MINISTERO DELL'UNIVERSITÀ E DELLA RICERCA
DIREZIONE GENERALE PER IL COORDINAMENTO E LO SVILUPPO DELLA RICERCA
PROGRAMMI DI RICERCA SCIENTIFICA DI RILEVANTE INTERESSE NAZIONALE
RICHIESTA DI COFINANZIAMENTO (DM n. 1175 del 18 settembre 2007)

PROGETTO DI RICERCA - MODELLO A
Anno 2007 - prot. 2007BHXCFH

1 - Titolo del Progetto di Ricerca

Testo italiano
Controllo e certificazione dell'uso delle risorse (CONCERTO)

Testo inglese
Control and Certification of Resources Usage (CONCERTO)

2 - Durata del Progetto di Ricerca

24 Mesi

3 - Area Scientifico-disciplinare

01: Scienze matematiche e informatiche 100%

4 - Settori scientifico-disciplinari interessati dal Progetto di Ricerca

INF/01 - Informatica
M-FIL/02 - Logica e filosofia della scienza

5 - Coordinatore Scientifico

RONCHI DELLA ROCCA	*SIMONETTA*	
Professore Ordinario	*20/11/1946*	*RNCSNT46S60B111P*
Università degli Studi di TORINO		
Facoltà di SCIENZE MATEMATICHE FISICHE e NATURALI		
Dipartimento di INFORMATICA		
011/6706734 *(Prefisso e telefono)*	*011/751603* *(Numero fax)*	*ronchi@di.unito.it*

6 - Curriculum scientifico

Testo italiano
= POSIZIONI ACCADEMICHE
- Professore ordinario di "Fondamenti dell'Informatica" presso l'Università di Torino, dal 1987.
- Membro del Comitato Scientifico della scuola di Dottorato in "Scienza e alta tecnologia" dell'Università di Torino, indirizzo Informatica.
= APPARTENENZA AD ORGANIZZAZIONI SCIENTIFICHE
- TLCA Steering Committee, dal 2007.
- Accademia delle Scienze di Torino, dal 2001.
- Consiglio Scientifico dell' AILA (Associazione italiana di Logica e sue applicazioni), dal 2000.
- LICS Organizing Committee, dal 1992 al 2003.
- Direzione Nazionale del Condiglio Italiano dell'EATCS, dal 1987 al 1997.
- Comitato scientifico del Centro Interuniversitario degli Studi per la Pace (Torino), dal 2000.
= APPARTENENZA A COMITATI DI PROGRAMMA E INVITI A CONFERENZE (dal 1992)
- LICS 92, Logic Colloquium 94 (invitata), Logic Colloquium 95, TLCA 99, FLOC 99 (conference-co-chair), TLCA 01, ICTCS 01 (co-chair), AILA 05, MFPS 06, Logic Model and Computer Science 06, LSFA 07(invitata), TLCA 07 (chair).
= ATTIVITA' EDITORIALE
- membro dell' Editorial Board del giornale "ACM Transactions on Computational Logic" (TOCL).
Attualmente sta curando una edizione speciale di TOCL sulla complessità Computazionale Implicita, insieme a Patrick Baillot and Jean Yves Marion.
- Editor della serie "Programs and Proofs" di Polimetrica Publisher.
= ATTIVITA' DIDATTICA
- "Fondamenti dell'informatica" nel corso di laurea triennale in Informatica presso l'Università di Torino.
- "Logiche della Programmazione e Teorie dei Tipi" nel corso di laurea magistrale in metodologie e Sistemi Informatici presso l'Università di Torino.

- "Logica della Computazione" nel corso di dottorato in Informatica presso l'Università di Torino.
- Dal 2000 al 2002, direttore del master in"Comunicazione Scientifica", organizzato congiuntamente dall'Università di Torino, Politecnico di Torino e Università del Piemonte Orientale.
= STUDENTI DI DOTTORATO
- Marco Gaboardi, attualmente studente in cotutela con l'Università di Nancy (correlatore Jean Yves Marion)
- Luca Fossati, attualmente studente in cotutela con PPS-Paris, correlatore Pierre Louis Curien
- Mauro Piccolo, attualmente studente in cotutela con PPS-Paris, correlatore Pierre Louis Curien
- Alberto Pravato (amministratore di PROSA e professore a contratto all' Università di Venezia) (ha preso il titolo nel 1997)
- Luigi Liquori (ricercatore INRIA, Sophia Antipolis), correlatore Mariangiola Dezani (ha preso il titolo nel 1996)
- Luca Roversi (professore associato, Università di Torino) (ha preso il titolo nel 1995)
= FORMAZIONE POST-DOTTORATO
- Elaine Pimentel (professore all' Università di Belo Horizonte, Brasile) (ha passato un anno all'Università di Torino nel 2004)
- Luca Paolini (ha avuto un assegno nel periodo 2002-2004, ora ricercatore)
- Olivier Bastonero (ha avuto un assegno nel 1997)
= PROGETTI DI RICERCA
- Coordinatore nazionale del progetto MIUR "Logical Foundations of Abstract Programming Languages" (FOLLIA) (2004 - 2006)
- Coordinatore nazionale del progetto MIUR "From Proofs to Computation Through Linear Logic" (PROTOCOLLO)(2002 - 2004)
- Coordinatore locale del progetto MIUR "Linear Logic and Behind" (coordinatore nazionale Andrea Asperti)(2000-2002)
- Coordinatore locale del progetto europeo TMR "Linear Logic" (coordinatore generale Laurent Regnier, Marseille)(1996-2000)
Coordinatore locale del progetto europeo"Typed Lambda Calculus" (coordinatore generale Jean Yves Girard) (1987-1995)
= INTERESSI DI RICERCA
- Semantica formale dei linguaggi di programmazione
- Logica della programmazione e teorie dei tipi
- Tipi intersezione
- Complessità Computazionale Implicita
= Il suo 60-esimo compleanno è stato celebrato con il colloquio "Types and Computations", associato all' ICTCS'07.

Testo inglese
= ACADEMIC POSITIONS
- Full professor of "Foundations of Computer Science" at the University of Torino, Department of Computer Science, since 1987.
- Member of the Scientific Committee of the PhD School "Science and Hight Technology" of the University of Torino, specialization in Computer Science.
= MEMBERSHIP OF RESEARCH ORGANIZATIONS
- TLCA Steering Committee, since 2007.
- Academy of Science of Turin, since 2001.
- Association Board of AILA (Italian Association of Logic and Applications), since 2000.
- LICS Organizing Committee, from 1992 until 2003.
- National direction of the Italian Council of EATCS, from 1987 until 1997.
- Scientific Committee of the Interuniversity Center for Peace Studies (Torino), since 2000.
= PC-MEMBERSHIP AND INVITATIONS TO CONFERENCES (from 1992)
- LICS 92, Logic Colloquium 94 (invited), Logic Colloquium 95, TLCA 99, FLOC 99 (conference-co-chair), TLCA 01, ICTCS 01 (co-chair), AILA 05, MFPS 06, Logic Model and Computer Science 06, LSFA 07(invited), TLCA 07 (chair).
= EDITING
- member of the Editorial Board of the journal "ACM Transactions on Computational Logic" (TOCL).
Actually she is editor of a special issue of TOCL on Implicit Computational Complexity, together with Patrick Baillot and Jean Yves Marion.
- Editor of the Series "Programs and Proofs" of the Polimetrica Publisher.
= TEACHING
- "Fondations of Computer Science" in the undergraduate course in Computer Science at the University of Torino.
- "Program Logics and Type Theories" in the graduate course in Computer Science at the University of Torino.
- "Computational Logic" in the PhD course in Computer Science at the University of Torino.
- From 2000 until 2002, director of the master in "Scientific Communications", jointly organized by the University of Torino, the Politecnico of Torino and the University of Piemonte Orientale.
= PhD STUDENTS
- Marco Gaboardi, actual student both of the University of Torino and the University of Nancy (co-advisor Jean Yves Marion)
- Luca Fossati, actual student both of the University of Torino and PPS-Paris, co-advisor Pierre Louis Curien
- Mauro Piccolo, actual student both of the University of Torino and PPS-Paris, co-advisor Pierre Louis Curien
Alberto Pravato (administrator of the factory PROSA and lecturer at the University of Venice) (he gave his title in 1997)
- Luigi Liquori (INRIA researcher, Sophia Antipolis), co-advisor Mariangiola Dezani (he gave his title in1996)
- Luca Roversi (associated professor, University of Turin) (he gave his title in 1995)
= POSTDOC FORMATIONS
- Elaine Pimentel (professor at the University of Belo Horizonte, Brazil) (she had had a grant for one year at the University of Torino in 2004)
- Luca Paolini (he had had a grant for two years at the University of Torino in 2002-2004, now he is researcher)
- Olivier Bastonero (he had had a grant for one year at the University of Torino in 1997)
= RESEARCH PROJECTS
- National coordinator of the MIUR project "Logical Foundations of Abstract Programming Languages" (FOLLIA) (2004 - 2006)
- National coordinator of the MIUR project "From Proofs to Computation Through Linear Logic" (PROTOCOLLO)(2002 - 2004)
- Local coordinator of the MIUR project "Linear Logic and Behind" (national coordinator Andrea Asperti)(2000-2002)
- Local coordinator of the TMR project "Linear Logic" (general coordinator Laurent Regnier, Marseille)(1996-2000)
Local coordinator of the european project "Typed lambda Calculus" (general coordinator Jean Yves Girard) (1987-1995)
= RESEARCH INTERESTS
- Formal Semantics of Programming Languages
- Program Logics and Type Theories
- Intersection Types
- Implicit Computational Complexity
= Her 60-th birthday has been celebrated with the colloquium "Types and Computations", co-located with ICTCS'07.

7 - Pubblicazioni scientifiche più significative del Coordinatore Scientifico

1. *GABOARDI M, MARION J.Y, RONCHI DELLA ROCCA S. (2007). A logical account of PSPACE. POPL 08. January 2008. to appear.*
2. *GABOARDI M, RONCHI DELLA ROCCA S. (2007). A Soft Type Assignment System for lambda-Calculus. LECTURE NOTES IN COMPUTER SCIENCE. vol. 4646, pp. 253-267 ISSN: 0302-9743. CSL 07.*
3. *LIQUORI L, RONCHI DELLA ROCCA S. (2007). Intersection Types a la Church. INFORMATION AND COMPUTATION. vol. 205, pp. 1371-1386 ISSN: 0890-5401.*
4. *RONCHI DELLA ROCCA S. (2007). Typed Lambda Calculi and Applications, 8th International Conference, TLCA 2007, Paris, France, June 26-28, 2007, Proceedings. (editor).*
5. *L. PAOLINI, E. PIMENTEL, RONCHI DELLA ROCCA S. (2006). An Operational Characterization of Strong Normalization. LECTURE NOTES IN COMPUTER SCIENCE. vol. 3921, pp. 367-381 ISSN: 0302-9743. FOSSACS'06.*
6. *COPPOLA P., DAL LAGO U., RONCHI DELLA ROCCA S. (2005). Elementary affine logic and the call by value lambda calculus. LECTURE*

NOTES IN COMPUTER SCIENCE. vol. 3461, pp. 131-145 ISSN: 0302-9743. TLCA'05.

7. *COPPOLA P., RONCHI DELLA ROCCA S. (2005). Principal Typing for Lambda Calculus in Elementary Affine Logic. FUNDAMENTA INFORMATICAE. vol. 65, pp. 87-112 ISSN: 0169-2968.*
8. *LIQUORI L., RONCHI DELLA ROCCA S. (2005). Towards an intersection typed system a la Church. ELECTRONIC NOTES IN THEORETICAL COMPUTER SCIENCE. vol. 136 C, pp. 43-56 ISSN: 1571-0661.*
9. *PAOLINI L., PIMENTEL E., RONCHI DELLA ROCCA S. (2005). Lazy strong normalization. ELECTRONIC NOTES IN THEORETICAL COMPUTER SCIENCE. vol. 136 C, pp. 103-116 ISSN: 1571-0661.*
10. *PAOLINI L., RONCHI DELLA ROCCA S. (2004). Lazy logical semantics. ELECTRONIC NOTES IN THEORETICAL COMPUTER SCIENCE. vol. 104, pp. 235-251 ISSN: 1571-0661.*
11. *PAOLINI L., RONCHI DELLA ROCCA S. (2004). Parametric Parameter Passing Lambda Calculus. INFORMATION AND COMPUTATION. vol. 186, pp. 87-106 ISSN: 0890-5401.*
12. *RONCHI DELLA ROCCA S., PAOLINI L. (2004). The Parametric lambda -Calculus: a Metamodel for Computation. Texts in Theoretical Computer Science: An EATCS Series. Springer-Verlag, Berlin,.*
13. *COPPOLA P, RONCHI DELLA ROCCA S. (2003). Principal Typing in Elementary Affine Logic. LECTURE NOTES IN COMPUTER SCIENCE. vol. 2701, pp. 90-104 ISSN: 0302-9743. TLCA 03.*
14. *DEZANI-CIANCAGLINI M., RONCHI DELLA ROCCA S. (2003). Intersection Types. MATHEMATICAL STRUCTURES IN COMPUTER SCIENCE. vol. 13 (1) ISSN: 0960-1295. (editors).*
15. *RONCHI DELLA ROCCA S. (2002). Intersection Typed Lambda-Calculus. ELECTRONIC NOTES IN THEORETICAL COMPUTER SCIENCE. vol. 70.1 ISSN: 1571-0661.*
16. *RESTIVO A., RONCHI DELLA ROCCA S., ROVERSI L. (2001). ICTCS'01 (proceedings). LECTURE NOTES IN COMPUTER SCIENCE. vol. 2201 ISSN: 0302-9743. (editors).*
17. *RONCHI DELLA ROCCA S., ROVERSI L. (2001). Intersection Logic. LECTURE NOTES IN COMPUTER SCIENCE. vol. 2142, pp. 414-429 ISSN: 0302-9743.*
18. *KFOURY A., RONCHI DELLA ROCCA S., TIURYN J., URZYCZYN P. (1999). Alpha-conversion and Typability. INFORMATION AND COMPUTATION. vol. 150, pp. 1 -- 21 ISSN: 0890-5401.*
19. *PAOLINI L., RONCHI DELLA ROCCA S. (1999). Call-by-value Solvability. THEORETICAL INFORMATICS AND APPLICATIONS. vol. 33, pp. 507 -- 534.*
20. *PRAVATO A., RONCHI DELLA ROCCA S., ROVERSI L. (1999). The call-by-value lambda calculus: a semantic investigation. MATHEMATICAL STRUCTURES IN COMPUTER SCIENCE. vol. 9(5), pp. 617 -- 650 ISSN: 0960-1295.*
21. *O. BASTONERO, A. PRAVATO, RONCHI DELLA ROCCA S. (1998). Structures for lazy semantics. In: DAVID GRIES , WILLEM P. DE ROEVER. Programming Concepts and Methods. (vol. 125, pp. 30-48). ISBN: 0-412-83760-9.*
22. *BAKEL S., LIQUORI L., RONCHI DELLA ROCCA S., URZYCZYN P. (1997). Comparing Cubes of Typed and Type Assignment systems. ANNALS OF PURE AND APPLIED LOGIC. vol. 86, pp. 267-303 ISSN: 0168-0072.*
23. *RONCHI DELLA ROCCA S., ROVERSI L. (1997). Lambda Calculus and Intuitionistic Linear Logic. STUDIA LOGICA. vol. 57 ISSN: 0039-3215.*
24. *PRAVATO A., RONCHI DELLA ROCCA S., ROVERSI L. (1995). Categorical semantics of the call-by-value lambda calculus. LECTURE NOTES IN COMPUTER SCIENCE. vol. 902, pp. 381-396 ISSN: 0302-9743.*
25. *GIANNINI P., RONCHI DELLA ROCCA S. (1994). A Type Inference Algorithm for a complete stratification of the Polymorphic Type Discipline. INFORMATION AND COMPUTATION. vol. 110, pp. 115-173 ISSN: 0890-5401.*
26. *F. HONSELL, RONCHI DELLA ROCCA S. (1992). An approximation theorem for topological lambda models and the topological incompleteness of the lambda calculus. JOURNAL OF COMPUTER AND SYSTEM SCIENCES. vol. 45, pp. 49-75 ISSN: 0022-0000.*
27. *L. EGIDI, F. HONSELL, RONCHI DELLA ROCCA S. (1992). Operational, denotational and logical descriptions: a case study. FUNDAMENTA INFORMATICAE. vol. 16(2), pp. 149-169 ISSN: 0169-2968.*
28. *GIANNINI P., RONCHI DELLA ROCCA S. (1988). Characterization of typings in polymorphic type discipline. In: YURI GUREVICH. LICS 88. (pp. 61-70). ISBN: 0-8186-0853-6. : IEEE.*
29. *RONCHI DELLA ROCCA S. (1988). Principal Type scheme and unification for intersection type discipline. THEORETICAL COMPUTER SCIENCE. vol. 59, pp. 1-29 ISSN: 0304-3975.*
30. *RONCHI DELLA ROCCA S., B. VENNERI. (1984). Principal Type Scheme for an extended type theory. THEORETICAL COMPUTER SCIENCE. vol. 28, pp. 151-169 ISSN: 0304-3975.*

8 - Elenco delle Unità operative

Unità	Responsabile dell'Unità di Ricerca	Qualifica	Ente	Impegno
I	*RONCHI DELLA ROCCA Simonetta*	*Professore Ordinario*	*Università degli Studi di TORINO*	*80*
II	*MARTINI Simone*	*Professore Ordinario*	*Università degli Studi di BOLOGNA*	*141*
III	*TORTORA DE FALCO Lorenzo*	*Professore Associato non confermato*	*Università degli Studi ROMA TRE*	*67*
IV	*PEDICINI Marco*	*Ricercatore*	*Consiglio Nazionale delle Ricerche*	*83*

9 - Abstract del Progetto di Ricerca

Testo italiano

In questo progetto noi intendiamo gettare le fondamenta teoriche per la definizione di strumenti di analisi (essenzialmente statici) finalizzati a garantire proprietà operazionali di programmi legate in particolare all'uso delle risorse, in modelli computazionali sia sequenziali che concorrenti. Una gestione statica dell'uso delle risorse è di importanza crescente nell'ambito della computazione mobile e distribuita. L'area di ricerca a cui ci riferiamo ha come scopo quello di associare ad un programma una certificazione, che ne garantisca particolari proprietà computazionali, in particolare la quantità di risorse (tempo e spazio, ma non solo) necessarie per la sua esecuzione. Per queste finalità ci baseremo su strumenti e tecniche derivati dalla logica, più specificamente teoria della dimostrazione, e sulla semantica, guardando in particolare alla logica lineare e al lambda calcolo. La ricerca che noi vogliamo portare avanti è essenzialmente fondazionale, e può essere idealmente suddivisa in due obiettivi principali. Noi vogliamo produrre:

1. Tecniche fondazionali per l'analisi e la verifica di proprietà operazionali di programmi;
2. Teorie computazionali che modellino l'interazione corretta con l'ambiente.

Per quanto riguarda il punto (1), saremo soprattutto interessati a studiare sistemi formali che generino programmi con complessità limitata (tempo di esecuzione e spazio usato), in contrapposizione a sistemi similari che caratterizzano solamente funzioni di complessità limitata. Partendo dalle cosiddette logiche "leggere", svilupperemo sistemi di assegnazione di tipo con queste caratteristiche, in cui si possa esprimere una vasta classe di programmi. Vogliamo inoltre sperimentare tali

tecniche incorporando un controllo delle risorse usate in implementazioni parallele di riduzioni ottimali per il lambda calcolo (PELCR). Tale obiettivo richiede però anche una analisi raffinata della semantica della computazione sequenziale. Riguardo a questo, approfondiremo lo studio di calcoli paradigmatici stabili derivati da PCF, e costruiremo modelli semantici di PCF e di calcoli a questo correlati.

Riguardo al punto (2), ci sono modelli computazionali che ancora necessitano di una semantica formale per studiare la loro interazione con le risorse (pensiamo in special modo, a non solo, a modelli emergenti, come quello quantistico e biologico). In quest'area non possiamo promettere implementazioni o realizzazione di tecniche specifiche, pensiamo invece di produrre teorie semantiche unificanti che possano essere di aiuto per futuri sviluppi formali. Intendiamo in particolare studiare la semantica dei sistemi di Complessità Computazionale Implicita, allo scopo di unificare sistemi diversi e avere una base comune per effettuare confronti o estensioni. Useremo strumenti semantici derivati dalla logica lineare, come la nozione di esperimento, la semantica a contesti e la geometria delle interazioni. Per sviluppare nuove tecniche logiche abbiamo anche bisogno di uno studio fondazionale. Ci proponiamo di studiare la connessione tra la Logica Lineare, in particolare le reti di prova, e i modelli di computazioni concorrenti e distribuite. La correttezza per le reti di prova di MALL (Multiplicative Additive Linear Logic) che sono stabili rispetto ai passi globali di eliminazione del taglio sono un passo importante in questa direzione. Quindi ci proponiamo di applicare questi strumenti fondazionali ai modelli computazionali quantistico e biologico. Più specificatamente, vogliamo disegnare un lambda calcolo quantistico, per studiare classi di complessità in questo nuovo contesto, tramite sistemi di tipi. Per quanto riguarda il modello biologico di computazione, vogliamo esplorare la possibilità di rappresentare reti biologiche usando una semantica stocastica delle reti di prova differenziali (un altro strumento derivato dalla logica lineare).

In conclusione, ci proponiamo di sviluppare metodi formali e di attaccare problemi fondazionali al confine tra logica (reti di prova e semantica), linguaggi di programmazione, e complessità computazionale, allo scopo di porre basi solide per il disegno lo sviluppo di linguaggi con certificazione di uso limitato di risorse.

Testo inglese
The project focuses on formal foundations for language-based (especially static) techniques guaranteeing resource-related runtime properties of programs, both in the sequential and in the concurrent computational model. Such static resource management is of increasing importance in the domain of embedded, mobile, and distributed computing. The project belongs to the research area whose aim is to associate to a program a certification assuring some computational properties, and in particular the quantification of resources (time, space, etc.) necessary to its execution.

We will derive the tools and techniques for our investigation from the field of logical proof-theory and semantics, with special interest on linear logic and lambda-calculus.

We will work on two main objectives:

1. Foundational techniques for the analysis and verification of runtime properties of programs;
2. Computational theories modeling the correct interaction with the environment.

As for (1), we will be mainly interested in studying formal systems expressing programs of limited complexity (running time or used space), as opposed to similar systems which merely characterize functions of limited complexity. Starting from the so-called "light" logics, we will develop type systems with these characteristics, expressing a wide range of programs. We plan to experiment with certain parallel implementation techniques for optimal lambda reduction (PELCR), incorporating resource control. The objective, however, calls also for a new, deep understanding of the semantics of sequential computation. Under this respect we will study stability for a paradigmatic calculus (PCF), and we plan to study the construction of semantical models for PCF and related calculi.

As for (2), existing computational models, and especially new, emerging ones (e.g., quantum or biological computation), still need formal semantic models of their interaction with resources. Here we cannot promise implementations or tools. We plan instead to have unifying, semantical theories helping in future formal development. We plan, in particular, to study the semantics of Implicit Computational Complexity, with the aim of unifying different systems and have common ground for their comparison or extension. We will use semantical tools derived from linear logic, such as the notion of experiment, context semantics, and geometry of interaction. To develop new logical tools we also need a foundational investigation. We plan to investigate the connections between Linear Logic, in particular proof-nets, and concurrent and parallel computation model. Correctness for MALL proof-nets which are stable under fully local cut reduction steps are an important step in this direction. Finally, we plan to apply these foundational tools to the quantum and biological computational models. We will design a quantum lambda-calculus, a type system for it, and we will study the relations of these with complexity classes. We will explore the possibility of representing biological networks by a stochastic version of the differential net semantics (another linear logic tool).

In summary, we plan to develop formal tools, and to attack foundational problems at the border of logic (both proof-theory and semantics), language design, and computational complexity, with the aim to provide sound grounds for the design and development of resource-bounded certified programs.

10 - Parole chiave

nº	Parola chiave (in italiano)	Parola chiave (in inglese)
1.	*LOGICA LINEARE*	*LINEAR LOGIC*
2.	*COMPLESSITA' COMPUTAZIONALE IMPLICITA*	*IMPLICIT COMPUTATIONAL COMPLEXITY*
3.	*LAMBDA CALCOLO*	*LAMBDA CALCULUS*
4.	*SEMANTICA DENOTAZIONALE*	*DENOTATIONAL SEMANTICS*
5.	*INTERAZIONE*	*INTERACTION*

11 - Obiettivi finali che il Progetto si propone di raggiungere

Testo italiano
La sempre maggior diffusione di sistemi di elaborazione distribuiti ha reso ancora più attuali le problematiche legate alla correttezza e alla certificazione di proprietà di programmi. In questo contesto, diventa essenziale garantire e certificare proprietà run-time per reti costituite da piccole componenti mobili con risorse computazionali limitate, che ricevono dalla rete stessa i programmi da eseguire. In particolare è importante garantire che l'esecuzione di tali programmi da una parte non richieda risorse in eccesso rispetto a quelle a disposizione del componente che li deve eseguire, dall'altra non costituisca una minaccia per la sicurezza globale del sistema. In questo progetto noi intendiamo gettare le fondamenta teoriche per la definizione di strumenti di analisi (essenzialmente statica) finalizzati a garantire proprietà operazionali di programmi legate in particolare all'uso delle risorse, in modelli computazionali sia sequenziali che concorrenti. Una gestione statica dell'uso delle risorse è di importanza crescente nell'ambito della computazione mobile e distribuita.

L'area di ricerca a cui ci riferiamo ha come scopo quello di associare ad un programma una certificazione, che ne garantisca particolari proprietà computazionali, in particolare la quantità di risorse (tempo e spazio, ma non solo) necessari per la sua esecuzione. In questo modo è possibile controllare, nella maggior parte dei casi staticamente, se il componente che si vuole usare possiede la quantità di risorse necessarie all'esecuzione di quel programma. Quindi noi vogliamo definire strumenti metodologici per studiare la complessità classica, da un nuovo punto di vista. E' importante notare che in questo contesto noi usiamo la parola complessità per denotare sia la misura delle risorse di tempo e spazio necessarie per la computazione sia l'analisi di tutti i vincoli, sia qualitativi che quantitativi, generati dalla necessità che il programma interagisca in modo corretto con l'ambiente in cui deve essere eseguito. L'interazione corretta con l'ambiente include anche problematiche di sicurezza, comunicazione e sincronizzazione. Il nostro sogno è una teoria unificante delle risorse computazionali, intendendo la parola risorse nel suo significato più vasto.

Gli strumenti teorici che vogliamo usare per la realizzazione del progetto sono basati essenzialmente sulla logica. L'uso della logica, in particolare della teoria della dimostrazione, come strumento per studiare la computazione nasce dall'isomorfismo di Curry-Howard, che mette in luce il profondo legame tra programmi e dimostrazioni logiche, ed è il punto di partenza per gran parte della ricerca sulla programmazione funzionale. Negli ultimi 20 anni, a partire dalla nascita della Logica Lineare, la teoria della dimostrazione ha contribuito allo sviluppo di un approccio nuovo, più raffinato, allo studio sia della teoria della ricorsione che della teoria della complessità. La nozione innovativa di computazione nata dalla Logica Lineare e dalle Logiche Leggere, derivate da questa, è essenzialmente basata sulla nozione di interazione. Da un punto di vista computazionale, la logica può essere vista come una mutua interazione di dimostrazioni, e la normalizzazione (o più precisamente l'eliminazione del taglio) è lo strumento che realizza l'interazione. Da questa prospettiva possiamo disegnare linguaggi in cui gli agenti computazionali sono completamente descritti dalle loro mutue interazioni, rendendo quindi la nozione di computazione indipendente da uno specifico modello computazionale. Di conseguenza diventa possibile formulare analisi facilmente adattabili ai nuovi paradigmi computazionali sorti in questi ultimi anni, come quello biologico e quantistico. La ricerca che noi vogliamo portare avanti è essenzialmente fondazionale, e può essere idealmente suddivisa in due obiettivi principali. Noi vogliamo produrre:

1. Tecniche fondazionali per l'analisi e a verifica di proprietà operazionali di programmi;
2. Teorie computazionali che modellino l'interazione corretta con l'ambiente.

Noi vogliamo anche fare una parte di lavoro sperimentale: in particolare l'unità IV vuole sperimentare l'uso delle nuove tecniche per produrre una nuova versione della macchina PELCR [Parallel Environment for optimal Lambda Calculus Reduction], che esegue in modo parallelo la riduzione dei lambda termini secondo la strategia di riduzione ottimale. Questo sarà un utile controllo sperimentale dei nostri risultati teorici.

Contribuiscono a questo progetto ricercatori che hanno già una lunga esperienza di lavoro in comune, insieme ad alcuni nuovi ricercatori con collaudata esperienza nello studio delle proprietà dei linguaggi di programmazione. L'esperienza di lavoro comune è maturata all'interno di due precedenti programmi di ricerca nazionali PROTOCOLLO (from PROof TO COmputations through Linear LOgic, http://protocollo.di.unito.it) e FOLLIA (FOndazioni Logiche di LInguaggi Astratti di Programmazione, http://follia.di.unito.it). PROTOCOLLO era focalizzato essenzialmente su problemi tecnici legati alla logica lineare, rilevanti per l'informatica teorica. FOLLIA era una continuazione del primo, e ha esteso lo spettro della ricerca dallo studio della Logica Lineare allo studio delle Logiche Leggere e della Ludica. Inoltre, in FOLLIA abbiamo iniziato lo studio di proprietà di programmi nell'ambito delle computazioni mobili con risorse limitate. Alcuni ricercatori che partecipano a questo nuovo progetto fanno anche parte della "Rete Italo-francese sulla logica e la geometria della computazione" (2006-2008), in collaborazione con la rete di ricerca francese GEOCAL (Geometrie de la Computation http://iml.univ-mrs.fr/geocal06/), ora terminata. Ora molti di noi collaborano con il progetto ANR "CHOCO" (Curry-HOward pour la COncurrence) diretto da Thomas Ehrhard (http://www.pps.jussieu.fr/ ehrhard/CHOCO/tout-definitif.pdf), e alcuni con il progetto NoCost (New tools for Complexity, Semantics and Types) diretto da Patrick Baillot (http://www-lipn.univ-paris13.fr/nocost/). Questa sinergia tra progetti diversi costituisce un'importante risorsa per il progetto attuale.

I nuovi ricercatori che si sono associati al nostro gruppo portano una preziosa esperienza nell'area della teoria della ricorsione e nel disegno di modelli denotazionali di linguaggi di programmazione, e quindi noi ci aspettiamo che possano portare idee innovative per lo studio di nuovi modelli computazionali e per il trasferimento di proprietà dalla logica allo studio dei linguaggi di programmazione. Il numero totale di ricercatori afferenti a questo progetto è rilevante: siamo 28, di cui 10 sono studenti di dottorato. Infatti il progetto raccoglie tutti i ricercatori che in Italia studiano la logica lineare finalizzata all'informatica, e noi pensiamo che sia molto produttivo, per i nostri scopi, raccogliere insieme tutta la nostra esperienza. Per noi, mantenere la coesione di questo gruppo di ricerca e continuare la nostra collaborazione è un obiettivo molto importante. Un altro obiettivo, altrettanto importante, è la formazione di nuovi nuovi ricercatori: infatti noi intendiamo usare una percentuale rilevante del contributo finanziario richiesto per assegni di ricerca (chiediamo 4 assegni, uno per ogni unità).

Testo inglese
The widespread diffusion of distributed computer systems has increased the need to solve the issues of program correctness and certification. In particular, it is of central importance to guarantee and certify runtime properties for computer networks constituted of small and mobile devices with bounded computational resources that receive programs to be executed from the network itself. In fact in this setting it is important to assure that such programs both do not exceed the availability of resources of the devices themselves, and do not pose a threat to the security of the system. In this project we plan to provide formal foundations for language-based (especially static) techniques guaranteeing resource-related runtime properties of programs, both in the sequential and concurrent computational models. Such static resource management is of increasing importance in the domain of embedded, mobile, and distributed computing.

This project belongs to the research area whose aim is to associate to a program a certification assuring some computational properties, and in particular the quantification of resources (time and space, but not only) necessary to its execution. In this way, we can control (in many cases statically) if the device that we want to use possesses the quantity of resources necessary for its execution. Thus, we want to arrange methodological tools for studying, from a new viewpoint, the classical complexity. In this setting we use the word complexity for denoting both the measure of time and space necessary for the computation, and the analysis of all constraints, qualitative and quantitative, originated from the need that the program interacts in a correct way with the environment in which it must be executed (here correct takes in account safety, communication and synchronization as instances). Our dream is a unifying theory for the computational resources control, where the word resource must be understood in the widest manner.

The tools that we want to use in order to tackle such challenge are essentially based on the logic. The use of logic, in particular of proof theory, as an instrument for studying the computation arises from the Curry-Howard isomorphism, that brings in the light the deep nexus between programs and logical proofs, and moreover it is the starting point of a great part of the research in functional programming. In the last 20 years, starting from the birth of the Linear Logic, the proof theory has concurred to redefine the notion of computation, allowing a different approach, more refined, both to recursion theory and to complexity theory. The new notion of computation born from the linear logic and from the light logics, derived from it, is mainly based on the notion of interaction. From a computational perspective, the logic can be seen as the mutual interaction of proofs; the normalization (more precisely the cut-elimination) is the instrument where the interaction takes place. This fact allows to conceive languages in which the computational agents are fully described through their interactions. So the notion of computation becomes free from a specific computational model. Consequently it is possible to formulate analysis which can be tailored on the new computational paradigms arisen in the last years, as the biological and the quantistic ones. The research that we want to pursue is mainly foundational, and it can be ideally split in two principal objectives. We want to supply:

1. Foundational techniques for the analysis and verification of runtime properties of programs;
2. Computational theories modeling the correct interaction with the environment.

But we want also to do some experimental job: in particular site IV wants to exploit the use of the before techniques for a new release of the PELCR machine [Parallel Environment for optimal Lambda Calculus Reduction], which supports a parallel execution of lambda-calculus according to the optimal reduction strategy. This will be an experimental check for part of our foundational results.
The sites of this project gather researchers which already have a long experience of common work. Moreover this team has been increased by some new researchers with the specific expertise and skills in the study of language properties. The common experience has been achieved since they collaborated in two previous Italian national projects, PROTOCOLLO (from PROof TO COmputations through Linear LOgic, http://protocollo.di.unito.it) and FOLLIA (FOndazioni Logiche di LInguaggi Astratti di Programmazione, http://follia.di.unito.it). The first project was mainly focused on the study of some technical questions related to linear logic, relevant for the theoretical computer science. The second project, which was a continuation of the previous one, has enlarged the focus from the study of logical tools to the study of "Light Logic" and to "Ludics". Moreover, it started the study of program properties in the scope of mobile computations with limited resources. Many researchers, participant to this new project, belong also to the "Italian-french research network on logic and computation geometry" (2006-2008), in collaboration with the research french research network GEOCAL (Geometrie du Calcul, http://iml.univ-mrs.fr/geocal06/), now expired. Now many of us collaborate with the ANR project "CHOCO" (Curry-HOward pour la COncurrence) directed by Thomas Ehrhard (http://www.pps.jussieu.fr/ ehrhard/CHOCO/tout-definitif.pdf). Moreover some of them collaborate with the project NoCost (New tools for Complexity, Semantics and Types) directed by Patrick Baillot (http://www-lipn.univ-paris13.fr/nocost/). This synergy between different projects will surely become an important resource for the incoming project.

The new researchers joining our working group have skills in the area of the recursion theory and in the design of denotational models of programming languages. Thus, we expect that they can bring new inputs on the study of new computational models and on the transfer of properties from the logic to the study of programming properties. The total number of researchers belonging to this project is quite big: we are 28, and 10 of these are PhD students. In fact this project collects all people in Italy working on applications of Linear Logic to Computer Science, and we think that putting together our longstanding experience will be very useful for reaching our goals. Sincerely speaking, to maintain the cohesion of our group and the possibility of collaborating is for us a very important goal. Another very important goal for us is the formation of new young researchers. In fact we plan to use a relevant percentage of the total requested amount to research grants: we ask for 4 grants,

one for every site.

12 - Stato dell'arte

Testo italiano
In anni recenti, sono emerse due importanti novità nell'ambito dello studio della computazione. In primo luogo, sistemi logici, derivati dalla logica lineare, nei quali si puó esprimere la nozione di risorsa. In secondo luogo, nuovi paradigmi computazionali come quello biologico e quello quantistico. Inoltre, anche dentro i paradigmi tradizionali, lo studio della computazione ha seguito direzioni meno specifiche rispetto ai classici modelli computazionali. Nozione comune alla logica e alla computazione è quella di interazione, con diversi significati a seconda dei contesti.

Per ridurre la frammentazione di questa sezione, descriveremo il retroterra del progetto secondo grandi aree, ciascuna concernente diversi temi di ricerca specifici. Per mancanza di spazio, i riferimenti sono incompleti: una lista più estesa di essi si può trovare nello stato dell'arte di ciascuno dei modelli B.

1. Complessità Computazionale Implicita

La Complessità Computazionale Implicita (ICC) studia con approcci indipendenti dalle macchine teorie e applicazioni della complessità computazionale, con particolare attenzione agli approcci basati sulla logica. La maggior parte del lavoro consiste nel caratterizzare classi di complessità con sistemi logici (attraverso la corrispondenza di Curry-Howard).

In particolare la Logica Lineare Leggera di Girard (LLL), la sua versione affine (LAL) di Asperti e Roversi, e la Logica Soft (SLL) di Lafont offrono differenti caratterizzazioni della computazione in tempo polinomiale. La Logica Lineare Elementare (ELL), invece, caratterizza le computazioni elementari.

Al momento attuale non sono state trovate relazioni dirette fra LLL e i sistemi sottoricorsivi basati su altri principi, come ad esempio, la teoria della ricorsione studiata da Bellantoni e Hofmann. Un passo avanti in questa direzione è stato compiuto da Dal Lago, Martini e Roversi in [12] con l'introduzione di HOLRR un lambda-calcolo lineare esteso con costanti un operatore di ricorsione, in cui è possibile immergere direttamente la ricorsione ramificata di Leivant. L'area è troppo frammentata ed è necessario un cambio di paradigma allo scopo sia di mantenere la qualità della ricerca ad un alto livello, sia di produrre applicazioni nel contesto di linguaggi di programmazione concreti (dove è cruciale mantenere la classe dei programmi catturati piú larga possibile).

Ció di cui si ha bisogno è di contesti unificanti sia sintattici che semantici per l'analisi quantitativa di prove e programmi. Tali contesti dovrebbero essere potenti abbastanza da catturare molti sistemi logici e linguaggi di programmazione, ma semplici abbastanza per essere utili nel valutare le proprietà quantitative dei sistemi.

Baseremo il nostro lavoro sulla ben nota semantica (denotazionale) "statica" e sulla semantica "dinamica" della logica lineare (geometria dell'interazione e semantica dei giochi), e anche su nuove proposte emergenti dall'area della semantica dei contesti. I nostri principali punti di partenza per questa indagine semantica saranno:

+ la caratterizzazione di ELL e SLL in termini di clique ossessive fornita da Laurent e Tortora de Falco in [26]
+ il modello nella geometria dell'interazione di ELL introdotta da Baillot e Pedicini in [4]
+ la misura della complessità computazionale di una prova (della logica lineare) fornita dalla semantica a contesti, come dimostrato da Dal Lago in [10, 11].

Sistemi della ICC derivati dalle Logiche Leggere sono stati usati con successo per generare sistemi di assegnazione dei tipi per il lambda-calcolo, attraverso l'isomorfismo di Curry-Howard. Una serie di risultati sull'inferenza dei tipi del lambda-calcolo sottologiche della Logica Lineare sono stati pubblicati negli ultimi anni. ([9, 8, 37, 5, 21]).

Tali algoritmi permettono di verificare limiti sul tempo di riduzione e la correttezza di particolari strategie di riduzione ed in particolare implementazioni della beta riduzione. Un risultato recente è la caratterizzazione logica di PSPACE da parte di Gaboardi, Marion e Ronchi della Rocca in [20].

2. Modelli Computazionali

2.1. Computazione funzionale classica.
Il lambda-calcolo è il nucleo della programmazione funzionale, e c'è ancora bisogno di risultati sia sintattici che semantici su questo calcolo di base. Uno strumento utile a studiare il comportamento dei linguaggi di programmazione è il lambda-calcolo parametrico, introdotto da Ronchi della Rocca e Paolini, caratterizzato da una regola di riduzione parametrizzata da un insieme di valori di input. Scegliendo in modo diverso i valori di input si ottengono calcoli diversi, tra cui il lambda-calcolo classico e quello call-by-value di Plotkin. Questi due calcoli sono modelli teorici per la computazione call-by-name e call-by-value rispettivamente. Il lambda-calcolo parametrico permette di dimostrare in modo uniforme per ognuna delle sue istanze proprietà sintattiche come la confluenza e la standardizzazione [31]. Inoltre permette di dimostrare interessanti relazioni tra i vari calcoli. Ad esempio porta a una nuova caratterizzazione della normalizzazione forte [29, 30].

L'usuale semantica denotazionale del lambda-calcolo (semantica continua, stabile e fortemente stabile) è equazionalmente incompleta, ovvero non corrisponde con tutte le possibili semantiche operazionali del lambda-calcolo. Un lavoro recente ([34]) ha introdotto una nuova tecnica per dimostrare in modo uniforme l'incompletezza di tutte le semantiche denotazionali finora proposte. La maggior parte delle semantiche, che coinvolgono la monotonicità rispetto a qualche ordine parziale, non introduce una continuità di lambda-teorie. In [7] sono state caratterizzate quelle lambda-teorie che non sono indotte da nessun modello del lambda-calcolo dato in termini di modelli a grafo. Ulteriore lavoro è stato recentemente indirizzato allo studio di modelli algebrici del lambda-calcolo ([33]), della struttura a reticolo delle lambda-teorie ([27]) e delle proprietà di enumerabilità ricorsiva delle lambda-teorie associate ai modelli del lambda-calcolo ([6]).

PCF è il linguaggio paradigmatico proveniente dal lambda-calcolo. Il problema dell'astrazione piena per PCF è uno dei problemi della semantica della computazione più noti e rimasti aperti più a lungo. Plotkin ha formalizzato un'estensione di PCF pienamente astratta rispetto alla semantica continua. Similmente, in [28], è descritta un'altra estensione di PCF pienamente astratta rispetto alla semantica coerente. Gli spazi coerenti portarono alla scoperta della logica lineare (LL) e a un nuovo e profondo punto di vista sulla nozione logica di dualità. In effetti la negazione lineare (ovvero la "negazione senza regole strutturali") corrisponde alla dualità algebrica ed ha una controparte interattiva: un programma è eseguito in un certo ambiente e questo processo può essere analizzato dal punto di vista del programma o da quello dell'ambiente; la dualità è qui lo scambio tra queste due posizioni. Il sistema DIN delle reti di interazione differenziali è stato costruito grazie a nuovi modelli della LL ([15], [16]) basati su questo approccio interattivo alla dualità logica. Ancora incentrato sull'interazione, [19] tenta di stabilire delle connessioni tra strutture di eventi, semantica dei giochi e ludica.

2.2. Modello quantistico della computazione
Il Quantum Computing definisce un paradigma di computazione alternativo, basato sulla meccanica quantistica, la cui ricerca è iniziata con Feynman. Riguardo ai modelli computazionali, oltre alla Macchina di Turing Quantistica dobbiamo citare le Famiglie di Circuiti Quantistici [13], formalizzate in [39] ed equivalenti alle Macchine di turing Quantistiche, visto che hanno un ruolo fondamentale nello studio della complessità computazionale quantistica. Negli ultimi anni è stato compiuto un grande sforzo per definire linguaggi paradigmatici di programmazione quantistica, e in particolare linguaggi funzionali ([35, 38, 36]), tutti inspirati dalla logica lineare. Tuttavia manca ancora un linguaggio paradigmatico analogo al lambda-calcolo per la programmazione funzionale.

In [23], J.-Y. Girard propone di riformulare la semantica della computazione chiamata GOI nell'ambito delle algebre di von Neumann. Guidati da questo nuovo approccio, Pedicini e Piazza hanno introdotto un modello per le Macchine di Turing che condivide lo stesso spirito della GOI tradizionale, dove la computazione è modellata dall'iterazione di un funzionale di valutazione. Il funzionale di valutazione è ottenuto come un endomorfismo definito su una sottoalgebra a dimensione finita dell'algebra di von Neumann costruita sul gruppo. Le configurazioni delle Macchine di Turing sono immerse nello spazio di Hilbert delle serie formali in quadrato sommabili indicizzate da elementi di G. Scegliendo un particolare G, un gruppo localmente finito, otteniamo una complessità implicita limitata per le macchine di Turing rappresentabili.

2.3. Computazioni non deterministiche e concorrenti
Partendo da un'analisi di un'interpretazione semantica di LL con spazi vettoriali, Ehrhard e Regnier hanno proposto un'estensione con operatori differenziali del

lambda-calcolo, il cosiddetto lambda-calcolo differenziale [18]. La regola di riduzione in questo calcolo coinvolge una scelta non-deterministica, per cui potrebbe essere utilizzato per studiare computazioni non deterministiche. L'analisi semantica che ha condotto all'introduzione del lambda-calcolo differenziale ha anche portato a un'estensione differenziale delle reti di prova della logica lineare [14]. Una delle proprietà più interessanti delle reti di interazione differenziali è il fatto che estendono la simmetria del sistema anche alle modalità esponenziali. In DIN, la somma formale definita sui termini del lambda-calcolo differenziale corrisponde alla somma formale di reti. Interpretando tale somma come una scelta non deterministica, possiamo definire un calcolo a reti di interazione non deterministico in cui Ehrhard e Laurent hanno codificato il pi-calcolo finitario [17].

3 Riduzione ottimale

Le reti di prova sono una rappresentazione delle prove mediante la teoria dei grafi che porta in primo piano la loro natura geometrica e fornisce un approccio geometrico alla computazione (cf. la geometria dell'interazione [22]). Il sistema PELCR (Parallel Environment for optimal Lambda-Calculus Reduction), sviluppato all'IAC, consiste di un'implementazione parallela della strategia di riduzione ottimale per il lambda-calcolo, basata sulle reti di prova come modello computazionale ([32]). PELCR sfrutta una strategia per la Riduzione Virtuale Orientata, e più precisamente la half combustion, che permette di desincronizzare i passi di riduzione condivisa dell'algoritmo di Lamping.

REFERENZE

[1] A. Asperti, P. Coppola, and S. Martini. (Optimal) duplication is not elementary recursive. In POPL'00, pages 96-107. ACM, 2000.

[2] A. Asperti and L. Roversi. Intuitionistic light affine logic. TOCL, 3(1):1-39, 2002.

[3] P. Baillot, U. Dal Lago, and P. Coppola. Light logics and optimal reduction: Completeness and complexity. In LICS'07, pages 421-430. IEEE, 2007.

[4] P. Baillot and M. Pedicini. Elementary complexity and geometry of interaction. Fundamenta Informaticae, 45(1-2):1-31, 2001.

[5] P. Baillot and K. Terui. A feasible algorithm for typing in elementary affine logic. In TLCA'05, pages 55-70, 2005.

[6] C. Berline, G. Manzonetto, and A. Salibra. Lambda theories of effective lambda models. In CSL'07, pages 298-312. LNCS 4646, 2007.

[7] A. Bucciarelli and A. Salibra. The sensible graph theories of lambda calculus. In LICS'04, pages 276-285. IEEE, 2004.

[8] P. Coppola, U. Dal Lago, and S. Ronchi Della Rocca. Elementary affine logic and the call by value lambda calculus. In Pawel Urzyczyn, editor, TLCA'05, volume 3461 of LNCS, pages 131-145. Springer, 2005.

[9] P. Coppola and S. Martini. Optimizing optimal reduction. a type inference algorithm for elementary affine logic. TOCL, 7(2):219-260, 2006.

[10] U. Dal Lago. The geometry of linear higher-order recursion. In LICS'05, pages 366-375. IEEE, 2005.

[11] U. Dal Lago. Context semantics, linear logic and computational complexity. In LICS'06, pages 169-178. IEEE, 2006.

[12] U. Dal Lago, S. Martini, and L. Roversi. Higher-order linear ramified recurrence. In TYPES'03, volume 3085 of LNCS. Springer, 2004.

[13] D. Deutsch. Quantum computational networks. Proc. of the Royal Society of London Ser. A, A425:73-90, 1989.

[14] T. Ehrhard and Laurent Regnier. Differential interaction nets. ENTCS, 123:35-74, 2005.

[15] Thomas Ehrhard. On köthe sequence spaces and linear logic. Math. Struct. Comput. Sci., 12(5):579-623, 2002.

[16] Thomas Ehrhard. Finiteness spaces. Math. Struct. Comput. Sci., 15(4):615-646, 2005.

[17] Thomas Ehrhard and Olivier Laurent. Interpreting a finitary pi-calculus in differential interaction nets. In Luís Caires and Vasco Thudichum Vasconcelos, editors, CONCUR 2007, volume 4703 of Lecture Notes in Computer Science, pages 333-348. Springer, 2007.

[18] Thomas Ehrhard and Laurent Regnier. The differential lambda-calculus. TCS, 309(1):1-41, 2003.

[19] Claudia Faggian and Mauro Piccolo. Ludics is a model for the finitary linear pi-calculus. In Proc. of TLCA, Typed Lambda Calculi and Applications, LNCS, 2007.

[20] Marco Gaboardi, Jean-Yves Marion, and Simona Ronchi Della Rocca. A logical account of PSPACE. POPL'08, To appear, 2008.

[21] Marco Gaboardi and Simona Ronchi Della Rocca. A soft type assignment system for lambda-calculus. In CSL'07, volume 4646 of LNCS, pages 253-267, 2007.

[22] J.-Y. Girard. Geometry of interaction. I. Interpretation of system F. In Logic colloquium '88, pages 221-260. North-Holland, 1989.

[23] Jean-Yves Girard. Le point aveugle - Cours de logique - Volume II. Visions des sciences. Hermann, Paris, 2007.

[24] S. Guerrini, S. Martini, and A. Masini. Proof nets, garbage, and computations. TCS, 253(2):185-237, 2001.

[25] Stefano Guerrini, Simone Martini, and Andrea Masini. Coherence for sharing proof-nets. TCS, 294:379-409, 2003.

[26] O. Laurent and L. Tortora de Falco. Obsessional cliques: a semantic characterization of bounded time complexity. In LICS'06. IEEE, 2006. To appear.

[27] S. Lusin and A. Salibra. The lattice of lambda theories. J. Logic Comp., 14(3):373-394, 2004.

[28] Luca Paolini. A stable programming language. Inf. and Comp., 204(3):339-375, 2006.

[29] Luca Paolini, Elaine Pimentel, and Simona Ronchi Della Rocca. Lazy strong normalization. In ITRS'04, volume 136C of ENTCS, pages 103-116, 2004.

[30] Luca Paolini, Elaine Pimentel, and Simona Ronchi Della Rocca. An operational characterization of strong normalization. LNCS, 3921:367-381, 2006.

[31] Luca Paolini and Simona Ronchi Della Rocca. Parametric parameter passing lambda-calculus. Inf. and Comp., 189(1):87-106, 2004.

[32] M. Pedicini and F. Quaglia. Pelcr: A parallel implementation for optimal lambda-calculus reduction. TOCL, 2006.

[33] A. Salibra. Nonmodularity results for lambda calculus. FI, 45(4):379-392, 2001.

[34] A. Salibra. Topological incompleteness and order incompleteness of the lambda calculus. TOCL, 4(3):379-401, 2003.

[35] P. Selinger. Towards a quantum programming language. MSCS, 14(4):527-586, 2004.

[36] P. Selinger and B. Valiron. A lambda calculus for quantum computation with classical control. In TLCA'05, volume 3461 of LNCS, pages 354-368, 2005.

[37] K. Terui and P. Baillot. Light types for polynomial time computation in lambda-calculus. In LICS'04, pages 266-275. IEEE, 2004.

[38] A. van Tonder. A lambda calculus for quantum computation. SIAM J. Comput., 33(5):1109-1135, 2004.

[39] A. Chi-Chih Yao. Quantum circuit complexity. In FOCS'93, pages 352-361. IEEE, 1993.

Testo inglese

In recent years, two important novelties emerged, as far as the study of computation is concerned. First, logical systems, derived from linear logic, in which we can express the notion of resource. Second, new computational paradigms like the biological and the quantum computing ones. Moreover, even inside the traditional paradigms, the study of computation followed directions less specifically related to classical computational models. Common to the notions of logic and computation is the notion of interaction, with different meanings in the various contexts.

To reduce the fragmentation of this section, we will describe the background of the project along wide areas each encompassing several specific research themes. Due to the lack of space, the more basic references are missing: a more extended list of references can be found in the background part of each one of the Modelli B.

1. Implicit Computational Complexity

Implicit computational complexity (ICC) studies theory and applications of machine-independent approaches to computational complexity, with particular emphasis on approaches based on logic. Most work consists in characterizations of complexity classes by logical systems (via the so-called Curry-Howard correspondence). In particular, Light Linear Logic (LLL) of Girard, its affine version (LAL) by Asperti and Roversi, and Lafont's Soft Logic (SLL) offer different characterizations of polynomial time computations. Elementary Linear Logic (ELL), instead, characterizes elementary computations. At present no direct relation has been found between LLL and subrecursive systems based on other principles like, e.g., the recursion theory, studied by Bellantoni and Hofmann. A step forward in this direction has been made by Dal Lago, Martini and Roversi in [12] with the introduction of HOLRR a linear lambda-calculus extended with constants and a recursor, in which it is possible to embed directly Leivant's ramified recursion. The area is too fragmented and a paradigm shift is necessary in order to both maintain the research quality at a high level and enforce applications in the context of concrete programming languages (where keeping the class of captured programs as large as possible is crucial). What is needed are some unifying syntactic and semantic frameworks for the quantitative analysis of proofs and programs. Such frameworks should be powerful enough to capture many logical systems and programming languages, but simple enough to be useful in evaluating quantitative properties of systems. We will base our work on the well-established "static" (denotational) semantics so as on the "dynamic" semantics of linear logic (geometry of interaction and game semantics) and also on new proposals in the area of context semantics. Our main starting points for this semantic investigation will be:

+ the characterization of ELL and SLL in terms of obsessional cliques given by Laurent and Tortora de Falco in [26]
+ the geometry of interaction model of ELL introduced by Baillot and Pedicini in [4]
+ the measure of the inherent computational complexity of a (linear logic) proof given by context semantics, as shown by Dal Lago in [10, 11].

ICC systems derived from Lights Logics have been fruitfully used for generating type assignment systems for lambda-calculus, through the Curry-Howard isomorphism. A series of results about type inference of lambda calculus in sublogics of Linear Logic have been published in the last years ([9, 8, 37, 5, 21]). Such algorithms allow to verify bounds on the reduction time and the correctness of particular reduction strategies and of particular implementations of the beta reduction. A recent result is a logical characterization of PSPACE by Gaboardi, Marion and Ronchi della Rocca in [20].

2. Computational Models

2.1. Classical functional computation
Lambda-calculus is the kernel of the functional programming paradigm and results on this basic calculus are still needed, both from a syntactical and a semantic point of view. A useful tool for studying the behaviour of programming languages is the parametric lambda-calculus, which has been introduced by Ronchi della Rocca and Paolini, characterized by a reduction rule parametrized over a set of input values. Different choices of input values give rise to different calculi, among which the classical lambda-calculus and Plotkin's call-by-value lambda-calculus. These two calculi are theoretical models for call-by-name and call-by-value computation, respectively. Parametric lambda-calculus allows to prove uniformly syntactic properties, like confluence and standardization, for each of its instances [31]. Furthermore, it allows to prove interesting relations among calculi. For example, it allows for a new characterization of strong normalization [29, 30].
The usual denotational semantics of lambda calculus (continuous, stable and strongly stable semantics) are equationally incomplete, namely they do not match all the possible operational semantics of lambda calculus. Recent work (e.g., [34]) has introduced a new technique to uniformly prove the incompleteness of all the denotational semantics of the untyped lambda calculus which have been proposed so far. Most of the semantics, which involve monotonicity with respect to some partial order, fail to induce a continuum of lambda theories. In [7] there is a characterization of those lambda theories that are not induced by any model of lambda calculus given in terms of graph models. Further recent work has been devoted to the investigation of algebraic lambda calculus models ([33]), to the study of the structure of the lattice of lambda theories ([27]) and to the properties of recursive enumerability of the lambda theories associated with lambda calculus models ([6]).
PCF is the paradigmatic language designed from typed lambda-calculus. The full abstraction problem for PCF is one of the best-known and longest standing problems in the semantics of computation. Plotkin formalized an extended version of PCF fully abstract with respect to the continuous semantics. Likewise, in [28], another extension of PCF which is fully abstract with respect to the coherent semantics is described.
Coherent spaces led to linear logic and to a remarkable insight on the logical notion of duality. Indeed, linear negation (that is "negation without structural rules") corresponds to algebraic duality and has an interactive counterpart: a program is executed in a given environment and this process can be either analyzed from the point of view of the program or from the one of the environment; duality is here the switch between these two positions. The system DIN of Differential Interaction Nets has been built thanks to new models of LL ([15], [16]) based on this interactive approach to logical duality. Also focused on interaction, an attempt to establish connections between events structures, game semantics and ludics is in [19].

2.2. Quantum model of computation
Quantum Computing defines an alternative computational paradigm, based on quantum mechanics. The starting point is by Feynman. Regarding the computational models, in addition to the Quantum Turing Machine we must cite the Quantum Circuit Families [13], as formalized in [39], since they are of fundamental use for the study of quantum complexity theory. Quantum Circuit Families are equivalent to Quantum Turing Machines. In the last years a great effort has been done in order to define paradigmatic quantum programming languages, and in particular functional languages [35, 38, 36]. All of them are inspired by linear logic. A paradigmatic language, as lambda calculus is for functional computation, is still missing.
In [23], J.-Y. Girard proposes to reshape the semantics of computation called GOI in the realm of von Neumann algebras. Being driven by this new approach, Pedicini and Piazza introduced a model of Turing Machines which shares the same spirit of the traditional GOI, where computation is modeled by the iteration of an evaluating functional. The evaluation functional is obtained as an endomorphism defined on a finite dimensional subalgebra of von Neumann group algebra. Configurations of the Turing Machines are embedded in the Hilbert space of the square summable formal series indexed by elements of G. By choosing G, a locally finite group, we obtain an implicit complexity bound on the representable Turing Machines.

2.3. Non deterministic and concurrent computations
Starting from an analysis of a semantic interpretation of Linear Logic in linear vector spaces Ehrhard has proposed with Regnier an extension of lambda-calculus with differential operators, the so-called differential lambda-calculus [18]. The reduction rule in this calculus involves a non-deterministic choice, so it could be used for studying non-deterministic computations. The semantic analysis that led to the introduction of differential lambda-calculus, led also to a differential extension of linear logic proof-nets [14]. One of the more interesting property of differential interaction nets is that they extend the symmetry of the system to the exponential modalities too. In DIN, the formal sum defined on terms of differential lambda-calculus corresponds to the formal sum of nets. By interpreting such a sum of nets as a non-deterministic choice, we can define a non-deterministic interaction net calculus in which Ehrhard and Laurent have encoded finitary pi-calculus [17].

3 Optimal reduction

Proof-nets are a graph-theoretical representation of proofs that brings to the fore the geometric nature of proofs and provides a geometric approach to computation (see the geometry of interaction [22]). PELCR (Parallel Environment for optimal Lambda-Calculus Reduction) system developed at IAC, which gives a parallel

implementation of the optimal reduction strategy on lambda-calculus, is based on proof-nets as computational model ([32]). PELCR relies on a strategy for Directed Virtual Reduction, namely half combustion which in some way desynchronize sharing reduction.

REFERENCES

[1] A. Asperti, P. Coppola, and S. Martini. (Optimal) duplication is not elementary recursive. In POPL'00, pages 96-107. ACM, 2000.

[2] A. Asperti and L. Roversi. Intuitionistic light affine logic. TOCL, 3(1):1-39, 2002.

[3] P. Baillot, U. Dal Lago, and P. Coppola. Light logics and optimal reduction: Completeness and complexity. In LICS'07, pages 421-430. IEEE, 2007.

[4] P. Baillot and M. Pedicini. Elementary complexity and geometry of interaction. Fundamenta Informaticae, 45(1-2):1-31, 2001.

[5] P. Baillot and K. Terui. A feasible algorithm for typing in elementary affine logic. In TLCA'05, pages 55-70, 2005.

[6] C. Berline, G. Manzonetto, and A. Salibra. Lambda theories of effective lambda models. In CSL'07, pages 298-312. LNCS 4646, 2007.

[7] A. Bucciarelli and A. Salibra. The sensible graph theories of lambda calculus. In LICS'04, pages 276-285. IEEE, 2004.

[8] P. Coppola, U. Dal Lago, and S. Ronchi Della Rocca. Elementary affine logic and the call by value lambda calculus. In Pawel Urzyczyn, editor, TLCA'05, volume 3461 of LNCS, pages 131-145. Springer, 2005.

[9] P. Coppola and S. Martini. Optimizing optimal reduction. a type inference algorithm for elementary affine logic. TOCL, 7(2):219-260, 2006.

[10] U. Dal Lago. The geometry of linear higher-order recursion. In LICS'05, pages 366-375. IEEE, 2005.

[11] U. Dal Lago. Context semantics, linear logic and computational complexity. In LICS'06, pages 169-178. IEEE, 2006.

[12] U. Dal Lago, S. Martini, and L. Roversi. Higher-order linear ramified recurrence. In TYPES'03, volume 3085 of LNCS. Springer, 2004.

[13] D. Deutsch. Quantum computational networks. Proc. of the Royal Society of London Ser. A, A425:73-90, 1989.

[14] T. Ehrhard and Laurent Regnier. Differential interaction nets. ENTCS, 123:35-74, 2005.

[15] Thomas Ehrhard. On köthe sequence spaces and linear logic. Math. Struct. Comput. Sci., 12(5):579-623, 2002.

[16] Thomas Ehrhard. Finiteness spaces. Math. Struct. Comput. Sci., 15(4):615-646, 2005.

[17] Thomas Ehrhard and Olivier Laurent. Interpreting a finitary pi-calculus in differential interaction nets. In Luís Caires and Vasco Thudichum Vasconcelos, editors, CONCUR 2007, volume 4703 of Lecture Notes in Computer Science, pages 333-348. Springer, 2007.

[18] Thomas Ehrhard and Laurent Regnier. The differential lambda-calculus. TCS, 309(1):1-41, 2003.

[19] Claudia Faggian and Mauro Piccolo. Ludics is a model for the finitary linear pi-calculus. In Proc. of TLCA, Typed Lambda Calculi and Applications, LNCS, 2007.

[20] Marco Gaboardi, Jean-Yves Marion, and Simona Ronchi Della Rocca. A logical account of PSPACE. POPL'08, To appear, 2008.

[21] Marco Gaboardi and Simona Ronchi Della Rocca. A soft type assignment system for lambda-calculus. In CSL'07, volume 4646 of LNCS, pages 253-267, 2007.

[22] J.-Y. Girard. Geometry of interaction. I. Interpretation of system F. In Logic colloquium '88, pages 221-260. North-Holland, 1989.

[23] Jean-Yves Girard. Le point aveugle - Cours de logique - Volume II. Visions des sciences. Hermann, Paris, 2007.

[24] S. Guerrini, S. Martini, and A. Masini. Proof nets, garbage, and computations. TCS, 253(2):185-237, 2001.

[25] Stefano Guerrini, Simone Martini, and Andrea Masini. Coherence for sharing proof-nets. TCS, 294:379-409, 2003.

[26] O. Laurent and L. Tortora de Falco. Obsessional cliques: a semantic characterization of bounded time complexity. In LICS'06. IEEE, 2006. To appear.

[27] S. Lusin and A. Salibra. The lattice of lambda theories. J. Logic Comp., 14(3):373-394, 2004.

[28] Luca Paolini. A stable programming language. Inf. and Comp., 204(3):339-375, 2006.

[29] Luca Paolini, Elaine Pimentel, and Simona Ronchi Della Rocca. Lazy strong normalization. In ITRS'04, volume 136C of ENTCS, pages 103-116, 2004.

[30] Luca Paolini, Elaine Pimentel, and Simona Ronchi Della Rocca. An operational characterization of strong normalization. LNCS, 3921:367-381, 2006.

[31] Luca Paolini and Simona Ronchi Della Rocca. Parametric parameter passing lambda-calculus. Inf. and Comp., 189(1):87-106, 2004.

[32] M. Pedicini and F. Quaglia. Pelcr: A parallel implementation for optimal lambda-calculus reduction. TOCL, 2006.

[33] A. Salibra. Nonmodularity results for lambda calculus. FI, 45(4):379-392, 2001.

[34] A. Salibra. Topological incompleteness and order incompleteness of the lambda calculus. TOCL, 4(3):379-401, 2003.

[35] P. Selinger. Towards a quantum programming language. MSCS, 14(4):527-586, 2004.

[36] P. Selinger and B. Valiron. A lambda calculus for quantum computation with classical control. In TLCA'05, volume 3461 of LNCS, pages 354-368, 2005.

[37] K. Terui and P. Baillot. Light types for polynomial time computation in lambda-calculus. In LICS'04, pages 266-275. IEEE, 2004.

[38] A. van Tonder. A lambda calculus for quantum computation. SIAM J. Comput., 33(5):1109-1135, 2004.

[39] A. Chi-Chih Yao. Quantum circuit complexity. In FOCS'93, pages 352-361. IEEE, 1993.

13 - Articolazione del Progetto e tempi di realizzazione

Testo italiano
Una descrizione scientifica del programma di ricerca può essere consultata al punto 14: "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione". In particolare, i temi specifici sono lì descritti. In questa sezione noi ci focalizzeremo sull'organizzazione del lavoro, logistica e pratica.

La ragione principale per cui un progetto di ricerca fondazionale come CONCERTO ha bisogno di di un finanziamento importante, oltre alla formazione di nuovi ricercatori tramite l'istituzione di assegni di ricerca, è la necessità di interazione e collaborazione tra ricercatori che lavorano in luoghi geograficamente distanti l'un l'altro. I ricercatori afferenti a questo progetto hanno collaborato molto nel passato, producendo importanti risultati ([1, 2, 3, 8, 24, 25] sono alcuni esempi citati al punto 12 "Stato dell'Arte"). Come in ogni altro progetto di ricerca di base, è assolutamente vitale per i componenti incontrarsi personalmente a intervalli regolari. Le competenze complementari delle varie unità insieme a una stretta interazione facilitano la nascita di conoscenze nuove. Sfortunatamente, l'interazione remota (tramite posta elettronica o telefono) diventa utile solo dopo che i problemi cruciali siano stati posti e le tecniche principali siano state isolate in una sessione di lavoro comune. Quindi noi pensiamo di usare gran parte delle risorse assegnatici per promuovere la reciproca collaborazione.

Interdipendenze tra i temi. I temi presentati a punto 14, "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione" hanno molte interdipendenze e influenze che sono schematizzate in quanto segue. - "Fondazioni Logiche" (tema 2.5) influenza tutti gli altri temi.

- Lo sviluppo di "Modelli Semantici" (tema 1.4) influenza quello di "Semantiche per ICC´ ' (tema 2.2).

- "Unificazione di sistemi ICC " (tema 2.1), "Semantiche per ICC´' (tema 2.2) e "Sistemi ICC " (tema 1.2) sono correlati l'un l'altro, quindi noi ci aspettiamo che si influenzino a vicenda.

- Lo sviluppo del goal "Riduzioni ottimali" (tema 1.1) potrebbe influenzare "Modelli per la concorrenza e per l'interazione" (tema 2.3).

- Infine "Modelli per la concorrenza e l'interazione" (tema 2.3) sembra essere rilevante sia per "Sicurezza e clash-freeness" (tema 1.3) e "Nuovi paradigmi computazionali" (tema 2.4).

Partendo dal "grafo delle inter-dipendenze" illustrato sopra, noi vogliamo facilitare la collaborazione in molti modi.

Organizzazione interna. Noi pensiamo di nominare in coordinatore per ogni tema del progetto, eventualmente aiutato da sotto-coordinatori per sotto-temi specifici. Il coordinatore si occuperà di ogni aspetto del progetto concernente il suo tema. In particolare, controllerà l'andamento dei lavori, raccogliendo i risultati intermedi e curando la loro distribuzione sia all'interno che all'esterno di CONCERTO. Sarà responsabile dell'organizzazione di workshop specifici. Ci si aspetta che i coordinatori di temi correlati lavorino in stretta collaborazione, per esempio organizzando workshop congiunti. Eventuali sotto-coordinatori avranno compiti più specifici, aiuteranno il coordinatore nell'organizzazione di specifici workshop tematici e terranno informato il coordinatore dell'andamento dei lavori nel sotto-tema loro assegnato.

Workshop tematici. Lo strumento portante per la collaborazione e la disseminazione dei risultati consisterà in una serie di workshop tematici, ognuno focalizzato su uno o più (sotto)temi, che saranno organizzati a cura dei coordinatori e sotto-coordinatori. Questi workshop saranno usati per discussioni scientifiche su un tema ben delimitato, avranno durata di uno o due giorni e saranno aperti a tutti gli interessati. Ci aspettiamo che abbiano un numero di partecipanti tra 5 e 10. Si intendono organizzati per la presentazione di nuovi risultati e per stabilire punti specifici per il successivo lavoro. Quindi consisteranno essenzialmente nella presentazione di problemi e nella discussione su tentativi di soluzioni.

Convegni. Organizzeremo tre convegni, diretti a tutti i partecipanti:

+ All'inizio del progetto si terrà il primo convegno, con lo scopo di discutere gli aspetti organizzativi, e di eleggere coordinatori e sotto-coordinatori, stabilire tentativamente date e luoghi per altri convegni e workshop. Inoltre verranno presentati alla comunità i nuovi ricercatori entrati a far parte del gruppo, e ci saranno alcune presentazioni dello stato dell'arte dei vari problemi.
+ Il secondo convegno si terrà un anno dopo l'inizio. Servirà per analizzare i progressi compiuti e per presentare i risultati ottenuti. I coordinatori dovranno presentare lo stato di avanzamento del progetto. Se necessario, durante il convegno si prenderanno le misure opportune per ribilanciare il progetto. Si terranno presentazioni tutoriali su argomenti correlati ai temi del progetto, in particolare relativi allo sviluppo di tesi di dottorato.
+ Il terzo convegno si terrà alla fine del progetto, per presentare i risultati finali.

Inviteremo a ogni convegno ricercatori che stanno collaborando con noi all'interno di altri progetti internazionali, in particolare ricercatori dei progetti francesi citati al punto 1 "Obiettivi". Questa interazione ci sembra essenziale per avere un preciso riscontro sullo stato del progetto.

Tempi. Sarebbe sforzato imporre una sequenzializzazione temporale di tutte le linee di ricerca del progetto. Però è possibile suddividere i temi in due classi:

+ La prima classe contiene quei temi che fanno parte di una linea di ricerca di lungo respiro, per cui esistono già specifiche competenze, e per il raggiungimento dei quali possiamo già prevedere puntuali soluzioni tecniche. Fanno parte di questa classe i temi 1.1, 1.2, 1.4, 2.2, 2.5.
+ La seconda classe invece comprende temi che rappresentano sfide per la nostra comunità di ricerca. Sono elementi di questa classe i temi 1.3, 2.1, 2.3 ,2.4.

Nel primo anno, noi ci occuperemo essenzialmente dei temi nella prima classe, e nel secondo ci dedicheremo agli obiettivi più avanzati della seconda classe. Durante il secondo convegno annuale faremo un controllo generale del progetto e in quella sede potremo riallocare nel secondo anno i temi del primo anno che avranno avuto problemi o ritardi.

Supporti elettronici alla collaborazione. Installeremo una pagina web dedicata al progetto, dove manterremo tutte le informazioni riguardanti i partecipanti, i convegni, gli workshop tematici. Questa pagina conterrà inoltre le referenze scientifiche e le bozze dei lavori in preparazione, contenenti i risultati parziali. Inoltre sarà installato un sistema di Subversion, per aiutare i partecipanti nella preparazione dei documenti comuni.

Testo inglese
A scientific description of the research program can be found in point 14 "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione". In particular, global and local research goals are described there. In this section, we will focus on the organization, logistics and practical implementation of this project.

The main reason why foundational research projects like CONCERTO need substantial funding, behind the formation of new researchers through research grants, lies in the necessity for interaction and collaboration between researchers working at remote sites. The researchers involved in this project have collaborated very much in the past, producing remarkable results ([1, 2, 3, 8, 24, 25] are few examples referenced in Section "Stato dell'Arte"). Like in any other basic research project, it is absolutely vital for the involved researchers to meet personally on a regular basis. The complementary competencies of the groups together with strict interaction greatly improves the generation of new knowledge. Unfortunately, remote interaction (via email or phone) become useful only after crucial problems have been posed and key techniques have been isolated in common work sessions. Therefore, we plan to spend many resources in order to promote collaboration.

Goal inter-dependencies. Goals presented in point 14, "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione" have many inter-dependencies sketched after. - "Logical Foundations" (goal 2.5) have an influence on all other goals.

- The development of "Semantic Models" (goal 1.4) have an influence on that of "Semantics of ICC ´ ' (goal 2.2).

- Moreover, "Unifying ICC systems" (goal 2.1), "Semantics of ICC´' (goal 2.2) and "ICC Systems" (goal 1.2) are interrelated, so we expect that results can influence

each other.

- The development of the goal "Optimal Reduction" (1.1) could influence "Modeling concurrency and interaction" (goal 2.3).

- Last, "Modeling concurrency and interaction" (goal 2.3) seems to be relevant for both "Security and clash-freeness" (goal 1.3) and "New Computational Paradigms" (goal 2.4).

Starting from the previous "graph of inter-dependencies" we plan to support the collaboration in many way.

Internal Organization. We plan to have a coordinator for each goal of the project, perhaps supported by subcoordinators for specific subgoals. The coordinator will take care of any aspect of the project concerning its goal. In particular, he will assess the progress towards the goal, keeping track of all the obtained intermediate results, and assuring proper dissemination of them inside and outside CONCERTO. He will be responsible for organizing periodic workshops. We expect that coordinators of related goals will work in strict collaboration, for example by organizing joint workshops. Eventual subcoordinators will have more specific duties, they will help the coordinator in the organization of periodic thematic workshops and report to the scientific coordinator about the progress of the subgoal.

Thematic Workshops. The main backbone for collaboration and dissemination will consists in a series of thematic, intermediate workshops, focusing on one or more (sub)goals. They will be organized by goal and subgoal coordinators. These workshops are intended as lightweight, purely scientific meetings. Lasting one or two days, open to anyone interested but focusing on a very specific theme, we expect them to have an attendance of 5 to 10 people each. They will be devoted to the presentation of new results and ongoing work. Time will be especially devoted to common discussion and actual working out of problems and tentative solutions.

Meetings. We will organize three general meetings:

+ The kickoff meeting will be held at the beginning of the project. The starting organizing aims will be discussed, coordinators and subcoordinators will be elected, expected meetings and workshops (together with presumed dates and meeting-place) will be established. Moreover the new people joining our research community will be presented, and some state-of-art presentations will be given.
+ The second meeting will held one year after the beginning of the project. Progress will be analyzed and new results will be presented. The coordinators will be responsible for a general assessment of the progress. The meeting will take actions to keep the project on track, if needed. Some tutorial presentation on arguments related to our goals will be given, specially related to PhD thesis development.
+ The third meeting will held at the end of the project. Final results will be presented.

At each of these meetings, we will invite researchers who are collaborating with us in other international projects. Special attention will be devoted to the French projects we mentioned in point 11, "Obiettivi". This interaction is essential in order to get proper feedback about the status of the project.

Timelines. Forcing all the research lines in this project to follow a strict timeline would be unnatural and artificial. We nevertheless subdivide the various goals into two classes:

+ The first class include goals which are part of long-running research efforts, for which specific expertise is already present, and for which we foresee - already at this stage - solution techniques. Goals 1.1, 1.2, 1.4, 2.2, 2.5 are part of this class.
+ The second class, on the other hand, include goals which represent challenges to our research community. Goals 1.3, 2.1, 2.3 ,2.4 are elements of this class.

In the first year, we will mainly focus our attention on the subgoals in the first class, while in the second year the emphasis will go to the more advanced research objectives in the second class. The second year meeting is the main checkpoint of the project - goals will be re-allocated to the second year in case of failure or insufficient progress in the first year.

Electronic Supports to collaboration. A HomePage of the project will be installed, where all the information about participants, meeting, thematic workshops will be inserted. This page will contain also all the scientific references, and the drafts of the papers in preparation, collecting the partial results. Moreover a Subversion system will be installed, to help participants in the preparation of the common documents.

14 - Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione

Testo italiano
Come specificato nella Sezione 11, "Obiettivi finali che il Progetto si propone di raggiungere", i ricercatori coinvolti in questo progetto hanno una lunga esperienza di lavoro comune, testimoniato da molte pubblicazioni in collaborazione, anche tra ricercatori di diverse unità. Siamo infatti interessati agli stessi problemi, ma possiamo contribuire al loro studio partendo da basi differenti, ma simili. L'unità III raccoglie soprattutto esperti in logica, specialmente Logica Lineare e Reti di Prova. Quindi il suo contributo consisterà essenzialmente in una analisi approfondita delle basi logiche delle nozioni di computazione, interazione e uso delle risorse. Le unità I e II sono composte di esperti della semantica dei linguaggi di programmazione, in particolare lambda-calcoli, e sono interessati alla definizione di metodi formali basati sulla logica per dimostrare proprietà di programmi. I membri dell'unità IV hanno una base scientifica logica e matematica, e hanno dimostrato di essere in grado di realizzare,, a partire da tali basi, applicazioni interessanti e affatto banali. La presenza di queste conoscenze ed esperienze complementari ci permetteranno di raggiungere i nostri scopi.
Per mettere in evidenza la collaborazione e il ruolo di ogni unità, ci richiameremo ai due obiettivi principali definiti nel punto 11, e per ognuno di questi elencheremo gli scopi e le unità coinvolte. Per carenza di spazio, gli scopi saranno descritti in modo non esaustivo e senza riferimenti bibliografici; presentazioni più dettagliate si possono trovare nei vari Modelli B.
Consideriamo il primo obiettivo:
1. ***Tecniche fondazionali per l'analisi e a verifica di proprietà operazionali di programmi.***
La letteratura offre una gran varietà di strumenti per dimostrare proprietà di programmi, basati su tecniche operazionali, denotazionali, logiche. Negli ultimi 20 anni, con la nascita della Logica Lineare (LL), la teoria della dimostrazione ha concorso a ridefinire la nozione di computazione, ottenendo di conseguenza un approccio diverso, più raffinato, sia alla teoria della ricorsione che alla teoria della complessità. Inoltre le Logiche Leggere, varianti di LL che caratterizzano classi di complessità, hanno fornito una base formale per ragionare sull'uso delle risorse da parte dei programmi. Intendiamo partire da queste basi per progettare nuove tecniche per l'analisi e la verifica di proprietà operazionali di programmi. I problemi che vogliamo studiare all'interno dell'obiettivo 1 sono i seguenti.

+ Riduzione ottimale.
Unità coinvolte: I, II, IV
La logica lineare ha trovato molte applicazioni nello studio dei linguaggi funzionali. Una di esse riguarda la costruzione di interpreti basati su riduzione ottimale. Tutti queste interpreti (p.e., BOHM, Bologna Optimal Higher Order Machine), implementano la strategia ottimale sui lambda termini secondo la chiamata per nome. Intendiamo sviluppare una teoria della riduzione ottimale per la chiamata per valore. Il lambda calcolo parametrico sarà un utile strumento a tal fine.
Una implementazione parallela della riduzione del lambda-calcolo è PELCR (Parallel Environment for Optimal Lambda Calculus Reduction). Intendiamo estendere PELCR con meccanismi di controllo sulla complessità delle funzioni di libreria. Infine, non ci sono risultati che colleghino la quantità di lavoro effettivamente eseguito da ogni valutatore ottimale ed il numero di beta-riduzioni per raggiungere la forma normale. La definizione di un ragionevole modello di costo per il lambda-calcolo è un altro argomento che ci proponiamo di studiare.

+ Sistemi ICC.
Unità coinvolte: I,II,III
Questo argomento si suddivide in due parti: uno studio fondazionale e l'applicazione ai linguaggi di programmazione. Iniziamo dal primo. I sistemi di Complessità Computazionale Implicita (ICC) possono essere progettati su differenti basi: teoria della ricorsione, teoria delle prove, teorie dei tipi, ecc.: intendiamo formalizzare le relazioni tra essi. Il punto di partenza sarà il confronto di due ben noti sistemi caratterizzanti PTime: "Light affine logic" (LAL) e "Safe Recursion on Notation" (SRN). Vogliamo progettare una nuova versione di LAL, semplificandone la struttura in modo da facilitarne il confronto con SRN.

Considereremo quindi le applicazioni di ICC ai linguaggi di programmazione, in particolare sistemi di assegnamento di tipo per linguaggi essenziali, dove i tipi testimonieranno, oltre alla correttezza, anche proprietà di complessità (sia in spazio che tempo). Un problema fondamentale è allargare la classe dei programmi esprimibili (in opposizione alla semplice definizione estensionale di funzioni). Tenteremo di arricchire il sistema con tipi polimorfi, ricorsivi ed intersezioni. Altro problema è il progetto di sistemi decidibili che permettano verifiche statiche a tempo di compilazione, in stile ML. Limitare l'uso del polimorfismo è un approccio possibile, purché una tale restrizione preservi la completezza dell'espressività funzionale. Chiaramente la soluzione ottimale sarebbe ottenere un sistema soddisfacente entrambe le richieste: decidibilità ed una buona espressività algoritmica.
Un'ulteriore questione è l'estensione dei sistemi ICC a rilevanti varianti del lambda-calcolo. Vogliamo estendere sistemi di tipo per l'ICC al lambda-calcolo differenziale. Questo ambizioso obiettivo corrisponde a estendere l'approccio dell'ICC a classi di complessità nondeterministiche, dato che la scelta nondeterministica è insita nella riduzione del lambda-calcolo differenziale.

+ Sicurezza e clash-freeness.
Unità coinvolte: IV
Nei modelli computazionali distribuiti, sicurezza e clash-freeness sono questioni cruciali. Vogliamo studiarle usando sia algoritmi crittografici sia tipi. Utilizzeremo la certificazione delle proprietà di sicurezza per ottenere un'implementazione verificata di alcune primitive per la crittografia con curve ellittiche tenendo conto di proprietà di complessità e di performance. Dall'altra parte, le reti di interazione differenziale (DINs) saranno il punto di partenza per la definizione e la codifica di calcoli per la concorrenza. In DINs un deadlock corrisponde ad un conflitto, cioè a due celle collegate sulle rispettive porte principali ma non possono interagire in quanto non è prevista nessuna riduzione per tale caso. Vogliamo individuare un assegnamento di tipo per DINs massimali rispetto alla clash-freeness.

+ Modelli Denotazionali.
Unità coinvolte: I,II,III,IV.
Vogliamo continuare lo studio dei modelli classici per la computazione sequenziale, al fine di esplorare la relazione tra logica e computazione, e definire nuovi strumenti semantici per modellare computazioni parallele e concorrenti (vedi anche il paragrafo 2.3). In questo contesto, intendiamo progettare ambienti formali generali per la costruzione di modelli del lambda-calcolo ed esplorarne le proprietà semantiche globali. Rispetto al primo tema ci vogliamo rifare a due modi classici di definire modelli: quello basato sulla costruzione di domini riflessivi in categorie CCC e quello basato sulle "Extended Abstract Type Structures". La connessione si basa sulla presentazione logica dei lambda-modelli in opportune categorie di ordini parziali. Vogliamo poi studiare proprietà semantiche globali dei lambda-modelli; tra queste, il problema di più ampia portata è quello di determinare se, nella classe dei modelli continui alla Scott, esiste un modello di beta-eta nel quale tutte le funzioni Scott-continue sono rappresentabili.
Passando dal lambda-calcolo a PCF, vogliamo continuare lo studio di StPCF, introdotto da Paolini, che estende PCF con due costrutti, stabili nel senso di Berry. Ci domandiamo se StPCF gode dell'universalità (full completeness) rispetto ai domini stabili ristretti ai loro elementi effettivi.
Gli spazi coerenti spiccano tra le semantiche stabili, in quanto hanno portato alla nascita della logica lineare. La questione dell'universalità per tali spazi è dunque centrale nella semantica denotazionale. Pagani, seguendo in un certo senso l'approccio di Paolini, ha definito un'estensione del frammento esponenziale moltiplicativo della logica lineare (basato sulla aciclicità visibile - una proprietà geometrica delle reti di prova) che osserva nella sintassi la nozione di clique degli spazi coerenti. Intendiamo estendere questi risultati agli spazi ipercoerenti ed agli spazi di finitezza.
Siamo convinti che la semantica denotazionale giocherà un ruolo cruciale nella ricerca in ICC. Forniremo modelli web-based (cioè relazionali o coerenti) delle classi di complessità. Strumento centrale di questo approccio è la nozione di esperimento di Girard, usata da Tortora de Falco e Laurent per lo studio delle classi di complessità.

Veniamo ora al secondo obiettivo:
2. ***Teorie computazionali che modellino l'interazione corretta con l'ambiente.***

Tutte le tecniche linguistiche che abbiamo sin qui menzionato hanno bisogno di una teoria semantica generale delle risorse computazionali, che possa essere essere usata per dimostrare (o verificare) che quelle tecniche linguistiche statiche sono davvero corrette.
Sappiamo che la speranza di delineare una teoria generale unificante è un obiettivo non proponibile sulla base delle conoscenze attuali. Ci proponiamo tuttavia di sviluppare più di una teoria semantica, ognuna delle quali prenda in considerazione un certo fenomeno computazionale, con il fine di di ottenere molteplici mattoni che siano da base per future costruzioni.
I temi che affronteremo sono:

+ Unificazione di sistemi ICC.
Unità coinvolte: I,II
I cosiddetti "sistemi leggeri" hanno per scopo la caratterizzazione di interessanti classi computazionali. Si tratta di sistemi basati su principi assai diversi, che vogliamo confrontare attraverso la definizione di sistemi parametrici. Fissata una classe di complessità C, cercheremo un sistema parametrico Ps tale che le istanze dei suoi parametri possano caratterizzare il maggior numero possibile di sistemi leggeri, relativi a C.

+ Geometria dell'interazione e semantiche per ICC.
Unità coinvolte: II,IV
ICC manca di una fondazione soddisfacente: modelli particolari sono stati ottenuti usando sia gli spazi coerenti, che la semantica dei giochi, la geometria dell'interazione e la semantica dei contesti. Nessuno di questi approcci è davvero incentrato sulla complessità: alcuni si applicano ad una classe troppo limitata di sistemi, altri adottano modelli di costo irragionevoli per la teoria della complessità. Studieremo pertanto ambienti formali più generali, nello stile della geometria dell'interazione e della semantica dei contesti.
Un approccio recente ai modelli "con risorse" dei processi computazionali è basato sulle algebre di von Neumann. In questo contesto il nostro obiettivo è quello di studiare la computazione classica (polinomiale o elementare) in termini del fattore iperfinito, cioè l'unica algebra di von Neumann ce ammette approssimazioni di dimensione finita e che sia anche un fattore.

+ Modelli per la concorrenza e l'interazione.
Unità coinvolte: I,III,IV
Le relazioni tra la logica e la computazione concorrente è stata assai studiata, con l'obiettivo, ancora non raggiunto, di stabilire connessioni forti e feconde tra questi due domini, sul modello delle connessione tra logica e computazione funzionale (corrispondenza di Curry-Howard).
I due approcci principali alla semantica della concorrenza sono i modelli causali e quelli ad interleaving. I primi si concentrano sulle dipendenze causali tra le azioni di un processo, mentre i secondi descrivono un processo come l'insieme di tutti i suoi possibili scheduling. Ci proponiamo di stabilire un contesto basato sulla teoria della dimostrazione nel quale poter definire sia i modelli causali che quelli ad interleaving. Lo strumento principale sarà costituito dalle reti di prova come descrizione dei processi.
Inoltre, intendiamo usare caratteristiche parallele per interpretare calcoli sequenziali, in particolare una versione lineare di PCF. La traduzione di termini PCF in processi del pi-calcolo (che sono una descrizione finitaria dello spazio delle funzioni lineari stabili) può costituire un ponte tra i modelli a giochi e quelli costruiti su domini.
Un modello computazionale collegato sia al lambda-calcolo che alla logica lineare è quello delle reti di interazione. Ci proponiamo di studiare un profondità due sistemi di reti di interazione: i combinatori di interazione e le reti di interazione differenziali (DIN). Nel primo caso, studieremo la costruzione di modelli denotazioni e le loro relazioni con le equivalenze osservazionali, nello stesso spirito di quello che abbiamo proposto per PCF ed il lambda-calcolo.
Ehrhard e Laurent hanno definito una codifica del pi-calcolo nelle DIN. Studieremo in modo semantico le DIN, basandoci sul lavoro di Mazza sulle reti di interazione multiporte e sulla semantica topologica dei combinatori di interazione di Lafont.

+ Nuovi paradigmi computazionali.
Unità coinvolte: II,IV
Alcuni nuovi paradigmi computazionali hanno ricevuto ampio studio, in particolare quello quantistico e quello biologico. Per descrivere la computazione quantistica, svilupperemo un lambda-calcolo quantistico, e progetteremo sistemi di tipo per le classi di complessità quantistiche. Definiremo una nuova logica modale che possa descrivere proprietà dei registri quantistici, in relazione alle due operazioni principali sui dati quantistici: trasformazioni unitarie e misura.
Sul fronte del paradigma biologico, esploreremo la possibilità di rappresentare le reti biologiche con varianti stocastiche della semantica delle reti differenziali, partendo dalla versione stocastica del pi-calcolo introdotta da Priami. Si tratta di un sistema che fornisce una semantica quantitativa al calcolo concorrente sottostante e permette dunque un'applicazione della teoria della concorrenza alla teoria dei sistemi biologici.

+ Fondazioni Logiche.
Unità coinvolte: III

Dal momento che le nostre tecniche sono basate su LL, per progettare nuovi strumenti è essenziale una riflessione fondazionale su di essa. In particolare, deve essere chiarita la connessione tra LL (reti di prova) e le computazioni concorrenti e parallele.
Inizieremo dal frammento moltiplicativo additivo di LL (MALL). Determineremo criteri di correttezza per le reti di prova in MALL che siano stabili per i passi di eliminazione dei tagli "fully local". Questi criteri permetterebbero di estendere la procedura di "focusing" in presenza di additivi. Questo paradigma permetterebbe di fornire una fondazione basata sulla teoria della dimostrazione per i sistemi concorrenti o middleware. Per quanto riguarda la geometria delle dimostrazioni logiche, un approccio interessante è quello costituito dallo studio della topologia delle reti di prova. La LL Permutativa è una recente variante non commutativa della LL. I sequenti permutativi posso rispecchiare la struttura fondamentale di ogni superficie orientata con perimetro. Intendiamo approfondire questo argomento, nella prospettiva di un'interazione tra la logica e la geometria delle varietà in due dimensioni.

Testo inglese
As we said in the point 11, "Obiettivi finali che il Progetto si propone di raggiungere", the researchers involved in this project have a long experience of common work, witnessed by a relevant number of joint publications, also between researchers belonging to different sites of the projects. This depends on the fact that we are interested in the same problems, and can contribute to them starting from different (although similar) backgrounds. Site III is particularly interested and expert in logics, especially Linear Logics and proof-nets. So it will contribute essentially in a deep analysis of the logical foundations of the notions of computation, interaction and resources consumption. Sites I and II collect experiences and interests in the study of the semantics of programming languages, in particular lambda-calculi, and they are interested in building formal tools for proving programs properties, based on logics. People in site IV has a logical and mathematical background, and they proved to be able to use them for building interesting applications. We are sure that all our expertises together will allow us to reach our goals.
In order to put into evidence the collaboration and the role of every site in the project, we will recall the two principal objectives listed in the part OBIETTIVO of this presentation, and for each one we will list the particular goals and the involved sites. For lack of space, the goals will be described in a non exhaustive way, and without references: a complete presentation can be found in the Modello B of the various sites.
Let us start from objective 1:
1. ***Foundational techniques for the analysis and verification of runtime properties of programs.***
The literature offers a wide variety of techniques for proving properties of programs, based on operational, denotational, logical tools. In the last 20 years, starting from the birth of the Linear Logic (LL), the proof theory has concurred to redefine the notion of computation, allowing a different approach, more refined, both to recursion theory and to complexity theory. Moreover, the discovery of the Lights Logics, variants of LL characterizing complexity classes, has provided a formal way to reason about the resources consumption of programs. We want to start from these basis for designing new techniques for the analysis and verification of runtime properties of programs, both exploiting the logical techniques based on LL and Light Logics, and revitalizing existing techniques with the help of the new point of view induced by LL. The problems we want to study inside objective 1 are the following.

+ Optimal Reduction.
Involved sites: I,II,IV
Linear Logic (LL) has found many applications in the study of programming languages. One is in the area of the interpreters using optimal reduction. All known interpreters (e.g., BOHM, Bologna Optimal Higher Order Machine) implement the optimal reduction strategy, realizing Levy's optimality theory, according to a call-by-name strategy. We plan to develop a theory of optimality adapted to other interesting reduction strategies, in particular call-by-value. A useful tool can be the Parametric Lambda-calculus.
A parallel implementation of the optimal reduction for lambda-calculus is PELCR [Parallel Environment for optimal Lambda Calculus Reduction]: we want to revisit it in order to include some control on the complexity of the library's functions. However, still there is no result connecting the amount of work actually performed by any optimal reducer and the cost of reducing the same lambda-term (under some suitable cost model). We plan to define and study reasonable cost models for the lambda-calculus.

+ ICC Systems.
Involved sites: I,II,III
This topic divides naturally in two fields: a foundational study and the application to programming languages. Let us consider the first. Systems developed in accordance with Implicit Computational Complexity (ICC) principles can be designed starting from different traditions: recursive theoretical, proof theoretical, typing/typed-systems theoretical, . We want to formalize the relation between them. A starting point could be the study of two well known systems, both characterizing PTime: Light affine logic (LAL), and Safe Recursion on Notation (SRN). We want to design a new version of LAL by relaxing some of its principles, to obtain a system more clearly related to SRN.
Now let us consider the applications of ICC to programming languages, in particular type assignment systems for paradigmatic languages, where types witness, behind correctness, complexity properties, both in time and space. A fundamental problem is to enlarge the class of expressible programs (as opposed to the simple definability of functions). We will try to do it by enriching the systems by polymorphic, recursive and intersection types. Another problem is to design decidable systems, allowing for a static type checking at compile time, in ML-style. A restriction of the use of polymorphism is a possible approach, provided that such a restriction preserves the completeness of the functional expressivity. Clearly the optimal solution would be to design a system satisfying both the requirements, so enjoying both decidability and a good algorithmic expressivity.
A further problem is to extend ICC systems to significant extensions of lambda-calculi. We plan to extend ICC type assignment systems for usual lambda-calculus to the differential lambda-calculus: the ambitious task is to extend the implicit computational complexity approach to non-deterministic complexity classes, in fact the reduction in the differential lambda-calculus imposes a non-deterministic choice.

+ Security and clash-freeness.
Involved site: IV
In a distributed computational model, both security and clash-freeness are crucial topics. We want to study them using different techniques, the former by designing cryptography algorithms, the latter by using types. For certifying security properties, we will use elliptic-curve based cryptography, with particular care on high performance code (usually written in C language). On the other side, we will start from differential Interaction Nets (DINs) as a starting framework for the definition/encoding of concurrent calculi. In DINs a deadlock corresponds to a clash, namely two cells that faces through their principal port but cannot interact because no reduction rule is defined for such a pair. We want to find a typing assignment for DINs maximal w.r.t. clash-freeness.

+ Denotational Models.
Involved sites: I,II,III,IV.
We want both to continue the study about classical models of sequential computations, in order to further explore the relation between logic and computation, and to define new semantic tools for modeling concurrent and parallel computations (see also paragraph 2.3). In this setting, we want to design general formal frameworks for the construction of models of lambda calculi, and to explore global semantic properties of lambda-models. Concerning the first aim we want to connect the two classical ways of defining models: the one based on building reflexive points in CCC categories, the other based on Extended Abstract Type Structures. The connection is based on logic presentations of lambda models computed in suitable categories of partial orders. Moreover, we plan to study global semantic properties of lambda models; among them, the most important is whether there exists a Scott continuous model of beta-eta where all Scott-continuous functions are representable.
Shifting from lambda-calculus to PCF, we want to continue the study on StPCF, introduced by Paolini, which extends PCF with two stable constructs, in the sense of Berry. We ask whether StPCF has the universality (full completeness) property with respect to the stable domains restricted to their effective elements.
A key stable semantics is that of coherent spaces, which gives rise to linear logic. The question of full completeness with respect to coherent spaces is a central one in denotational semantics. In a spirit similar to Paolini's work, Pagani defines an extension of the multiplicative exponential fragment of linear logic (based on visible acyclicity - a geometric property on proof-nets) which catches in the syntax the notion of clique in coherent spaces. We plan to extend that result to the more recent developments of coherent spaces, such as hypercoherent spaces and finiteness spaces.
We guess that denotational semantics will play a key role in the research on ICC. We will provide web-based models (like the relational and the coherent models) of complexity classes. A central tool of this approach will be the notion of experiment, introduced by Girard and already used by Tortora de Falco and Laurent for the mathematical study of complexity classes.

Let us speak about objective 2:
2. ***Computational theories modeling the correct interaction with the environment.***
All the language-based techniques mentioned so far, call for a general, sound, semantic theory of computational resources, to be used to prove (and to check) that those static language-based techniques are indeed correct. By our theoretical experience we know that the hope of designing a unique general theory is a not realizable goal. So we will develop different semantic theories, each one giving an account of a different computational phenomenon, with the hope to obtain several bricks that could be put together for a further construction. Our particular goals are the following.

+ Unifying ICC systems.
Involved sites: I,II
The class of Light systems contains many systems whose goal, we recall, is to characterize interesting computational classes. These systems are based on different principles, that we want to compare, through the construction of parametric systems. Fixed a computational class C, we shall look for a parametrized system Ps such that instances of its parameters can capture a set, as large as possible, of known Light systems, relative to C.

+ Geometry of Interaction and context semantics of ICC.
Involved sites: II,IV
ICC lacks a satisfactory foundation. Some punctual models have been built, using coherent domains, game semantics, geometry of interactions, context semantics. However, none of them focused on complexity while being applicable to a wide variety of systems. Either they focus on a limited class of systems, or they adopt cost models that are not directly related to computational complexity. We want to study more general frameworks in the style of geometry of interaction and context semantics.
A very recent approach to resource sensitive models of computational processes is based on von Neumann algebras. In this framework, our aim is to gain an understanding of classical (polynomial or elementary) computation in terms of the hyperfinite factor, the unique von Neumann algebra which admits finite dimensional approximations and it is a factor.

+ Modeling concurrency and interaction.
Involved sites: I,III,IV
The relationship between logic and concurrent computation has been for years the subject of intense study. In particular, one would hope to establish a strong and fruitful connection between these two domains, similarly to what happened in the '60s with the Curry-Howard isomorphism.
The two main approaches to a semantic description of concurrency are causal models and interleaving models. The first ones focus on causal dependencies between actions in a process, the second ones describe instead a parallel process as the set of its possible schedulings. We aim to give a proof-theoretical framework for both causal and interleaving models - the key ingredient will be proof-nets, as a description of parallel processes.
Moreover we want to use parallel features for interpreting sequential calculi, using as case study a linear version of the programming language PCF. The translation of the PCF-terms in pi-calculus processes (which are a finitary description of the space of the linear stable functions) can build a bridge between game semantics and domain semantics.
An interesting computational model related to the lambda-calculus and linear logic is that of interaction nets. We plan to make an extensive semantic study of two highly expressive interaction net systems: the interaction combinators and differential interaction nets (DIN). For the first, we shall investigate the construction of denotational models and their relationship with observational equivalences, much in the spirit of what proposed above for the lambda-calculus and PCF.
Recently, Ehrhard and Laurent have defined an encoding of the pi-calculus in DIN. We propose a semantic study of differential interaction nets, based both on Mazza's work on multiport interaction nets, and on a topological semantics developed by the same author for Lafont's interaction combinators.

+ New Computational Paradigms.
Involved site: II,IV
Some new computational paradigms are now under study, in particular quantum and biological paradigm. For modeling quantum computation, we plan to develop a quantum lambda calculi and to design type systems for quantum complexity classes. The key ingredient will be the use of modal systems in order to model properties of quantum registers, with respect to the two main operations on quantum data: unitary transformations and measurement.
As far as the biological paradigm is concerned, we would like to explore the possibility of representing biological networks by a stochastic version of differential net semantics, using as starting point the stochastic version of pi-calculus introduced by Priami. This system provided a quantitative semantics for the underlying concurrent calculus and opened the way to the application of concurrency theory to computational system biology.

+ Logical Foundations.
Involved site: III
Being our techniques essentially based on LL, a foundational investigation on it is essential for building new logical tools. In particular the connections between LL, especially proof-nets, and concurrent and parallel computation need to be clarified, and so a further study of the additive connectives is needed.
We want to start from the multiplicative additive fragment of LL (MALL). We would like to find correctness criteria for MALL proof-nets which are stable under fully local cut reduction steps. These criteria should allow to naturally extend the focusing proof construction paradigm in presence of additives. This paradigm gives a proof-theoretical foundation of concurrent/middle-ware systems. As far as the geometry of logical proofs is concerned, an interesting approach is constituted by studies on the topology of proof-nets. Permutative Linear Logic is a non-commutative variant of LL recently introduced within this specific framework of studies. Permutative sequents are able to reflect the fundamental structure of any oriented surface with boundary. We propose a further development of this subject in the perspective of a dialogue between logic and geometry of 2-dimensional manifolds.

15 - Risultati attesi dalla ricerca, il loro interesse per l'avanzamento della conoscenza e le eventuali potenzialità applicative

Testo italiano
Come abbiamo già illustrato al punto 11, gli obiettivi di questo progetto sono principalmente fondazionali, e di conseguenza ci aspettiamo risultati soprattutto di tipo metodologico, da ottenere mediante il raffinamento e il miglioramento del quadro teoretico ispirato dalla logica lineare sui temi del disegno e dell'analisi delle proprietà a runtime dei programmi. In particolare, vogliamo ampliare l'applicazione delle teorie e dei sistemi di tipo per la misura e il controllo dell'uso di risorse e di sviluppare a partire da queste delle tecniche generali applicabili anche nell'ambito della programmazione concorrente, e, allo stesso tempo, sviluppare nuove teorie e modelli che permettano la specifica, il controllo e la verifica della corretta interazione con l'ambiente (ad esempio, allo scopo di caratterizzare nuove classi di complessità oppure di analizzare nuovi paradigmi computazionali). Vogliamo anche verificare l'applicabilità di alcuni dei risultati ottenuti implementando alcuni prototipi. Descriveremo prima i risultati metodologici che ci attendiamo, seguendo lo schema con cui abbiamo presentato i temi di ricerca al punto 14, "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione". Considereremo prima i risultati attesi nell'ambito
Obiettivo uno:
Tecniche fondazionali per l'analisi e la verifica di proprietà operazionali di programmi.

** Riduzione ottimale.*
- Definizione di una teoria generale per le riduzioni ottimali per varie tipologie di lambda-calcoli.
- Definizione di un modello di costo per il lambda calcolo, che potrebbe essere utilizzato per collegare le differenti misure di complessità determinate da sistemi ICC con la beta riduzione.
- Algoritmi di riduzione che operano nel rispetto della nozione di strategia di riduzione ottimale.
** Sistemi ICC.*
- Una relazione formale chiara tra l'approccio logico e quello ricorsivo alla ICC ottenuta mediante un sistema formale unico dal quale entrambe possono essere ottenute.
- Sistemi di assegnazione di tipo ICC per (estensioni del) lambda calcolo con una buona espressività algoritmica.
- Sistemi di assegnazione di tipo ICC per (estensioni del) lambda calcolo con inferenza di tipo decidibile, con associata la specifica dell'algoritmo di inferenza.
- Sistemi di assegnazione di tipo ICC per calcoli non deterministici, ad esempio per il lambda calcolo differenziale.
** Sicurezza e clash-freeness.*
- Algoritmi per la certificazione di proprietà di sicurezza, applicati al caso della implementazione di primitive crittografiche basata su curve ellittiche.
- Algoritmi di certificazione per la clash-freeness nel caso delle reti di interazione differenziali.
** Modelli semantici.*
- Un ambito generale formale per la costruzione di modelli del lambda calcolo e per il confronto delle proprietà di diversi lambda modelli.
- La prova della proprietà di universalità di StPCF.
- Un modello ICC dei processi computazionali, ottenuto nel quadro delle algebre di von Neumann.

I risultati attesi nell'ambito dell'obiettivo 2:

Teorie computazionali che modellino l'interazione corretta con l'ambiente
sono i seguenti.

** Unificazione dei sistemi ICC*
- Un sistema parametrico per ICC che generi sistemi costruiti su principi diversi, all'interno di una data classe di complessità.
- Un sistema parametrico per ICC dove la classe di complessità è determinata dalla scelta dei parametri.
** Semantica per i sistemi ICC*
- Un ambito generale nello stile della geometria dell'interazione e della semantica a contesti per ottenere una semantica dei sistemi ICC.
- Modelli denotazionali di ICC utilizzando i tipi intersezione.
** Modelli per la concorrenza e l'interazione.*
- Un ambito proof-teoretico che catturi i diversi approcci alla concorrenza, sia quello causale che quello a interleaving.
- Modelli dei combinatori ad interazione e delle reti di prova differenziali.
** Nuovi paradigmi computazionali.*
- Un lambda-calcolo quantistico e sistemi di assegnazione di tipo per classi di complessità quantistiche.
- Un modello delle reti biologiche a partire varianti stocastiche della semantica delle reti differenziali.
** Fondamenti Logici.*
- Studio di criteri di correttezza per i proof-net MALL che sono stabili rispetto alla riduzione del taglio contando i passi di computazione.
- Studio di caratterizzazioni geometriche di sistemi non-commutativi, derivanti dalla logica lineare, come nel caso della Permutative Linear Logic (PLL).

L'orizzonte applicativo di questo progetto deriva direttamente dai risultati che speriamo di ottenere sulla natura dei linguaggi di programmazione, e che sono sopra elencati. Come abbiamo detto l'indagine di tipo fondazionale mirerà a realizzare specifiche per il controllo, la certificazione, la specifica di proprietà a runtime. L'applicazione di questi risultati sarà effettuata sulle nuove versioni che verranno rilasciate e su prototipi sviluppati espressamente nell'ambito del calcolo distribuito, del calcolo concorrente applicato alla biologia dei sistemi, della certificazione di proprietà di sicurezza. I nostri lavori sul raffinamento dei sistemi di assegnazione di tipo per la ICC ci mettono nella posizione migliore per poter definire delle discipline di tipo che con un maggiore potere di risoluzione permettano trattare i dettagli delle questioni legate alla ICC nei linguaggi di programmazione. La caratterizzazione delle classi di complessità mediante l'inquadramento in differenti aree della logica e in diversi linguaggi, con i contributi a diverse aree di questa disciplina, aumenta i gradi di libertà nel momento in cui cercheremo di applicare i nuovi concetti provenienti dalla ICC alla esecuzione dei programmi attraverso la riduzione ottimale.

Vogliamo realizzare:

** nell'ambito del calcolo distribuito (a partire dai risultati ottenuti in 1.1, 2.1):*
- una nuova release di PELCR con un algoritmo di inferenza di tipo per sistemi ICC;
- certificazione di complessità e implementazione di un algoritmo di distribuzione del carico che tenga conto dei certificati;
- l'interfacciamento di PELCR con linguaggi esterni;
** nell'ambito della sicurezza (a partire dai risultati ottenuti in 1.3., 1.2):*
- l'implementazione di una libreria (dimostrabilmente sicura) che implementi alcune primitive di crittografia su curve ellittiche;
- una certificazione dell'assenza di overflow, basata su metodi di analisi statica;
** nell'ambito della biologia dei sistemi (a partire dai risultati ottenuti in 2.4):*
- un sistema di simulazione stocastica delle interazioni micromolecolari basato sulla semantica quantitativa per le reti di interazione differenziali.
** nell'ambito della programmazione funzionale (a partire dai risultati ottenuti in 1.2, 2.4)*
- implementazioni di sistemi di inferenza di tipo per vari lambda calcoli

L'interesse pratico dei risultati che ci aspettiamo di ottenere è dato dalla possibilità di applicazione al vasto dominio delle architetture distribuite, come dove è necessario produrre certificati che accompagnino la distribuzione di oggetti, e nello sviluppo di tools per i sistemi contenuti in dispositivi mobili dove i certificati hanno un'importanza principalmente dettata dalla necessità di una certificazione del software rispetto alla disponibilità limitata di risorse.

Testo inglese
As we said in the part 11, our aim is mainly foundational, so the main outcome that we expect is the realization of methodological tools. We will obtain it by a refinement and improvement of the theoretical framework inspired by linear logic for the design and analysis of run-time properties of programs. In particular, we expect to widen the application of the theories and typing systems for the measurement and control of the consumption of resources and to develop general techniques applicable to concurrent programs also; but at the same time, we expect to develop new theories and models allowing to specify, control and verify the correct interaction with the environment (for instance, in order to characterize new complexity classes or to analyze new computational paradigms). Moreover, we would like to test the applicability of some of our results by some prototypical implementation. First we will describe the methodological results, using the same classification we used at point 14, "Ruolo di ciascuna unità operativa in funzione degli obiettivi previsti e relative modalità di integrazione e collaborazione". Let us consider first the objective 1:
Foundational techniques for the analysis and verification of runtime properties of programs.

** Optimal Reduction.*
- Definition of a general theory of optimal reductions for various kind of lambda calculi.
- Definition of a cost model for lambda calculus, that could be used for relating the complexity measure of ICC systems with the beta reduction.
- Reduction algorithms according to optimal reduction strategy.
** ICC Systems.*
- A clear formal relation between the logical and the recursion theoretical approach to ICC through a formal system from which both can be derived.
- ICC type assignment systems for (extensions of) lambda calculus with a good algorithmic expressivity.
- ICC type assignment systems for (extensions of) lambda calculus with decidable type inference algorithms, with a design of the type inference algorithm.
- ICC type assignment systems for non deterministic calculi, e.g. differential lambda calculus.
** Security and clash-freeness.*
- Algorithms certifying security properties, using elliptic-curve based cryptography.
- Algorithms certifying clash-freeness for differential Interaction Nets.
** Semantic Models.*
- A general formal framework for the construction of models of lambda calculi, and for comparing properties of different lambda models.
- A proof of the universality property of StPCF.
- An ICC model of computational processes, through a von Neumann algebras framework.

The expected results inside objective 2:
Computational theories modeling the correct interaction with the environment
are the following.

** Unifying ICC systems.*
- A parametric ICC system, where the complexity classes are determined by different parameter choices.
- Given a complexity class, a parametric ICC system, from which different ICC principles can be caught by different parameter choices.
** Semantics of ICC.*
- A general frameworks in the style of geometry of interactions and context semantics for the ICC semantics.
- Denotational models of ICC using intersection types.
** Modeling concurrency and interaction.*
- A proof-theoretical framework for both causal and interleaving models of concurrency.
- An algebraic model of interaction and concurrency, based on differential proof nets.
** New Computational Paradigms.*
- A quantum lambda calculus language, and type assignment systems for quantum complexity classes.
- A stochastic version of the differential nets semantics.
** Logical Foundations.*

- Correctness criteria for MALL proof-nets which are stable under fully local cut reduction steps.
- Purely geometric characterization of non-commutative formal systems, obtained from linear logic; for instance, it is the case with Permutative Linear Logic (PLL).

Applicative horizon of this proposal directly comes from theoretical results around the nature of programming languages we could obtain with our work. Since the investigation on foundations has as a target the description of specifications for control, certification of runtime properties, we expect to apply these results on new software versions of our prototypes developed in the areas of distributed computing, concurrent systems for computational systems biology, certification of security properties of software.

Refinement of type assignment systems puts ourselves in position to provide in turn fine-grained type disciplines dealing with ICC issues of programming languages. The characterization of complexity classes by different logical frameworks and language settings add flexibility to optimal reduction especially when oriented to issues concerning complexity.

We aim to implement:

** in the distributed computing setting (starting from the results obtained in 1.1, 2.1):*
- a new release of PELCR including a type inference algorithm for ICC;
- a complexity properties certification and implementation of a load balancing of tasks taking in account these certificates;
- an interface between PELCR and foreign languages and libraries;
** in the software security setting (starting from the results obtained in 1.3,1.2):*
- a certification of no buffer overflow by static analysis;
- a provably secure library which implements some primitives of elliptic curves cryptography;
** in the systems biology setting (starting from the results obtained in 2.4):*
- stochastic simulation software based on the quantitative semantics we plan to provide to differential interaction nets.
** in the functional programming setting (starting from the results obtained in 1.2, 2.4):*
- type assignment systems for various king of lambda-calculi:

Practical interest on this kind of results is given the possibility of application to the broad domain of distributed architectures, where it is customary to produce certificates accompanying broadcasting of objects and in developing of tools for embedded mobile systems where certificates concern the use of limited resources.

16 - Elementi e criteri proposti per la verifica dei risultati raggiunti

Testo italiano
Proponiamo, per la valutazione dei risultati teorici di questo progetto, i seguenti criteri:

+ Le pubblicazioni prodotte durante il progetto saranno valutate sulla base all'impact factor e della pertinenza del giornale o dei proceeding delle conferenze su cui saranno pubblicati.
+ A ogni convegno del progetto inviteremo referee esterni, a cui sarà chiesto di scrivere un rapporto sulla qualità dei risultati prodotti. Questi rapporti saranno allegati al rendiconto finale del progetto.
+ Pubblicheremo i proceeding del convegno finale. Le modalità della pubblicazione e il successo del volume saranno una misura dei risultati ottenuti.

Riguardo ai risultati applicativi, tutti i prodotti software sviluppati all'interno del progetto saranno pubblicati sul web, per esser liberamente usati dalla comunità scientifica. A tutti gli utenti sarà chiesto un rapporto sulla loro esperienza.

Poichè la formazione di nuovi ricercatori è uno degli scopi più importanti del progetto, noi chiederemo a ogni persona che avrà usufruito di un assegno di ricerca pagato sui fondi del progetto di scrivere una relazione finale, dove dovranno riportare sia i risultati ottenuti che un giudizio sulla loro esperienza di lavoro. Queste relazioni saranno pubbliche, e potranno costituire una misura ulteriore del successo del progetto, relativamente a questo particolare scopo.

Testo inglese
We propose, for the evaluation of the theoretical results of this project, the following criteria:

+ The publications developed during the project will be evaluated with respect to the impact factor and pertinence of either the journal or the conference proceedings, when they will appear.
+ We will invite, at every meeting of the project, external referees, which will be asked to write reports on the quality of our scientific research. Such reports will be attached to the final one.
+ The proceedings of the final meeting will be published. The modalities of the publication and the success of the volume will be a measure of the result of the project.

As far as the implementations that will be realized, all the software developed inside the project will be put in the web, for be freely used by the scientific community. To all users a report on their experience will be asked.

Since the formation of new researchers is one of the most important goal of this project, we will ask to every person with a grant payed by the project to write a final relation, where both the obtained results and a judgment on this research experience. These relations will be put in the web, and could be used as measure of the success of the project, with respect to this particular goal.

17 - Mesi persona complessivi dedicati al Progetto di Ricerca

		Numero	Impegno 1° anno	Impegno 2° anno	Totale mesi persona
Componenti della sede dell'Unità di Ricerca		*8*	*60*	*55*	*115*
Componenti di altre Università /Enti vigilati		*10*	*67*	*65*	*132*
Titolari di assegni di ricerca		*0*			
Titolari di borse	*Dottorato*	*8*	*33*	*30*	*63*
	Post-dottorato	*0*			
	Scuola di Specializzazione	*0*			
Personale a contratto	*Assegnisti*	*4*	*1*	*34*	*35*
	Borsisti	*0*			
	Altre tipologie	*0*			
Dottorati a carico del PRIN da destinare a questo specifico progetto		*0*	*0*	*0*	*0*

Altro personale		*3*	*14*	*12*	*26*
TOTALE		**33**	**175**	**196**	**371**

18 - Costo complessivo del Progetto articolato per voci

Voce di spesa	Unità I	Unità II	Unità III	Unità IV	TOTALE
Materiale inventariabile	*5.000*	*5.000*	*0*	*1.900*	11.900
Grandi Attrezzature	*0*	*0*	*0*	*0*	0
Materiale di consumo e funzionamento	*6.000*	*7.000*	*2.500*	*5.100*	20.600
Spese per calcolo ed elaborazione dati	*0*	*0*	*0*	*100*	100
Personale a contratto	*19.000*	*19.000*	*18.100*	*19.000*	75.100
Dottorati a carico del PRIN da destinare a questo specifico progetto	*0*	*0*	*0*	*0*	0
Servizi esterni	*0*	*0*	*0*	*100*	100
Missioni	*12.500*	*15.000*	*14.500*	*7.100*	49.100
Pubblicazioni	*0*	*0*	*0*	*500*	500
Partecipazione / Organizzazione convegni	*3.000*	*10.000*	*3.500*	*12.600*	29.100
Altro	*5.000*	*3.500*	*3.000*	*4.000*	15.500
TOTALE	**50.500**	**59.500**	**41.600**	**50.400**	**202.000**

19 - Prospetto finanziario suddiviso per Unità di Ricerca

	Unità I	Unità II	Unità III	Unità IV	TOTALE
a.1) finanziamenti diretti, disponibili da parte di Università/Enti vigilati di appartenenza dei ricercatori dell'unità operativa	*12.400*	*0*	*0*	*2.000*	14.400
a.2) finanziamenti diretti acquisibili con certezza da parte di Università/Enti vigilati di appartenenza dei ricercatori dell'unità operativa	*4.500*	*17.900*	*12.500*	*13.200*	48.100
b.1) finanziamenti diretti disponibili messi a disposizione da parte di soggetti esterni	*0*	*0*	*0*	*0*	0
b.2) finanziamenti diretti acquisibili con certezza, messi a disposizione da parte di soggetti esterni	*0*	*0*	*0*	*0*	0
c) cofinanziamento richiesto al MUR	*33.600*	*41.600*	*29.100*	*35.200*	139.500
TOTALE	**50.500**	**59.500**	**41.600**	**50.400**	**202.000**

(per la copia da depositare presso l'Ateneo e per l'assenso alla diffusione via Internet delle informazioni riguardanti i programmi finanziati e la loro elaborazione necessaria alle valutazioni; D. Lgs, 196 del 30.6.2003 sulla "Tutela dei dati personali")

Firma _____ Data 31/10/2007 ore 13:07