PROGETTO DI UNA UNITÀ DI RICERCA - MODELLO B
Anno 2005 - prot. 2005015491_004

PARTE I

1.1 Programma di Ricerca afferente a

Area Scientifico Disciplinare *01: Scienze matematiche e informatiche 100%*

1.2 Durata del Programma di Ricerca

24 Mesi

1.3 Coordinatore Scientifico del Programma di Ricerca

ROSSI ***FRANCESCA*** ***frossi@math.unipd.it***

INF/01 - Informatica

Università degli Studi di PADOVA

Facoltà di SCIENZE MATEMATICHE FISICHE e NATURALI

Dipartimento di MATEMATICA PURA ED APPLICATA

1.4 Responsabile Scientifico dell'Unità di Ricerca

BISTARELLI ***STEFANO***

Ricercatore non confermato ***23/06/1968*** ***BSTSFN68H23C745F***

ING-INF/05 - Sistemi di elaborazione delle informazioni

Università degli Studi "G. d'Annunzio" CHIETI-PESCARA

Facoltà di ECONOMIA

Dipartimento di SCIENZE

+39 3488260770	***+39 0854549755***	***bista@sci.unich.it***
(Prefisso e telefono)	*(Numero fax)*	*(Indirizzo posta elettronica)*

1.5 Curriculum scientifico del Responsabile Scientifico dell'Unità di Ricerca

Stefano Bistarelli,
Laureato con lode in Scienze dell'Informazione presso l'università di Pisa nell'aprile 1994.
Dal 1994 al 1996 ha avuto contratti di collaborazione alla ricerca con il dipartimento di Informatica di Pisa.
Nell'ottobre 1996 vince il concorso per l'ammissione al dottorato di ricerca in informatica presso lo stesso dipartimento. Nell'ambito del corso di dottorato partecipa a numerose scuole:
SNDIS97,98 (Bertinoro),
Distributed Systems and Security 98,
E-Commerce and On-line Algorithms 2000 (Lipari),
Compulog CLP School 99 (LasCruces USA),
Esprit school on Constraints in Computational Logic 99 (Parigi)

Nell' aprile 2001 consegue il titolo di dottore di Ricerca con la tesi "Soft Constraint Solving and programming" che vince due importanti premi dell'informatica italiana:
- Migliore tesi in Informatica teorica (premio assegnato dal capitolo italiano della European Association of Theoretical Computer Science (EATCS)), e
*- Migliore Neo Dottore di Ricerca In Intelligenza Artificiale (premio assegnato dall'Associazione Italiana per l'Intelligenza Artificiale (AI*IA))*

Nel Giugno 2001 vince un assegno di ricerca presso l'Università di Padova.
Nel Settembre 2001 vince un assegno di Ricerca presso l'istituto di Informatica e telematica (IIT) del CNR a Pisa.
Nel Settembre 2002 diventa ricercatore presso il dipartimento di Scienze, Universita' di Chieti-Pescara.
Dal dicembre 2002 è anche collaboratore scientifico dell'IIT - CNR a Pisa. Il CNR lo propone per il Cor Baayen award e viene nominato rappresentante per l'Italia. Il premio annuale è dato al giovane ricercatore più promettente nel campo dell'Informatica e della Matematica.

Ha svolto attività didattica dal 2000 ad oggi prima presso l'Universita' di Pisa e attualmente presso l'Università degli Studi "G. D'Annunzio" di Chieti-Pescara.
Per la sua attività di ricerca in campo "Constraint Solving and Programming" e in campo "Security" ha avuto e continua ad avere ampissima collaborazione internazionale:
- invited talks e visite presso università e centri di ricerca (INRIA Parigi; IC-Park Imperial College Londra; Dipartimento di Linguaggi e Sistemi Informatici Barcellona; Istituto di Logica, Linguaggi e Computazione Amsterdam; Istituto di Informatica LMU Monaco; S.R.I. San Francisco; Chinese University of HongKong, Cork Constraint Computational Centre e Dipartimento di Computer Science, Cork, Irlanda;
- Organizzazione di Conferenze e Workshop Internazionali (Publicity Chair a CP98 (Pisa); Chair della AI and Computational Logic track al SAC (Symposyum in Applied Computing) della ACM negli anni 2002, 2003e 2004; PC member dell'Appia-Gulp-Prode Conference 2002 (Madrid); Chair del Workshop on Soft Constraint negli anni 2002, 2003 e 2004, PC member del RCORP workshop nel 2002; Chair del Constraint Track a FLAIR negli anni 2003 e 2004, pc member dello IADIS International Conference on e-commerce (2004). Membro inoltre dello IERB dell'International Journal of Mobile Computing and E-Commerce.

Nel suo curriculum vanta decine di pubblicazione su conference (CP, IJCAI, PADL, ETAPS, SAC, FLAIR, Ercim constraint workshops,) e journal internazionali (JACM, Constraint, AI, Toplas, TOCL, JCS, TPLP,).
La sua tesi estesa con gli ultimi è stata pubblicata come Libro dalla Springer.

1.6 Pubblicazioni scientifiche più significative del Responsabile Scientifico dell'Unità di Ricerca

1. *GIAMPAOLO BELLA, STEFANO BISTARELLI (in stampa). Information Assurance for Security Protocols. COMPUTERS & SECURITY. ISSN: 0167-4048 in stampa.*
2. *STEFANO BISTARELLI, UGO MONTANARI, FRANCESCA ROSSI (in stampa). Soft Concurrent Constraint Programming. ACM TRANSACTIONS ON COMPUTATIONAL LOGIC. ISSN: 1529-3785 in stampa.*
3. *GIAMPAOLO BELLA, STEFANO BISTARELLI, SIMON N. FOLEY (in stampa). Soft Constraints for Security. First International Workshop on Views On Designing Complex Architectures (VODCA 2004). To appear.*
4. *GIAMPAOLO BELLA, STEFANO BISTARELLI (in stampa). A protocol's life after attacks... 11th International Workshop on Security Protocols. in stampa.*
5. *STEFANO BISTARELLI, SIMON N. FOLEY, BARRY OSULLIVAN (in stampa). Reasoning about Secure Interoperation using Soft Constraints. IFIP TC1 WG1.7 Workshop on Formal Aspects in Security and Trust (FAST). (vol. 173). Kluwer.*
6. *GIAMPAOLO BELLA, STEFANO BISTARELLI (in stampa). Biometrics to Enhance Smartcard Security (Simulating MOC using TOC). 11th International Workshop on Security Protocols. in stampa.*
7. *BISTARELLI S., ILIANO CERVESATO, GABRIELE LENZINI AND FABIO MARTINELLI (2005). Relating Multiset Rewriting and Process Algebras for Security Protocol Analysis. JOURNAL OF COMPUTER SECURITY. vol. 13 ISSN: 0926-227X*
8. *BISTARELLI S. (2004). Semirings for Soft Constraint Solving and Programming. (vol. 2962). ISBN: -540-21181-0 Lecture Notes in Computer Science.: Springer*
9. *BISTARELLI S., THOM FRWIRTH, MICHAEL MARTE, AND FRANCESCA ROSSI (2004). Soft Constraint Propagation and Solving in Constraint Handling Rules. COMPUTATIONAL INTELLIGENCE. vol. 20 pp. 287-307 ISSN: 0824-7935*
10. *GIAMPAOLO BELLA, STEFANO BISTARELLI (2004). Advancing Assurance for Secure Distributed Communications. 5th Annual IEEE Information Assurance Workshop, "The West Point Workshop". (pp. 306-313). IEEE.*
11. *STEFANO BISTARELLI, SIMON N. FOLEY, BARRY O'SULLIVAN (2004). Detecting and Eliminating the Cascade Vulnerability Problem from Multi-level Security Networks using Soft Constraints. Innovative Applications of Artificial Intelligence Conference (IAAI-04). AAAI Press.*
12. *STEFANO BISTARELLI, SIMON N. FOLEY, BARRY O'SULLIVAN (2004). Modelling and Detecting the Cascade Vulnerabiliy Problem using Soft Constraints. ACM Symposium on Applied Computing (SAC 2004). (pp. 383-390). ACM Press.*
13.

GIAMPAOLO BELLA, BISTARELLI S. (2004). Soft Constraint Programming to Analysing Security Protocol. JOURNAL OF COMPUTER SECURITY. vol. 4 pp. 1-28 ISSN: 0926-227X

14. *BISTARELLI S., ROSSELLA GENNARI AND FRANCESCA ROSSI (2003). General Properties and Termination Conditions for Soft Constraint Propagation. CONSTRAINTS. vol. 8 pp. 79-97 ISSN: 1383-7133*
15. *STEFANO BISTARELLI, ILIANO CERVESATO, GABRIELE LENZINI, FABIO MARTINELLI (2003). Relating Process Algebras and Multiset Rewriting for Immediate Decryption Protocols. Second International Workshop on Mathematical Methods, Models and Architectures for Computer Network. (vol. 2776 pp. 88-101). Springer, Lecture Notes in Artificial Intelligence.*
16. *STEFANO BISTARELLI, FRANCESCA ROSSI, ISABELLA PILAN (2003). Abstracting Soft Constraints: Some experimental results on Fuzzy CSPs. Recent Advances in Constraints, Joint ERCIM/CoLogNET International Workshop on CSP. (vol. 3010 pp. 107-123). Springer, Lecture Notes in Computer Science.*
17. *STEFANO BISTARELLI, SIMON N. FOLEY (2003). Analysis of Integrity Policies using Soft Constraints. IEEE 4th International Workshop on Policies for Distributed Systems and Networks (POLICY2003). (pp. 77-80). IEEE.*
18. *STEFANO BISTARELLI, SIMON N. FOLEY (2003). A Constraint Based Framework for Dependability Goals: Integrity. 22nd International Conference on Computer Safety, Reliability and Security (SAFECOMP2003). (pp. 130-143). Springer, Lecture Notes in Computer Science.*
19. *BISTARELLI S., PHILIPPE CODOGNET AND FRANCESCA ROSSI (2002). Abstracting soft constraints: Framework, properties, examples. ARTIFICIAL INTELLIGENCE. vol. 139 pp. 175-211 ISSN: 0004-3702*
20. *BISTARELLI S., UGO MONTANARI AND FRANCESCA ROSSI (2002). Soft Constraint logic Programming and Generalized Shortest Path Problems. JOURNAL OF HEURISTICS. vol. 8 pp. 25-41 ISSN: 1381-1231*
21. *STEFANO BISTARELLI, UGO MONTANARI, FRANCESCA ROSSI (2002). Soft Concurrent Constraint Programming. Programming Languages and Systems: 11th European Symposium on Programming, ESOP 2002 Part of ETAPS. (vol. 2305 pp. 53-67). Springer, Lecture Notes in Computer Science.*
22. *GIAMPAOLO BELLA, STEFANO BISTARELLI (2002). Confidentiality levels and deliberate/indeliberate protocol attacks. Security Protocols: 10th International Workshop, Revised Papers. (vol. 2845 pp. 104-119). Springer, Lecture Notes in Computer Science.*
23. *BISTARELLI S., UGO MONTANARI AND FRANCESCA ROSSI (2001). Semiring-Based Constraint Logic Programming: Syntax and Semantics. ACM TRANSACTIONS ON PROGRAMMING LANGUAGES AND SYSTEMS. vol. 23 pp. 1-29 ISSN: 0164-0925*
24. *GIAMPAOLO BELLA, STEFANO BISTARELLI (2001). Soft Constraints for Security Protocol Analysis: Confidentiality. Practical Aspects of Declarative Languages: Third International Symposium, PADL. (vol. 1990 pp. 108-122). Springer, Lecture Notes in Computer Science.*
25. *STEFANO BISTARELLI, PHILIPPE CODOGNET, FRANCESCA ROSSI (2000). An Abstraction Framework for Soft Constraint and Its Relationship with Constraint Propagation. Abstraction, Reformulation, and Approximation: 4th International Symposium, SARA. (vol. 1864 pp. 71-86). Springer, Lecture Notes in Artificial Intelligence.*
26. *STEFANO BISTARELLI, PHILIPPE CODOGNET, YAN GEORGET, FRANCESCA ROSSI (2000). Abstracting Soft Constraints. New Trends in Constraints: Joint ERCIM/Compulog Net Workshop. Selected Papers,. (vol. 1865 pp. 108-133). Springer, Lecture Notes in Artificial Intelligence.*
27. *BISTARELLI S., HELENE FARGIER, UGO MONTANARI, FRANCESCA ROSSI, THOMAS SCHIEX AND GERARD VERFAILLIE (1999). Semiring-Based CSPs and Valued CSPs: Frameworks, Properties, and Comparison. CONSTRAINTS. vol. 4 pp. 199-240 ISSN: 1383-7133*
28. *BISTARELLI S., UGO MONTANARI AND FRANCESCA ROSSI (1997). Semiring-based Constraint Solving and Optimization. JOURNAL OF THE ASSOCIATION FOR COMPUTING MACHINERY. vol. 44 pp. 201-236 ISSN: 0004-5411*
29. *STEFANO BISTARELLI, UGO MONTANARI, FRANCESCA ROSSI (1997). Semiring-based Constraint Logic Programming. 15th International Joint Conference on Artificial Intelligence (IJCAI97),. (pp. 352-357).*
30. *STEFANO BISTARELLI, UGO MONTANARI, FRANCESCA ROSSI (1995). Constraint Solving over Semirings. 14th International Joint Conference on Artificial Intelligence (IJCAI95),. (pp. 624-630).*

1.7 Risorse umane impegnabili nel Programma dell'Unità di Ricerca

1.7.1 Personale universitario dell'Università sede dell'Unità di Ricerca

Personale docente

nº	Cognome	Nome	Dipartimento	Qualifica	Settore Disc.	Mesi Uomo

						1° anno	2° anno
1.	*BISTARELLI*	*Stefano*	*Dip. SCIENZE*	*Ricercatore Universitario*	*ING-INF/05*	*7*	*5*
2.	*MEO*	*Maria Chiara*	*Dip. SCIENZE*	*Prof. Associato*	*INF/01*	*5*	*5*
3.	*AMATO*	*Gianluca*	*Dip. SCIENZE*	*Ricercatore Universitario*	*INF/01*	*5*	*5*
4.	*FIORAVANTI*	*Fabio*	*Dip. SCIENZE*	*Ricercatore Universitario*	*INF/01*	*6*	*6*
	TOTALE					**23**	**21**

Altro personale

Nessuno

1.7.2 Personale universitario di altre Università

Personale docente

Nessuno

Altro personale

Nessuno

1.7.3 Titolari di assegni di ricerca

Nessuno

1.7.4 Titolari di borse

nº	Cognome	Nome	Dipartimento	Anno di inizio borsa	Durata (in anni)	Tipologia	Mesi Uomo 1° anno	Mesi Uomo 2° anno
1.	*Gubiani*	*Donatella*	*Dip. SCIENZE*	*2005*	*3*	*Dottorato*	*7*	*5*
	TOTALE						**7**	**5**

1.7.5 Personale a contratto da destinare a questo specifico programma

Nessuno

1.7.6 Personale extrauniversitario indipendente o dipendente da altri Enti

Nessuno

PARTE II

2.1 Titolo specifico del programma svolto dall'Unità di Ricerca

Tecniche di Astrazione, Concorrenza e Vincoli Soft per Sicurezza Informatica e studio si Sistemi Informatici

2.2 Settori scientifico-disciplinari interessati dal Programma di Ricerca

INF/01 - Informatica

2.3 Parole chiave

RISOLUZIONE DI VINCOLI ; VINCOLI SOFT ; PROGRAMMAZIONE LOGICA CON VINCOLI ; ASTRAZIONE DI VINCOLI ; PROGRAMMAZIONE CONCORRENTE CON VINCOLI ; VINCOLI DI SICUREZZA ; PROTOCOLLI DI SICUREZZA ; VALUTAZIONE DEL RISCHIO DI SICUREZZA

2.4 Base di partenza scientifica nazionale o internazionale

La programmazione con vincoli e le tecniche di soddisfacimento di vincoli stanno emergendo come una promettente metodologia formale per l'analisi di sistemi ed applicazioni. La principale motivazione per tale successo sembra essere legata alla capacità del framework basato su vincoli di facilmente adattarsi a svariate tecnologie e applicazioni. Nel seguito di questo documento descriveremo i punti chiave della nostra ricerca basata sulla nozione di vincolo.

I vincoli Soft

Le tecniche di soddisfacimento di Vincoli e la programmazione con vincoli hanno mostrato di essere semplici ma potenti idee. Un vincolo è semplicemente una restrizione sulla combinazione di valori permessi per un insieme di variabili. Se riusciamo a formulare il nostro problema come un insieme di vincoli, e abbiamo algoritmi e metodologie per soddisfare tali vincoli otteniamo la soluzione del nostro problema. L'idea e' molto generale perchè può essere applicata a varie classi di vincoli e di algoritmi di risoluzione. In più, è un formalismo potente perchè generale, dichiarativo e applicabile ad ampi casi reali.
Negli ultimi anni, comunque, questa semplice nozione di vincolo ha mostrato alcune quando il problema da risolvere era troppo vincolato o in caso di presenza di preferenze.
Se il problema e' troppo vincolato e non ci sono soluzioni, le tecnologie classiche non possono dare alcuna risposta. In maniera simile se più soluzioni sono presenti non esiste un modo per discriminare in base ad una nozione di preferenza.
La non possibilità di non poter gestire problemi troppo vincolati o preferenze non e' un problema solo teorico ma anche applicativo. Per questa ragione molti studiosi in questo ambito hanno studiato formalismi ad hoc per indirizzare queste necessità. Questo ha portato alla nozione dei cosiddetti "vincoli soft". Dopo vari sforzi per definire specifiche classi di vincoli soft, come i fuzzy, parziali, e gerarchici, e' apparsa evidente la necessità di un trattamento generale dei vincoli soft capace di modellare le differenti classi in maniera uniforme e capace di provare proprietà in maniera più generale.
Più precisamente, un vincolo soft può essere visto come un vincolo dove ogni istanza delle sue variabili ha associato un valore di un insieme parzialmente ordinato interpretato come un livello di preferenza. La combinazione di più vincoli insieme dovrà quindi tenere di conto questi valori di preferenza assegnati alle istanze dei vincoli e fornire appositi operatori di un semiring per la combinazione di valori (x) e per il loro confronto (+).

Programmazione concorrente con vincoli (temporizzata) (t)ccp

Molti programmi applicativi della vita reale coinvolgono aspetti critici rispetto al tempo. Caratteristiche di tali applicazioni,usualmente dette sistemi integrati real-time, sono la specifica di vincoli temporali, come per esempio, la necessità di esprimere limiti superiori sul tempo di realizzazione di un evento (timeouts).
Al fine di modellare tali situazioni un linguaggio deve permettere di specificare che in caso di timeout dovrà intraprendere un'azione alternativa.
Sono stati sviluppati diversi formalismi per specificare, verificare e programmare sistemi reattivi, incluse le algebre di processo temporizzate [39, 40, 42, 46], le logiche temporali [47, 45] e i linguaggi concorrenti sincroni ESTEREL [41] e LUSTRE [44].
L'assunzione alla base del loro modello computazionale è l'ipotesi di reazione istantanea o di sincronia perfetta: un programma attivato da segnali di input e reagisce istantaneamente producendo l'output richiesto. In [48, 44] sono state definite delle estensioni del ccp sotto l'ipotesi di sincronia limitata (come introdotta in [48]): la computazione prende un periodo di tempo limitato (piuttosto che essere istantanea, come accade in ESTEREL) e l'intero sistema evolve in cicli corrispondenti ad unità di tempo: in ogni fase viene eseguito un processo ccp per produrre un responso ad un input fornito dall'ambiente. Il processo accumula informazioni in modo monotono nello stato, in accordo al modello computazionale standard del ccp.
Mentre il linguaggio definito in [48] è un linguaggio deterministico, ispirato ai linguaggi sincroni ed utile soprattutto per la programmazione di piccoli nuclei real-time, il timed ccp, definito in [44] include il non-determinismo ed più adatto soprattutto per specificare grandi sistemi che coinvolgono, diversi processi, che magari sono eseguiti su diversi processori e che comunicano per mezzo di collegamenti asincroni.
In [44] sono anche introdotte le sequenze reattive temporizzate, che descrivono ad ogni istante di tempo la reazione di un processo tccp all'input fornitogli dall'ambiente esterno. Formalmente una tale reazione è una coppia di vincoli (c,d), dove c l'input dato dall'ambiente e d il vincolo prodotto dal processo in risposta all'input c.

L'uso di tali sequenze permette di modellare ad ogni istante il comportamento di input-output di programmi tccp. Questo si riflette nella logica definita in [49].

Analisi statica e interpretazione astratta

Con il termine "analisi statica" si intende l'insieme delle tecniche per cui, dato un programma P, si ricavano delle proprietà sulla sua futura esecuzione già a tempo di compilazione. Un esempio classico di analisi da compiere su di un programma è quello di determinare gli input per cui esso termina. Purtroppo, molte proprietà sono in decidibili e l'analisi può spesso soltanto fornire risposte parziali.
L'importante è che tali risposte siano irrefutabili: è accettabile rispondere "non so", ma se la risposta è "si" allora questo deve avvenire veramente.
L'attività di ricerca collegata all'analisi statica varia molto e coinvolge diversi settori dell'informatica, sia in ambito applicativo che teorico. Questa è una situazione molto interessante: la teoria è decisamente orientata verso le applicazioni pratiche e la pratica trae quasi immediato beneficio dalla teoria.
L'Interpretazione Astratta un framework semantico per l'analisi statica di sistemi, introdotto da Patrick e Radhia Cousot in [29].
L'idea dell'interpretazione astratta quella di calcolare la semantica formale di un qualunque sistema, hardware o software, in un dominio astratto diverso da quello standard. Ogni oggetto del dominio astratto rappresenta un insieme di elementi del dominio concreto. Se il dominio astratto è finito, o soddisfa la ascending chain condition, la computazione astratta termina. Ovviamente il risultato sarà solo una approssimazione della semantica effettiva, ma se il dominio è stato scelto con cura, questo può essere lo stesso sufficiente a ricavare informazioni rilevanti.
La formalizzazione del legame tra dominio concreto e astratto può avvenire in diversi modi [30]. Nella forma pi semplice, si tratta di fornire una relazione tra i due domini, che faccia corrispondere ad ogni oggetto astratto l'insieme degli oggetti concreti che rappresenta.
Nella forma più completa e ricca di proprietà, abbiamo invece una giunzione tra i due domini.
La teoria della interpretazione astratta è stata studiata in profondità.
In particolare, sono state proposte metodologie standard per lo sviluppo di domini astratti complessi, a partire da domini più semplici [31, 32]. Gran parte di questi risultati si è rivelato particolarmente utile nel campo dell'analisi statica di programmi logici, dove il dominio standard è quello degli insiemi di sostituzione, che ha la struttura di un Sistema di Vincoli.

Verifica di Protocolli utilizzando la Programmazione Logica con Vincoli
---
Uno dei problemi più difficili nell'area della verifica dei sistemi reattivi riguarda l'estensione del model checking ai sistemi a stati infiniti. In questa situazione molte proprietà d'interesse sono indecidibili, e spesso nemmeno semidecidibili.
Per affrontare queste limitazioni si sono seguiti due approcci principali.
Il primo approccio consiste nella ricerca ed individuazione di sottoclassi decidibili di sistemi e di proprietà. In questo caso si possono definire tecniche di verifica completamente automatiche, che però non sono applicabili al di fuori delle ristrette classi di sistemi e di proprietà individuate.
Il secondo approccio consiste nell'arricchire il model checking a stati finiti con più generali tecniche deduttive o vari tipi di astrazione.
Tale approccio fornisce una grande generalità ma spesso necessita di guida umana, il che potrebbe rilevarsi un inconveniente nei sistemi di grande dimensione.
Come terza alternativa, e' stato proposto un metodo di verifica che combina la generalità degli approcci basati sulla deduzione con l'automazione degli approcci basati su astrazioni. Tale metodo è automatico, sebbene incompleto, ed utilizza due concetti di base: (1) la programmazione logica con vincoli per specificare i sistemi reattivi e le loro proprietà, e (2) la specializzazione dei programmi come meccanismo di inferenza per la verifica delle proprietà d'interesse.
Il metodo proposto permette di specificare e di verificare automaticamente le proprietà temporali dei protocolli che regolano le attivita' di sistemi multiagente.
Per verificare la validità di una proprietà temporale data, viene applicato un metodo di trasformazione che preserva la semantica del programma che modella il comportamento del sistema, al fine di ottenere un programma equivalente dove la verifica di tale proprietà può però essere verificata in tempo costante.
Con queste tecniche sono state provate proprietà di safety e di liveness per vari protocolli a stati infiniti [34,35].
Inoltre, è stato proposto un metodo di verifica per sistemi a stati infiniti basato su un linguaggio di programmazione logica con vincoli che e' stato applicato con successo per verificare proprietà di safety di un protocollo di mutua esclusione per un sistema multiagente il cui numero di partecipanti può variare al trascorrere del tempo.

2.4.a Riferimenti bibliografici

1. D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Rfc2702: Requirements for traffic engineering over MPLS. Technical report, Network Working Group, 1999.
2. G. Bella and S. Bistarelli. Protocol analysis using soft constraints. Invited talk at S.R.I. Security group, Menlo Park, USA, February 2001.
3. G. Bella and S. Bistarelli. Soft Constraints for Security Protocol Analysis: Confidentiality. In Proc. of PADL'01, LNCS 1990, pages 108-122, 2001.
4. S. Bistarelli. Semirings for Soft Constraint Solving and Programming, volume 2962 of LNCS. Springer, 2004.
5. S. Bistarelli, H. Fargier, U. Montanari, F. Rossi, T. Schiex, and G. Verfaillie. Semiring-based CSPs and Valued CSPs: Frameworks, properties, and comparison. CONSTRAINTS: An international journal. Kluwer, 4(3):199-240, 1999.
6. S. Bistarelli and S. Foley. Analysis of integrity policies using soft constraints. In Proc. of IEEE Workshop Policies for Distributed Systems and Networks, pages 77-80, June 2003.
7. S. Bistarelli and S. Foley. A constraint based framework for dependability goals: Integrity. In Proc. of 22nd International Conference on Computer Safety, Reliability and Security (SAFECOMP2003), volume 2788 of Lecture Notes in Computer Science, pages 130-143. Springer, 2003.
8. S. Bistarelli, S. N. Foley, and B. O'Sullivan. Detecting and eliminating the cascade vulnerability problem from multi-level security

networks using soft constraints. In Proceedings Innovative Applications of Artificial Intelligence Conference (IAAI04). ACM Press, 2004. To appear.
9. S. Bistarelli, S. N. Foley, and B. O'Sullivan. Modelling and detecting the cascade vulnerabiliy problem using soft constraints. In Proc. ACM Symposium on Applied Computing (SAC 2004), pages 383-390. ACM Press, 2004.
10. S. Bistarelli, J. Kelleher, and B. O'Sullivan. Tradeoff Generation using soft constraints. In Springer, editor, Recent Advances in Constraints, Joint ERCIM/CoLogNET International Workshop on Constraint Solving and Constraint Logic Programming, CSCLP'03, Selected Papers, volume 3010 of Lecture Notes in Computer Science, pages 124-139, 2003.
11. S. Bistarelli, U. Montanari, and F. Rossi. Constraint Solving over Semirings. In Proc. IJCAI95, pages 624-630, San Francisco, CA, USA, 1995. Morgan Kaufman.
12. S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based Constraint Solving and Optimization. JACM, 44(2):201-236, 1997.
13. S. Bistarelli, U. Montanari, and F. Rossi. Soft concurrent constraint programming. ACM Transactions on Computational Logic (TOCL), 2001. To Appear.
14. S. Bistarelli, U. Montanari, and F. Rossi. Soft concurrent constraint programming. In Proc. ESOP, April 6 - 14, 2002, Grenoble, France, LNCS, pages 53-67. Springer-Verlag, 2002.
15. S. Bistarelli and B. O'Sullivan. A theoretical framework for tradeoff generation using soft constraints. In Springer, editor, Research and Development in Intelligent Systems XX, Proceedings of AI-2003, the Twenty-third SGAI International Conference on Knowledge-Based Systems and Applied Artificial Intelligence, BCS Conference Series 'Research and Development in Intelligent Systems xx', pages 69-82, 2003.
16. M. Calisti and B. Faltings. Distributed constrained agents for allocating service demands in multi-provider networks,. Journal of the Italian Operational Research Society, XXIX(91), 2000. Special Issue on Constraint-Based Problem Solving.
17. S. Chen and K. Nahrstedt. Distributed QoS routing with imprecise state information. In Proc. 7th IEEE International Conference on Computer, Communications and Networks (ICCCN'98), pages 614-621, 1998.
18. D. Clark. Rfc1102: Policy routing in internet protocols. Technical report, Network Working Group, 1989.
19. J. Fitch and L. Hoffman. A shortest path network security model. Computers and Security, 12(2):169-189, 1993.
20. L. Gong and X. Qian. The complexity and composability of secure interoperation. In Proceedings of the Symposium on Security and Privacy, pages 190-200, Oakland, CA, May 1994. IEEE Computer Society Press.
21. E. Gray. American national standard T1.523-2001, telecom glossary 2000. published on the Web at http://www.its.bldrdoc.gov/projects/telecomglossary2000, 2001.
22. S. Gritalis and D. Spinellis. The cascade vulnerability problem: The detection problem and a simulated annealing approach to its correction. Microprocessors and Microsystems, 21(10):621-628, 1998.
23. J. Horton, R. Harland, E. Ashby, R. Cooper,W. Hyslop, B. Nickerson, W. Stewart, and O. Ward. The cascade vulnerability problem. Journal of Computer Security, 2(4):279-290, 1993.
24. R. Jain and W. Sun. QoS/Policy/ConstraintBased routing. In Carrier IP Telephony 2000 Comprehensive Report. International Engineering Consortium, 2000.
25. G. Lowe. An Attack on the Needham-Schroeder Public-Key Authentication Protocol. Information Processing Letters, 56(3):131-133, 1995.
26. J. Millen and M. Schwartz. The cascading problem for interconnected networks. In 4th Aerospace Computer Security Applications Conference, pages 269-273, Dec. 1988.
27. T. Schiex. Possibilistic constraint satisfaction problems, or "how to handle soft constraints?". In Proc. 8th Conf. of Uncertainty in AI, pages 269-275, 1992.
28. TNI. Trusted computer system evaluation criteria: Trusted network interpretation. Technical report, National Computer Security Center, 1987. Red Book.
29. Cousot, P. and Cousot, R. Systematic design of program analysis frameworks. In Proc. Sixth ACM Symp. Principles of Programming Languages (POPL '79). ACM Press, New York, 269-282, 1979.
30. Cousot, P. and Cousot, R. Abstract Interpretation Frameworks. Journal of Logic Programming 2(4):511-549, 1992.
31. G. Fil`e, R. Giacobazzi and F. Ranzato A Unifying View on Abstract Domain Design ACM Computing Surveys 28(2):333-336, 1996.
32. R. Giacobazzi, F. Ranzato, F. Scozzari Making Abstract Interpretations Complete Journal of the ACM 47(2):361-416, 2000.
33. F. Fioravanti, A. Pettorossi, and M. Proietti. Verification of sets of infinite state systems using program transformation. In A. Pettorossi (ed.), Proc. LOPSTR 2001, LNCS 2372, pp. 111-128. Springer, 2002.
34. F. Fioravanti, A. Pettorossi, and M. Proietti. Automatic Proofs of Protocols via Program Transformation. In Proceedings of the International Workshop on Monitoring, Security, and Rescue Techniques in Multi-Agents Systems (MSRAS '04), June7-9, 2004, Poland, 2004.
35. MAP group. The MAP transformation system. Available from: http://www.iasi.rm.cnr.it/~proietti/system.html, 1995-2004.
36. M. Abadi Security protocols and specifications. In Proceedings of FOSSACS '99, pages 1-13, 1999
37. R. Giacobazzi and F. Scozzari A logical model for relational abstract domains. ACM Transactions on Programming Languages (TOCL), 20(5):1067-1109, 1998
38. F. Scozzari Logical optimality of groundness analysis Theoretical Compiter Science 277(1-2):149-184, 2002
39. L. Aceto and D. Murphy. Timing and causality in process algebra. Acta Informatica, 33(4):317-350, 1996.
40. J. Baeten and J. Bergstra. Real time process algebra. Formal Aspects of Computing, 3(2):142-188, 1991.
41. G. Berry and G. Gonthier. The esterel programming language: Design, semantics and implementation. Science of Computer Programming, 19(2):87-152, 1992.
42. P. Bremond-Gregoire and I. Lee. A Process Algebra of Communicating Shared Resources with Dense Time and Priorities. Theoretical Computer Science, 189(1-2):169-219, 1997.
43. P. Caspi, N. Halbwachs, D. Pilaud, and P. Raymond. The synchronous dataflow programming language LUSTRE. In Special issue on Another Look at Real-time Systems, volume 79 of Proceedings of the IEEE, pages 1305-1319. IEEE Computer Society Press, 1991.
44. F. S. de Boer, M. Gabbrielli, and M. C. Meo. A Timed CCP Language. Information and Computation, 161(1):45-83, 2000.
45. M. Fisher. An introduction to Executable Temporal Logics. Knowledge Engineering Review, 6(1):43-56, 1996.
46. M. Hennessy and T. Regan. A temporal process algebra. Information and Computation, 117:221-239, 1995.
47. Z. Manna and A. Pnueli. The temporal logic of reactive systems. Springer-Verlag, 1991.
48. V. A. Saraswat, R. Jagadeesan, and V. Gupta. Timed Default Concurrent Constraint Programming. Journal of Symbolic Computation, 22(5-6):475-520, 1996.

49. F. S. de Boer, M. Gabbrielli and M. C. Meo. Proving Correctness of Timed Concurrent Constraint Programs. ACM Transactions on Computational Logic (TOCL), 5(4), 2004.

2.5 Descrizione del programma e dei compiti dell'Unità di Ricerca

La proposta di ricerca ha le basi sulle conoscenze già acquisite dai componenti dell'unità nei campi dei vincoli soft, dell'astrazione e della concorrenza.
L'attività da una parte mirerà ad estendere il formalismo dei vincoli soft e dall'altra parte studierà possibili utilizzi di vincoli soft con astrazione e concorrenza per lo studio di sistemi e applicazioni.
In particolare, nostro scopo è lo studio di sistemi ed applicazioni nei campi della sicurezza informatica, e nell'analisi e valutazione del rischio in sicurezza. Ci proponiamo più in dettaglio i seguenti obiettivi:

Estensione del framework dei Vincoli Soft

Il framework dei vincoli basati su semiring (semiring-based CSP (SCSP)) [12, 4, 11, 5, 14, 13] è capace di rappresentare molte classi di vincoli soft (fuzzy, pesati, probabilistici, multicriteria, ...), a seconda del significato dato ai valori associati ad ogni tupla (interpretati come livelli di preferenza, di importanza o costi). I vincoli soft basati su semiring sono così chiamati perchè basati proprio su una sottostante struttura di semiring che definisce l'insieme delle preferenze, l'ordinamento tra queste e la modalità di combinarle. Questo concetto è molto generale e può essere istanziato al fine di ottenere molte delle classi di vincoli soft già proposte, i loro algoritmi di risoluzione e anche di nuovi.
Tuttavia, il framework, pur ricco, non è sufficiente quando deve essere rappresentato non solo ciò che l'utente desidera ma anche ciò che non desidera.
Inoltre il framework necessita ancora molto studio riguardo agli algoritmi risolutivi.
*** Risultati attesi: ****
Ci si propone di arricchire la struttura di semiring sia dal punto di vista dell'insieme di preferenze (in modo da rappresentare non solo valori positivi per ciò che l'utente desidera ma anche negativi per ciò che l'utente non desidera), che dal punto di vista degli operatori.
In particolare saranno introdotti altri operatori (con semantica in qualche modo opposta agli operatori "+" e "x") che saranno utili per introdurre l'uso dì vincoli soft in nuovi ambiti applicativi. Un ambito di particolare interesse sembra quello dei Data Base a vincoli. Qui un operatore opposto al "+" sembra necessario per estendere al caso soft la nozione di differenza insiemistica.
Ci si propone anche di studiare e modificare vari algoritmi risolutivi usati nel caso dei vincoli classici e verificare la loro applicazione nel caso di vincoli soft.

Vincoli Soft per l'analisi di sistemi con sicurezza multilivello
---
La sicurezza di una rete è basata non solo sulla sicurezza delle sue componenti e delle interconnessione dirette tra loro, ma anche sulla potenziale possibilità che i sistemi possono avere di interoperare indirettamente attraverso le connessioni di rete. Tale possibilità può infatti potenzialmente creare dei "cascading path" [28] che violano la sicurezza dei sistemi attraverso le connessioni di rete. In [9, 8] è mostrato come la programmazione con vincoli fornisce un approccio naturale per esprimere i vincoli necessari per assicurare la sicurezza multilivello attraverso la rete. In particolare i vincoli soft possono essere usati per evidenziare ed eliminare i cascading path nella rete che violano la sicurezza.
L'approccio con vincoli soft sembra fornire un avanzamento rispetto alle tecniche usate per risolvere questo problema. La ricerca attualmente studia come evidenziare i "cascading paths" e come eliminarli riconfigurando la rete. Se evidenziare il problema può essere fatto facilmente [26, 23, 19, 20], l'eliminazione delle vuulnerabilità eliminando il minimo numero di connessioni e' un problema NP-completo [23, 22, 20].
Usando la programmazione con vincoli soft e' possibile evidenziare l'insieme di tutti i cascading path e quindi ottenere una soluzione minimale (anche se non ottima) in tempo polinomiale.
*** Risultati attesi: ****
Il modello a vincoli può fornire una descizione naturale di una arbitraria rete a sicurezza multilivello. Ogni soluzione del modello rappresenta un cascading path nella rete fornendo molte più informazioni sulle vulnerabilità degli approcci classici e fornendo informazioni per la loro eliminazione.
Usando un modello a vincoli possiamo utilizzare le tecniche risolutive sviluppate in questo campo per trovare l'insieme dei cascading path con uno sforzo ragionevole, e quindi eliminare il problema in tempo polinomiale.
Ci proponiamo di applicare la tecnologia a vincoli ad alcuni casi reali, iniziando alcune collaborazioni con società interessate. Siamo in trattativa per collaborazioni su questi argomenti.

Analisi Quantitativa di Politiche di Integrità

Una politica di integrità definisce le situazioni in cui le modifiche alle informazioni sono autorizzate e attuate dai meccanismi di sicurezza del sistema. In complessi sistemi applicativi è però possibile che una politica di integrità sia specificata in maniera non corretta e, come conseguenza, un utente sia autorizzato a modificare informazioni che possono portare ad una inattesa compromissione del sistema. In [6, 7] è stata proposta una tecnica quantitativa scalabile che usa la risoluzione di vincoli per modellare ed analizzare l'efficacia delle politiche di integrità di un sistema applicativo.
*** Risultati Attesi: ****
Piuttosto che tentare di modellare il comportamento completo del sistema e dell'infrastruttura, nell'approccio basato su vincoli, solo quei componenti che sono considerati rilevanti per la politica di sicurezza hanno bisogno di essere modellati imponendo dei vincoli sulle parti del sistema rilevanti per la sicurezza. In questa maniera è possibile risolvere la consistenza delle politiche di sicurezza come un problema di soddisfazione di vincoli. Inoltre, usando vincoli soft diventa possibile effettuare un'analisi quantitativa delle policy.
Un'analisi quantitativa fornisce una misura a grana fine di quanto sia sicuro un sistema, piuttosto che utilizzare una misura grossolana (vero/falso) fornita dai convenzionali vincoli 'crisp'.

Analisi Quantitativa di Protocolli di Sicurezza

I protocolli di sicurezza prescrivono la maniera in cui i soggetti remoti presenti su una rete di computer devono interagire per ottenere specifici obiettivi di sicurezza. Un obiettivo principale dei protocolli è la confidenzialità, che garantisce che un messaggio rimane incomprensibile a soggetti "maliziosi". Un altro obiettivo cruciale è l'autenticazione, che garantisce la partecipazione di un soggetto ad una sessione di protocollo. Questi obiettivi sono formalizzati con una semplice alternativa si/no nella letteratura esistente. In questa maniera è possibile solo affermare se una chiave è confidenziale oppure no, o se un soggetto si autentica con un altro oppure no.
D'altro canto, l'esperienza mostra che la sicurezza nel mondo reale è basata su livelli di sicurezza piuttosto che su garanzie categoriche e definitive di sicurezza. In particolare, i livelli di sicurezza caratterizzano gli obiettivi di confidenzialità e di autenticazione di un protocollo. In riferimento al primo di questi obiettivi, ricordiamo che messaggi diversi richiedono specifici livelli di protezione contro la divulgazione [21].
Ad esempio, una password di un utente richiede un livello di protezione maggiore di una chiave di sessione, che viene usata per una sola sessione di protocollo. Intuitivamente, una password dovrebbe essere "più confidenziale" di una chiave di sessione. Inoltre, un attacco alla confidenzialità basato su criptoanalisi online non dovrebbe essere imputato al design del protocollo. E' abbastanza sorprendente che gli obiettivi di confidenzialità e di autenticazione di un protocollo siano formalizzati con una semplice alternativa si/no nella letteratura esistente.
*** Risultati Attesi: ***
La motivazione della nostra ricerca è lo sviluppo di una nozione quantitativa degli obiettivi di sicurezza (confidenzialità, autenticazione, delega, ecc.). Ci proponiamo inoltre di estendere il nucleo esistente [3, 2], e di dimostrare la validità della nostra idea su protocolli largamente diffusi (come abbiamo fatto per Kerberos, ad esempio). In Kerberos, la nostra analisi preliminare del protocollo evidenzia il fatto che la perdita di una chiave di autorizzazione sarebbe più grave della perdita di una chiave di servizio, e che l'autenticazione del responder con l'initiator è più debole dell'autenticazione dell'initiator con il responder.
Il prossimo passo sarà la meccanizzazione del framework. Infatti, poiché abbiamo a che fare esclusivamente con quantità illimitate ma finite, il framework può essere automatizzato utilizzando il model checking.

Verifica di Protocolli di Sicurezza

Ci si propone di studiare i vantaggi e gli svantaggi derivanti dall'applicazione di tecniche basate sulla programmazione logica con vincoli all'analisi, la specifica, la verifica e l'attuazione di proprietà di sicurezza in sistemi multiagente.
Particolare attenzione sarà posta sulla specifica e sullo studio di politiche di autorizzazione in ambienti distribuiti attraverso l'uso dei vincoli.
*** Risultati attesi: ***
Ci si propone di ottenere risultati teorici e sperimentali riguardo l'uso della programmazione logica con vincoli come framework per la costruzione e la validazione di modelli di autorizzazione distribuiti in sistemi multiagente.
In particolare, concentreremo la nostra attenzione sui problemi che nascono dalla composizione e dall'interoperabilita' tra politiche di autorizzazione appartenenti ad organizzazioni diverse.

Completezza dei domini astratti per la verifica di protocolli di sicurezza

Un protocollo di sicurezza ([36]) è un metodo per trasmettere informazioni attraverso la rete. Il ruolo principale dei protocolli di sicurezza è di garantire l'autenticità e la segretezza della comunicazione.
Altre proprietà richieste possono riguardare il riconoscimento o l'integrità del sistema. I metodi comunemente utilizzati per analizzare le proprietà di sicurezza sono basati sul controllo del flusso di informazione. L'idea è di partizionare l'informazione in classi di sicurezza e di garantire che non vi siano flussi di informazione tra le varie classi. Questo problema viene tipicamente risolto fissando una logica particolare o un type system, oppure utilizzando tecniche di model checking basate sulla semantica operazionale.
*** Risultati attesi: ***
Noi vogliamo fornire delle tecniche formali per la progettazione sistematica di domini astratti orientati alle proprietà di sicurezza, e quindi di rendere applicabili le tecniche di trasformazione dei domini [32] all'analisi dei protocolli di comunicazione. Le tecniche di trasformazione dinamica del dominio di analisi giocano un ruolo chiave nel caso delle proprietà di sicurezza, dove non è sufficiente dimostrare che il sistema soddisfi una certa specifica (correttezza dell'analisi) bensì deve essere garantito che il protocollo resista agli attacchi. La novità introdotta è l'inferenza attiva dell'ambiente circostante, che rende necessario l'utilizzo di tecniche dinamiche di modifica dell'analisi, in accordo all'evoluzione dell'ambiente. Si intendono quindi definire ed implementare delle operazioni di trasformazione dei domini (raffinamenti e semplificatori) per la progettazione di analisi ottimali (con il migliore tradeoff tra la complessità computazionale e la precisione dell'analisi). Esempi di analisi ottimali basate su varie logiche sono gi state considerate in molte applicazioni dell'interpretazione astratta (vedi, ad esempio, [37, 38]).
La stessa idea può essere estesa all'analisi di sicurezza, sfruttando le strutture logiche alla base delle specifiche dei protocolli.

Metodologia di analisi uniforme

Le metodologie utilizzate per analizzare i protocolli di sicurezza e le politiche di integrità sono differenti ma strettamente correlate. Da una parte c'à la nozione di "ambiente" dentro la quale la politica viene implementata, dall'altra c'è l'attaccante che mira a violare il protocollo. L'idea base da usare per integrare i due approcci è quella di considerare, in entrambi i framework, la nozione di "modello ideale" che rappresenta la corretta configurazione della politica e la corretta esecuzione del protocollo. Si tratterà poi di confrontare la poltica e la "implementazione" del protocollo con tale modello.
***** Risultati attesi *****
Abbiamo in mente di sviluppare un approccio uniforme allo studio degli attacchi ai protocolli di sicurezza e dei bug sulle politiche di integrità. Pensiamo anche di svuluppare una implementaione che sarà utile per controllare automaticamente protocolli e politiche rispetto a questa nuova nozione di sicurezza. Pensiamo di partire da una lista di protocolli di sicurezza e da un insieme di politiche, da utilizzare come test per la nostra teoria.
E` importante sottolineare che la metodologia proposta potrebbe riconoscere nuovi bug, e dare una misura a bug o attacchi già noti. In particolare, le nozioni di "attack detection", "attack suspicion" e "attack retaliation" saranno studiate in dettaglio.

Analisi di rischi di sicurezza e tradeoff tra privacy ed efficienza
---
Una importante nuova area di ricerca è collegata alla analisi di rischio di sicurezza. Pensiamo di usare i vincoli soft per rappresentare le attività che necessitano di protezione e le minacce che potrebbero causare dei danni.
L'uso di vincoli soft è necessario per rappresentare il "costo" associato a ogni specifica minaccia o attività. Inoltre, la probabilità delle varie minacce deve essere presa in considerazione. Una volta che il sistema è stato modellato, la nozione di tradeoff [15,10] potrebbe essere usata per modificare parzialmente la configurazione del sistema.
Un'altra preoccupazione è la privacy. Durante l'esecuzione, un agente potrebbe non voler rivelare troppa informazione. Pertanto, si viene a creare un tradeoff, potenzialmente importante, tra la tutela della privacy e il potenziamento dell'efficienza della ricerca. In questo lavoro, mostriamo come misure quantitative della perdita di privacy posso essere fatto all'interno del framework del soddisfacimento di vincoli distribuito.
****risultati attesi****
Vogliamo mostrare come gli agenti posso fare inferenze sui problemi o sottoproblemi di altri agenti a partire da comunicazioni che non trasportano nessuna informazione privata in maniera esplicita. Ciò può avvenite usando dei ragionamenti basati su vincoli, in un framework costituito da un CSP ordinario, che è noto solo parzialmente, e un sistema di CSP ombra che rappresentano varie forme di conoscenza probabilistica. Questo tipo di ragionamento, combinato con l'elaborazione della consistenza degli archi, può velocizzare la ricerca sotto condizioni di comunicazione limitata, allo stesso tempo compromettendo la tutela della privacy.
In alcuni casi, un piccola quantità di informazione privata è richiesta per migliorare l'efficienza della ricerca; con l'uso di euristiche più sofisticate, la ricerca può essere migliorata anche sotto condizioni di comunicazione minimale. Allo stesso tempo, questi metodi consentono, talvolta, di inferire informazioni nascoste, sollevando nuove problematiche riguardanti la privacy.
In quest'area pensiamo di intraprendere una notevole mole di lavoro. Vorremmo creare un modello per rappresentare sia informazione di sicurezza che aspetti economici (tipicamente il costo della minaccia e il costo delle patch o delle contromisure di sicurezza). In questo contesto, la teoria della probabilità e la teoria delle possibilità saranno usate per trattare il problema dell'incertezza.

2.6 Descrizione delle attrezzature già disponibili ed utilizzabili per la ricerca proposta con valore patrimoniale superiore a 25.000 Euro

Nessuna

2.7 Descrizione delle Grandi attrezzature da acquisire (GA)

Nessuna

2.8 Mesi uomo complessivi dedicati al programma

		Numero	Mesi uomo 1° anno	Mesi uomo 2° anno	Totale mesi uomo
Personale universitario dell'Università sede dell'Unità di Ricerca		*4*	*23*	*21*	*44*
Personale universitario di altre Università		*0*	*0*	*0*	*0*
Titolari di assegni di ricerca		*0*			
Titolari di borse	*Dottorato*	*1*	*7*	*5*	*12*
	Post-dottorato	*0*			
	Scuola di Specializzazione	*0*			
Personale a contratto	*Assegnisti*	*0*			
	Borsisti	*0*			
	Dottorandi	*0*			
	Altre tipologie	*0*			
Personale extrauniversitario		*0*			
TOTALE		**5**	**30**	**26**	**56**

PARTE III

3.1 Costo complessivo del Programma dell'Unità di Ricerca

Voce di spesa	Spesa in Euro	Descrizione
Materiale inventariabile	*12.000*	*Acquisto di PC, notebooks e stampanti*
Grandi Attrezzature		
Materiale di consumo e funzionamento	*6.000*	*carta, cancelleria, ricambi, toner stampanti e 8% del progetto (4240)*
Spese per calcolo ed elaborazione dati		
Personale a contratto		
Servizi esterni	*2.000*	*Spese varie di fotocopisteria, fax e stampa*
Missioni	*13.000*	*Spese per visite di ricerca presso le unità componenti del progetto e presso centri di ricerca la cui attività è legata al progetto*
Pubblicazioni	*3.000*	*Spese per technical report e pubblicazioni*
Partecipazione / Organizzazione convegni	*17.000*	*Partecipazioni a convegni collegati ai topics della ricerca, in Italia, Europa e Asia/America*
Altro		
TOTALE	**53.000**	

3.2 Costo complessivo del Programma di Ricerca

		Descrizione
Costo complessivo del Programma dell'Unità di Ricerca	*53.000*	
Fondi disponibili (RD + RA) ***comprensivi dell'8% max per spese di gestione***	*15.900*	*Da fondi di Ateneo dei componenti strutturati*
Cofinanziamento di altre amministrazioni		
Cofinanziamento richiesto al MIUR	*37.100*	

3.3.1 Certifico la dichiarata disponibilità e l'utilizzabilità dei fondi di Ateneo (RD e RA)

SI