cryptographic Protocols (Online version).* Dipartimento di Sistemi ed Informatica, Università di Firenze, and Dipartimento di Informatica, Università di Pisa,, http://www.dsi.unifi.it/ boreale/tool.html, 2002.

[BM02] M. Buscemi and U. Montanari. A first order coalgebraic model of pi-calculus early observational equivalence. In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002. Full version in Technical Report TR-02-14, Dipartimento di Informatica, Universitá di Pisa, August 2002.

[FGM+02] G. Ferrari, S. Gnesi, U. Montanari, R. Raggi, G. Trentanni, and E. Tuosto. Verification on the web. Technical Report TR-02-18, Dipartimento di Informatica, Università di Pisa, 2002.

[FGMP02] G. Ferrari, S. Gnesi, U. Montanari, and M. Pistore. A model checking verification environment for mobile processes. *Submitted to ACM TOSEM (under revision)*, 2002.

[FMP02] G. Ferrari, U. Montanari, and M. Pistore. Minimizing transition systems for name-passing calculi: A co-algbraic formulation. In *Proc. FOSSACS'02*, volume 2303 of *LNCS*. Springer Verlag, 2002.

[FMRT02] G. Ferrari, U. Montanari, R. Raggi, and E. Tuosto. From coalgebraic specification to toolkit development. Technical Report TR-02-19, Technical Report, Dipartimento di Informatica Universita' di Pisa, 2002.

[Mon02] L. Monteiro. Transition systems with spatial structures: A coalgebraic framework. Manuscript, 2002.

[RT02] R. Raggi and E. Tuosto. *HD-Reducer (Online version).* Dipartimento di Informatica, Universita' di Pisa, http://jordie.di.unipi.it:8080/mihda, 2002.

[Van02a] V. Vanackère. *The TRUST protocol analyser.* Lab. Informatique de Marseille, http://www.cmi.univ-mrs.fr/ vvanacke/trust.html, 2002.

[Van02b] V. Vanackère. The TRUST protocol analyser, automatic and efficient verification of cryptographic protocols. In *Verification Workshop - Verify02*, 2002.

WP1 Scientific Contributions

APPENDIX

Appendix 1.1.1
G. Ferrari, U. Montanari, and M. Pistore. Minimizing transition systems for name-passing calculi: A co-algbraic formulation. In *Proc. FOSSACS'02*, volume 2303 of *LNCS*. Springer Verlag, 2002.

Appendix 1.1.2
M. Buscemi and U. Montanari. Pi-calculus early observational equivalence: a first order coalgebraic model. Technical Report TR-02-18, Dipartimento di Informatica, Università di Pisa, 2002.

Appendix 1.2.1
R. Amadio, D. Lugiez, and V. Vanackere. On the symbolic reduction of processes with cryptographic functions. *Theoretical Computer Science*, To appear.

Appendix 1.2.2
M. Boreale and M. Buscemi. A framework for the analysis of security protocols. In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

Appendix 1.2.3
R. Amadio and C. Meyssonnier. On decidability of the control reachability problem in the asynchronous pi-calculus. *Nordic Journal of Computing*, 9(2):70–101, 2002.

Appendix 1.2.4
R. Amadio and W. Charatonik. On name generation and set-based analysis in Dolev-Yao model (extended abstract). In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

Appendix 1.3.1
G. Ferrari, S. Gnesi, U. Montanari, R. Raggi, G. Trentanni, and E. Tuosto. Verification on the web. Technical Report TR-02-18, Dipartimento di Informatica, Università di Pisa, 2002.

Appendix 1.3.2
V. Vanackère. The trust protocol analyser, automatic and efficient verification of cryptographic protocols. In *Verification Workshop - Verify02*, 2002.

Appendix 1.3.3
G. Ferrari, U. Montanari, R. Raggi, and E. Tuosto. From coalgebraic specification to toolkit development. Technical Report TR-02-19, Technical Report, Dipartimento di Informatica Universita' di Pisa, 2002.

Appendix 1.3.4
G. Ferrari, S. Gnesi, U. Montanari, and M. Pistore. A model checking verification environment for mobile processes. *Submitted to ACM TOSEM (under revision)*, 2002.