

# *Curriculum* dell'attività scientifica e didattica di Igor Melatti

redatto ai sensi degli artt. 46 e 47 del d.p.r. 28.12.2000, n. 445 (dichiarazioni  
sostitutive di certificazioni e dell'atto di notorietà)

Sapienza Università di Roma, Dip. di Informatica, via Salaria 113, 00198 Roma  
melatti@di.uniroma1.it  
Tel: +39 06 4991 8438

## Esperienze accademiche

- Dal 30/12/2010 è Ricercatore Universitario presso il Dipartimento di Informatica dell'Università “La Sapienza” di Roma
- Dal 01/05/2010 al 29/12/2010 è stato titolare, presso il Dipartimento di Informatica dell'Università “La Sapienza” di Roma, di un assegno di ricerca di durata annuale, dal titolo “Progettazione ed implementazione di algoritmi e tools per la verifica automatica (model checking) e la sintesi di sistemi ibridi”, con responsabile scientifico il Prof. Enrico Tronci.
- Dal 01/02/2006 al 31/01/2010 è stato titolare, presso il Dipartimento di Informatica dell'Università “La Sapienza” di Roma, di un assegno di ricerca, dal titolo “Algoritmi e tools per la verifica automatica di sistemi deterministici e stocastici”, con responsabile scientifico il Prof. Enrico Tronci.
- Dal 01/07/2005 al 31/12/2005 (e ancora dal 15/07/2006 al 15/09/2006) è stato *Post Doctoral Research Associate* presso la School of Computing della University of Utah (Salt Lake City, UT, USA), con referente il Prof. Ganesh Gopalakrishnan.

## Titoli di studio

- 06/06/2005: dottore di ricerca in Informatica ed Applicazioni presso l'Università dell'Aquila, discutendo la tesi “Explicit Algorithms for Probabilistic Model Checking”, con tutore il Prof. Benedetto Intrigila
- 10/04/2001: laurea in Informatica con 110/110 e lode, presso l'Università dell'Aquila. Completa gli studi nei 5 anni previsti dal piano didattico, discutendo la tesi dal titolo “Uso di SPIN in un approccio probabilistico alla verifica automatica di sistemi concorrenti”, relatore Prof. Enrico Tronci
- 1997: diploma in Pianoforte Principale con 10/10 presso il Conservatorio di Musica “A. Casella” dell'Aquila
- 1995: diploma di maturità classica con 60/60 presso il Liceo Ginnasio “D. Cotugno” dell'Aquila

## Pubblicazioni

### Journal

- (j1). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Model Based Synthesis of Control Software from System Level Formal Specifications.” *ACM TRANSACTIONS ON SOFTWARE ENGINEERING AND METHODOLOGY* To appear (2014). ACM. ISSN: 1049-331X.
- (j2). Giuseppe Della Penna, Benedetto Intrigila, Daniele Magazzeni, Igor Melatti, and Enrico Tronci. “CGMurphi: Automatic synthesis of numerical controllers for nonlinear hybrid systems.” *European Journal of Control* 19, no. 1 (2013): 1436. Elsevier North-Holland, Inc.. ISSN: 0947-3580.
- (j3). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Linear Constraints and Guarded Predicates as a Modeling Language for Discrete Time Hybrid Systems.” *International Journal on Advances in Software* vol. 6, nr 1&2 (2013): 155169. IARIA. ISSN: 1942-2628.
- (j4). Federico Mari, Igor Melatti, Enrico Tronci, and Alberto Finzi. “A multi-hop advertising discovery and delivering protocol for multi administrative domain MANET.” *Mobile Information Systems* 3, no. 9 (2013): 261280. IOS Press. ISSN: 1574-017x (Print) 1875-905X (Online).
- (j5). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Synthesizing Control Software from Boolean Relations.” *International Journal on Advances in Software* vol. 5, nr 3&4 (2012): 212223. IARIA. ISSN: 1942-2628.
- (j6). G. Della Penna, D. Magazzeni, B. Intrigila, I. Melatti, E. Tronci, M. Venturini Zilli, E. Ciancamerla, M. Minichino, A. Tofani. Automatic Verification of Hybrid System Controllers with the CMurphi Verifier. *IJDECS*, 1(1): 69-82, 2011, Serial Publications
- (j7). I. Melatti, R. Palmer, G. Sawaya, Y. Yang, R. M. Kirby, and G. Gopalakrishnan. Parallel and Distributed Model Checking in Eddy. *STTT*, 11(1): 13-25, 2009, Springer
- (j8). G. Della Penna, B. Intrigila, D. Magazzeni, I. Melatti, A. Tofani, E. Tronci. Automated Generation Of Optimal Controllers Through Model Checking Techniques. *Informatics in Control, Automation and Robotics: Selected Papers from the International Conference on Informatics in Control, Automation and Robotics 2006*, pp. 107-122, 2008, Springer Publishing Company, Incorporated (book chapter)
- (j9). B. Intrigila, I. Melatti, A. Tofani, and G. Macchiarelli. Computational Models of Myocardial Endomysial Collagen Arrangement. *Computer Methods and Programs in Biomedicine*, 86(3):232-244, 2007, Elsevier North-Holland, Inc.
- (j10). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Finite Horizon Analysis of Markov Chains with the Murphi Verifier. *STTT*, 8(4):397 - 410, 2006, Springer

- (j11). G. Della Penna, A. Di Marco, B. Intrigila, I. Melatti, A. Pierantonio Interoperability Mapping from XML Schemas to ER Diagrams. *Data & Knowledge Engineering*, 59:166 – 188, 2006, Elsevier
- (j12). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Exploiting Transition Locality in Automatic Verification of Finite State Concurrent Systems. *STTT*, 6(4):320–341, 2004, Springer

### **Proceedings di Conferenze Internazionali**

- (c1). Toni Mancini, Federico Mari, Annalisa Massini, Igor Melatti, and Enrico Tronci. “System Level Formal Verification via Distributed Multi-Core Hardware in the Loop Simulation.” In Proc. of the 22nd Euromicro International Conference on Parallel, Distributed and Network-Based Computing. IEEE Computer Society, 2014.
- (c2). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “A Map-Reduce Parallel Approach to Automatic Synthesis of Control Software.” In Proc. of International SPIN Symposium on Model Checking of Software (SPIN 2013), 4360. Lecture Notes in Computer Science 7976. Springer - Verlag, 2013. ISSN: 0302-9743. ISBN: 978-3-642-39175-0.
- (c3). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “On-the-Fly Control Software Synthesis.” In Proc. of International SPIN Symposium on Model Checking of Software (SPIN 2013), 6180. Lecture Notes in Computer Science 7976. Springer - Verlag, 2013. ISSN: 0302-9743. ISBN: 978-3-642-39175-0.
- (c4). Toni Mancini, Federico Mari, Annalisa Massini, Igor Melatti, Fabio Merli, and Enrico Tronci. “System Level Formal Verification via Model Checking Driven Simulation.” In Proceedings of the 25th International Conference on Computer Aided Verification (CAV 2013), 296312. Lecture Notes in Computer Science 8044. Springer - Verlag, 2013. ISSN: 0302-9743. ISBN: 978-3-642-39798-1.
- (c5). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Automatic Control Software Synthesis for Quantized Discrete Time Hybrid Systems.” In Proceedings of the 51th IEEE Conference on Decision and Control (CDC 2012), 61206125. IEEE, 2012. ISBN: 978-1-4673-2065-8.
- (c6). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “On Model Based Synthesis of Embedded Control Software.” In Proceedings of the 12th International Conference on Embedded Software (EMSOFT 2012), 227236. ACM, 2012. ISBN: 978-1-4503-1425-1.
- (c7). Ed Kuijpers, Luigi Carotenuto, Jean-Cristophe Malapert, Daniela Markov-Vetter, Igor Melatti, Andrea Orlandini, and Ranni Pinchuk. “Collaboration on ISS Experiment Data and Knowledge Representation.” In Proceedings of the 63rd International Astronautical Congress (IAC 2012). Vol. D.5.11, 2012.
- (c8). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Control Software Visualization.” In Proceedings of the Second International Conference on Advanced Commu-

nications and Computation (INFOCOMP 2012), 1520. ThinkMind, 2012. ISSN: 978-1-61208-226-4.

- (c9). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Linear Constraints as a Modeling Language for Discrete Time Hybrid Systems.” In Proceedings of the Seventh International Conference on Software Engineering Advances (ICSEA 2012), 664671. ThinkMind, 2012.
- (c10). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. “Undecidability of Quantized State Feedback Control for Discrete Time Linear Hybrid Systems.” In Theoretical Aspects of Computing (ICTAC 2012), 243258. Lecture Notes in Computer Science 7521. Springer Berlin Heidelberg, 2012. ISBN: 978-3-642-32942-5.
- (c11). Verzino Giovanni, Federico Cavaliere, Federico Mari, Igor Melatti, Giovanni Minei, Ivano Salvo, Yuri Yushstein, and Enrico Tronci. “Model checking driven simulation of sat procedures.” In Proc. of 12th International Conference on Space Operations (SpaceOps 2012), 2012.
- (c12). F. Mari, I. Melatti, I. Salvo, and E. Tronci, From Boolean Relations to Control Software. In Proc. of the Sixth International Conference on Software Engineering Advances *ICSEA '11*, ISBN: 978-1-61208-165-6
- (c13). F. Cavaliere, F. Mari, I. Melatti, G. Minei, I. Salvo, E. Tronci, G. Verzino, and Y. Yushstein, Model Checking Satellite Operational Procedures. In Proc. of the Eurospace DATA Systems In Aerospace Conference *DASIA '11*, ESA Proceedings ESA SP-694, ISBN 978-92-9092-258-2
- (c14). F. Mari, I. Melatti, I. Salvo, E. Tronci. Synthesis of Quantized Feedback Control Software for Discrete Time Linear Hybrid Systems. In Proc. of the 22nd International Conference on Computer Aided Verification *CAV '10*, volume 6174 of *Lecture Notes in Computer Science*, pages 180–195. Springer, 2010.
- (c15). S. Mazzini, S. Puri, F. Mari, I. Melatti, E. Tronci, Formal Verification at System Level. In ESA SP-669, Proc. of the Eurospace DATA Systems In Aerospace Conference *DASIA '09*, ESA Proceedings ESA SP-669, ISBN 978-92-9221-233-9
- (c16). A. Bobbio, E. Ciancamerla, S. Di Blasi, A. Iacomini, F. Mari, I. Melatti, M. Minichino, A. Scarlatti, E. Tronci, R. Terruggia, E. Zendri. Risk analysis of SCADA systems interconnecting Power Grids and Telco Networks via heterogeneous models and tools. In Proc. of the 4th International Conference on Risks and Security of Internet and Systems *CRISIS '09*, pages 90–97, IEEE Proceedings.
- (c17). F. Mari, I. Melatti, I. Salvo, E. Tronci, L. Alvisi, A. Clement and H. Li. Model Checking Coalition Nash Equilibria in MAD Distributed Systems. In Rachid Guerraoui and Franck Petit, editors, *Stabilization, Safety, and Security of Distributed Systems, 11th International Symposium, SSS 2009, Lyon, France, November 3-6, 2009. Proceedings*, volume 5873 of *Lecture Notes in Computer Science*, pages 531–546. Springer, 2009.
- (c18). F. Mari, I. Melatti, I. Salvo, E. Tronci, L. Alvisi, A. Clement and H. Li. Model Checking Nash Equilibria in MAD Distributed Systems. In Proc. of the 8th Conference on Formal Methods in Computer Aided Design *FMCAD '08* (IEEE Computer Society).

- (c19). F. Brizzolari, G. Della Penna, I. Melatti, E. Tronci. Disk Based Software Verification via Bounded Model Checking. In Proc. of the 14th Asia-Pacific Software Engineering Conference *APSEC '07* (IEEE Computer Society).
- (c20). G. Della Penna, D. Magazzeni, A. Tofani, B. Intrigila, I. Melatti, E. Tronci. Automatic Synthesis of Robust Numerical Controllers. In Proc. of the 3rd International Conference on Autonomic and Autonomous Systems *ICAS '07* (IEEE Computer Society).
- (c21). G. Della Penna, B. Intrigila, D. Magazzeni, I. Melatti, A. Tofani, E. Tronci. Automatic Generation Of Optimal Controllers Through Model Checking Techniques. Proceedings of 3rd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2006)
- (c22). G. Della Penna, B. Intrigila, I. Melatti, M. Pecorari, A. Tofani, E. Tronci. A Case Study on Automated Generation of Integration Tests. Proceedings of Forum on specification & Design Languages (FDL 2006)
- (c23). I. Melatti, R. Palmer, G. Sawaya, Y. Yang, R. M. Kirby, and G. Gopalakrishnan. Parallel and Distributed Model Checking in Eddy. A. Valmari, editor, *Model Checking Software, 13th International SPIN Workshop, Vienna, Austria, March 30 – April 1, 2006, Proceedings*, volume 3925 of *Lecture Notes in Computer Science*. Springer, 2005.
- (c24). B. Intrigila, D. Magazzeni, I. Melatti, A. Tofani, E. Tronci. A Model Checking Technique for the Verification of Fuzzy Control Systems. IEEE proceedings of the *International Conference on Computational Intelligence for Modelling Control and Automation (CIMCA 2005)*.
- (c25). G. Della Penna, B. Intrigila, I. Melatti, and E. Tronci. Exploiting Hub States in Automatic Verification. D.A. Peled and Y.-K. Tsay, editors, *Automated Technology for Verification and Analysis: Third International Symposium, ATVA 2005, Taipei, Taiwan, October 4-7, 2005, Proceedings*, volume 3707 of *Lecture Notes in Computer Science*, pages 54–68. Springer, 2005.
- (c26). B. Intrigila, G. Macchiarelli, I. Melatti, A. Tofani. Computational Models of the Micro Architecture of the Cardiac Endomysial Collagen. *IFMBE Proceedings EMBEC'05 "3rd European Medical & Biological Engineering Conference, IFMBE European Conference on Biomedical Engineering"*, Vol. 11, 2005, Prague, Czech Republic, CD
- (c27). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Bounded Probabilistic Model Checking with the Murphi Verifier. In Carlo Blundo and Cosimo Laneve, editors, *Formal Methods in Computer-Aided Design, 5th International Conference, FMCAD 2004, Austin, TX, USA, November 14-17, 2004, Proceedings*, volume 3312 of *Lecture Notes in Computer Science*, pages 214–229. Springer, 2004.
- (c28). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Finite Horizon Analysis of Markov Chains with the Murphi Verifier. In Daniel Geist and Enrico Tronci, editors, *Correct Hardware Design and Verification Methods, 12th IFIP WG 10.5 Advanced Research Working Conference, CHARME 2003, L'Aquila, Italy, October 21-24, 2003, Proceedings*, volume 2860 of *Lecture Notes in Computer Science*, pages 394–409. Springer, 2003.

- (c29). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Integrating Ram and Disk Based Verification within the Murphi Verifier. In Daniel Geist and Enrico Tronci, editors, *Correct Hardware Design and Verification Methods, 12th IFIP WG 10.5 Advanced Research Working Conference, CHARME 2003, L'Aquila, Italy, October 21-24, 2003, Proceedings*, volume 2860 of *Lecture Notes in Computer Science*, pages 277–282. Springer, 2003.
- (c30). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Finite Horizon Analysis of Stochastic Systems with the Murphi Verifier. In Carlo Blundo and Cosimo Laneve, editors, *Theoretical Computer Science, 8th Italian Conference, ICTCS 2003, Bertinoro, Italy, October 13-15, 2003, Proceedings*, volume 2841 of *Lecture Notes in Computer Science*, pages 58–71. Springer, 2003.
- (c31). G. Della Penna, B. Intrigila, I. Melatti, M. Minichino, E. Ciancamerla, A. Parisse, E. Tronci, and M. Venturini Zilli. Automatic Verification of a Turbogas Control System with the Murphi Verifier. In Oded Maler and Amir Pnueli, editors, *Hybrid Systems: Computation and Control, 6th International Workshop, HSCC 2003 Prague, Czech Republic, April 3-5, 2003, Proceedings*, volume 2623 of *Lecture Notes in Computer Science*, pages 141–155. Springer, 2003.
- (c32). G. Della Penna, A. Di Marco, B. Intrigila, I. Melatti, A. Pierantonio Xere: Towards a Natural Interoperability between XML and ER Diagrams. In Mauro Pezzè, editor, *Fundamental Approaches to Software Engineering, 6th International Conference, FASE 2003, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2003, Warsaw, Poland, April 7-11, 2003, Proceedings*, volume 2621 of *Lecture Notes in Computer Science*, pages 356–371. Springer, 2003.

## Rapporti Tecnici

- (tr1). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. Model Based Synthesis of Control Software from System Level Formal Specifications. Vol. abs/1107.5638. CoRR, Technical Report, 2013. <http://arxiv.org/abs/1107.5638> (accessed January 4, 2014).
- (tr2). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. A Map-Reduce Parallel Approach to Automatic Synthesis of Control Software. Vol. abs/1210.2276. CoRR, Technical Report, 2012. <http://arxiv.org/abs/1210.2276> (accessed January 4, 2014).
- (tr3). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. Automatic Control Software Synthesis for Quantized Discrete Time Hybrid Systems. Vol. abs/1207.4098. CoRR, Technical Report, 2012. <http://arxiv.org/abs/1207.4098> (accessed January 4, 2014).
- (tr4). Vadim Alimguzhin, Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. On Model Based Synthesis of Embedded Control Software. Vol. abs/1207.4474. CoRR, Technical Report, 2012. <http://arxiv.org/abs/1207.4474> (accessed January 4, 2014).
- (tr5). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. From Boolean Functional Equations to Control Software. Vol. abs/1106.0468. CoRR, Technical Report, 2011. <http://arxiv.org/abs/1106.0468> (accessed January 4, 2014).

- (tr6). Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci. Quantized Feedback Control Software Synthesis from System Level Formal Specifications for Buck DC/DC Converters. Vol. abs/1105.5640. CoRR, Technical Report, 2011. <http://arxiv.org/abs/1105.5640> (accessed January 4, 2014).

#### **Proceedings di Conferenze Nazionali**

- (aa1). G. Della Penna, B. Intrigila, I. Melatti, E. Tronci, and M. Venturini Zilli. Automatic Analysis of Hybrid Systems with the Murphi Verifier. *Atti Ufficiali del Congresso Annuale dell'Associazione Italiana per l'Informatica ed il Calcolo Automatico (AICA 2005)*

## Software di ricerca prodotto

- QKS (*Quantized Kontrol Synthesizer*, una versione preliminare è disponibile su [http://mclab.di.uniroma1.it/software\\_qks.html](http://mclab.di.uniroma1.it/software_qks.html)) implementa gli algoritmi di sintesi automatica di software di controllo descritti in (c14) e (c12). QKS prende in input la descrizione del sistema da controllare (*plant*) come *Discrete Time Linear Hybrid System*, le caratteristiche della conversione AD/DA che si desidera usare e le specifiche formali del sistema a ciclo chiuso. QKS ritorna in output il software che implementa il controllore quantizzato, soddisfa le specifiche formali a ciclo chiuso ed ha un WCET (*Worst Case Execution Time*) noto e garantito.
- NashMV (una versione preliminare è disponibile su [http://mclab.di.uniroma1.it/software\\_nashmv.html](http://mclab.di.uniroma1.it/software_nashmv.html)). NashMV implementa l'algoritmo descritto in (c18), modificando opportunamente il model checker NuSMV.
- Parallel Murphi (Eddy\_Murphi, disponibile su [http://www.cs.utah.edu/formal\\_verification/software/murphi/eddy\\_murphi/](http://www.cs.utah.edu/formal_verification/software/murphi/eddy_murphi/)). Eddy\_Murphi è una versione parallela (ovvero per *cluster* di computer) del model checker Murphi della Stanford University, ed implementa l'algoritmo descritto in (c23) e (j7) usando le MPI (Message Passing Interface) e i *thread* di Linux/Unix.
- Murphi a 64 bit (CMurphi 5.4.6, disponibile su [http://mclab.di.uniroma1.it/software\\_cmurphi.html](http://mclab.di.uniroma1.it/software_cmurphi.html)). Porting di Murphi per architetture a 64 bits.
- Finite Horizon Probabilistic Murphi (CMurphi 5.4.6, disponibile su [http://mclab.di.uniroma1.it/software\\_cmurphi.html](http://mclab.di.uniroma1.it/software_cmurphi.html)). FHP-Murphi (Finite Horizon Probabilistic Murphi), descritto in (j10), (c27), (c28), (c30), è un model checker per la verifica di proprietà ad orizzonte finito su processi stocastici a tempo discreto. È stato usato per la verifica di protocolli probabilistici e per valutare l'affidabilità di sistemi complessi. FHP-Murphi è in grado di gestire numeri reali a precisione finita, e per questo è stato usato per la verifica di sistemi ibridi stocastici non lineari.
- Caching & Disk Murphi (CMurphi 5.4.6, disponibile su [http://mclab.di.uniroma1.it/software\\_cmurphi.html](http://mclab.di.uniroma1.it/software_cmurphi.html)). CMurphi, descritto in (c25), (j12), (c31), è una versione migliorata del model checker Murphi. CMurphi sfrutta la località delle transizioni di stato nel sistema da verificare per risparmiare sull'uso della memoria RAM e per velocizzare algoritmi di verifica basati su disco. CMurphi è stato molto usato da INTEL per la verifica di Cache Coherence Protocols. Infine, CMurphi è in grado di gestire numeri reali a precisione finita, e per questo è stato usato con successo nella verifica di sistemi ibridi non lineari.



## Progetti di Ricerca

Come membro dell'MCLab (Model Checking Laboratory, gruppo di ricerca del Dipartimento di Informatica della Sapienza Università di Roma coordinato dal prof. Enrico Tronci) è costantemente coinvolto in molti progetti di ricerca finanziati da EC (comunità europea), ESA (ente spaziale europeo), ENEA, CNR, MIUR (Ministero dell'Istruzione e dell'Università), MSE (Ministero dello Sviluppo Economico) ed industrie private. Questo permette una continua osmosi tra i risultati della ricerca ed applicazioni industriali avanzate. Ecco una lista selezionata di progetti (N.B.: nei progetti a partire dal 2006 ha partecipato anche alla fase di preparazione della proposta di progetto):

**WFR (MSE) - 2014** *Web Fitting Room* è un progetto sponsorizzato dal Ministero dello Sviluppo Economico nell'ambito del programma "Industria 2015". L'obiettivo di WFR consiste nel progettare ed implementare un sistema web based per la produzione on-demand di abiti connettendo via Web un Camerino Virtuale (dove gli abiti vengono virtualmente indossati) ad un Decision Support System (DSS) che, in tempo reale, organizza la produzione degli abiti selezionati. Il ruolo di MCLab consiste nella progettazione e realizzazione del DSS usando tecniche di Model Checking based Planning per contrastare il problema dell'esplosione dello spazio degli stati insito nei DSS del tipo illustrato sopra.

**SmartHG (EC FP7) - 2013** *Energy Demand-Aware Open Services for Smart Grid Intelligent Automation* è un progetto finanziato dalla Comunità Europea nell'ambito del programma FP7. Il compito di MCLab in questo progetto è di coordinare il progetto stesso (con consorzio costituito da 11 partner europei), sia dal punto di vista scientifico che amministrativo. Dal punto di vista scientifico, il compito di MCLab è progettare ed implementare dei servizi intelligenti che, raccogliendo dati in tempo reale dalle case degli utenti finali, realizzino due obiettivi principali: minimizzare il costo e l'uso di energia elettrica per ciascuna casa, e supportare l'operatore di rete nell'ottimizzare le operazioni sulla rete elettrica stessa.

**PAEON (EC FP7) - 2013** *Model Driven Computation of Treatments for Infertility Related Endocrinological Diseases* è un progetto finanziato dalla Comunità Europea nell'ambito del programma FP7. Il compito di MCLab in questo progetto è di coordinare il progetto stesso (con consorzio costituito da 5 partner europei), sia dal punto di vista scientifico che amministrativo. Dal punto di vista scientifico, il compito di MCLab è progettare ed implementare delle tecniche per i) predire, sfruttando modelli forniti da altri partner di progetto, il risultato di un trattamento su una specifica paziente affetta da malattie endocrinologiche correlate all'infertilità, e supportare lo sviluppo di un trattamento personalizzato a partire da un trattamento generico; ii) validare modelli e tool sulla base di dati clinici raccolti da altri partner di progetto.

**ULISSE (EC FP7) - 2009 – 2011** *USOCs KnowLedge Integration and Dissemination for Space Science Experimentation* è un progetto finanziato dalla Comunità Europea nell'ambito di FP7. Il compito di MCLab in questo progetto è consistito nel studiare tecniche di model checking per la verifica automatica delle procedure e dei piani relativi agli esperimenti di bordo nelle stazioni orbitanti (ad esempio la stazione Columbus).

**ESA ITI AO6067 - 2010** *Model Checker Validator for Satellite Operational Procedure*. L'obiettivo di questa Innovation Triangle Initiative (ITI) dell'ESA è consistito nel progettare e realizzare un model checker per la Verifica e Validazione automatica delle

Procedure Operative (OP) per satelliti. Il contributo di MCLab al progetto si concentra sulla progettazione del model checker per OP e della sua interfaccia con il simulatore dell'ESA SIMSAT.

**SAPP (FILAS) - 2008** *Sistema Avanzato per la Progettazione e la Pianificazione di Reti Wireless* è un progetto finanziato da FILAS (Finanziaria Laziale di Sviluppo). Il compito di MCLab in questo progetto è consistito nella progettazione e realizzazione di algoritmi per il posizionamento fault tolerant dei nodi relay di una rete wireless dati i modelli radio e le posizioni dei nodi gateway e sensori. Nello specifico si vuole garantire che anche in presenza di  $k$  nodi relay guasti la rete soddisfi le specifiche date.

**ESA 5459 SSFRT - 2008** *System and Software Functional Requirements Techniques* è un progetto finanziato dall'ESA (European Space Agency) con main contractor del progetto l'azienda INTECS. Il compito di MCLab in questo progetto è consistito nell'investigare la possibilità di utilizzare tecniche di model checking per sistemi ibridi per la verifica automatica dei requisiti di sistema per satelliti e veicoli spaziali in genere. La modellazione di tali requisiti coinvolge sia il software che i sistemi (attuatori e sensori) con cui tale software interagisce. Da questo deriva la necessità di usare sistemi ibridi per la modellazione, validazione e verifica.

**SINTESI - 2008** *Sintesi Automatica di Regole di Reazione per la Gestione di Processi Aziendali* è un progetto finanziato dal MIUR. Il ruolo di MCLab in questo progetto è consistito nel progettare e realizzare algoritmi innovativi basati sul model checking per la sintesi automatica di regole di reazione in sistemi SaR (Sense and Respond) per la gestione di processi aziendali, con particolare enfasi al caso dell'allocazione automatica di risorse in imprese operanti nel settore dei multimedia.

**CRESCO (MIUR) - 2007** *Centro Computazionale di RicErca sui Sistemi Complessi*. Il contributo al presente progetto è stato nelle vesti di consulenti per l'Università del Salento. Il contributo di MCLab al progetto è consistito nella progettazione e realizzazione di algoritmi per l'analisi automatica di indicatori di vulnerabilità di reti e servizi di telecomunicazione.

**TRAMP (MIUR) - 2007** *Sistema Integrato di Gestione e Controllo per il TRASporto in Sicurezza di Merci Pericolose* è un progetto finanziato dal MIUR. L'obiettivo del progetto è consistito nell'aumentare la sicurezza del trasporto di materiali pericolosi usando tecnologie informatiche. Il compito di MCLab in questo progetto è di usare tecniche di model checking per verificare la safety dell'intero sistema con riferimento ai protocolli di livello applicativo ed alle politiche di controllo e gestione.

**SETRAM (MIUR - ENEA) - 2006** *Sistema esperto con funzioni di simulazione delle modalità di trasporto merci e di selezione dei percorsi sulla base di multicriteria*. Il contributo di MCLab al presente progetto è stato nelle vesti di consulente ENEA. L'obiettivo del progetto è consistito nel studiare l'applicazione dell'ottimizzazione multimodale a grandi ed esistenti sistemi di trasporto merci. Il coinvolgimento di MCLab in questo progetto è consistito nella progettazione e realizzazione di algoritmi di ottimizzazione per il trasporto multimodale.

**IRRIIS (EC FP6 - ENEA) - 2005** *Integrated Risk Reduction of Information-based Infrastructure*. Il contributo di MCLab al presente progetto Europeo è stato nelle vesti di consulente ENEA. L'obiettivo di questo progetto è stato di definire tecniche di riduzione

del rischio per grandi sistemi di infrastrutture critiche. Il contributo di MCLab al progetto è consistito nel progettare e realizzare algoritmi basati sulla programmazione lineare intera mista per minimizzare l'inoperabilità di infrastrutture critiche.

**SAFEGUARD (EC - ENEA) - 2004** *Intelligent Agents Organization to Enhance Dependability and Survivability of Large Complex Critical Infrastructure*. Il contributo di MCLab al presente progetto Europeo è stato nelle vesti di consulente ENEA. L'obiettivo di questo progetto è stato l'investigazione di metodi per aumentare l'affidabilità di grandi infrastrutture critiche. Il contributo di MCLab al progetto è consistito nel progettare ed implementare un tool basato su catene di Markov per la rivelazione automatica di anomalie nel traffico TCP.

**SAFE TUNNEL (EC - ENEA) - 2003** *Innovative systems and frameworks for enhancing of traffic safety in road tunnels*. Il contributo di MCLab al presente progetto Europeo è stato nelle vesti di consulente ENEA. L'obiettivo di questo progetto è stato di investigare l'utilizzo di tecniche dell'information technology per aumentare la sicurezza all'interno dei trafori alpini. Il contributo di MCLab al progetto è consistito nella verifica della safety dei protocolli di comunicazione e controllo all'interno di tunnel stradali usando tecniche di model checking.

**MEFISTO (MIUR - COFIN) - 2002** *MEtodi FormalI per Sicurezza e TempO*. Il presente progetto è stato un COFIN. L'obiettivo del progetto è stato di investigare come i metodi formali possano essere usati per verificare la sicurezza di sistemi in cui il tempo gioca un ruolo importante. Il contributo di MCLab al progetto è consistito nell'ideazione e realizzazione di algoritmi e tool oltre lo stato dell'arte per la verifica di protocolli probabilistici e sistemi ibridi.

**ICARO (ENEA) - 2001** . Il presente progetto è stato finanziato da ENEA. L'obiettivo di questo progetto è stato l'investigazione di metodi per l'analisi di affidabilità di sistemi di controllo complessi come l'impianto di cogenerazione ICARO in uso all'ENEA di Casaccia. Il contributo di MCLab al progetto è consistito nella modellazione e verifica del sistema di controllo di ICARO usando tecniche di model checking per sistemi ibridi.

**TOSCA (MIUR - COFIN) - 2000** *Teoria della Concorrenza, Linguaggi di Ordine Superiore e Strutture di Tipi* è stato un progetto COFIN. Il contributo di MCLab al progetto è consistito nell'ideazione e realizzazione di algoritmi e tool innovativi per la verifica di sistemi concorrenti.

## **Altre attività di ricerca**

È stato nel comitato organizzatore delle seguenti conferenze:

- CHARME 2005 (Correct Hardware Design and Verification Methods)
- ETAPS 2013 (European Joint Conferences on Theory and Practice of Software)

È nel program committe delle seguenti conferenze:

- ICSEA (Conference on Software Engineering Advances)
- INFOCOMP (International Conference on Advanced Communications and Computation)

Ha servito come revisore peer-to-peer per le seguenti riviste internazionali:

- IEEE Distributed Systems Online
- International Journal of Business Data Communications and Networking

Ha servito e serve come revisore peer-to-peer per le seguenti conferenze internazionali:

- CAV (Computer Aided Verification)
- FMCAD (Formal Methods for Computer-Aided Design)
- PSI (Ershov Informatics Conference)
- ICALP (International Colloquium on Automata, Languages and Programming)
- SAT (Theory and Applications of Satisfiability Testing)
- LICS (Logic In Computer Science)
- CHARME (Correct Hardware Design and Verification Methods).

## **Attività didattica**

### **Ricercatore Universitario**

#### *Didattica Frontale*

- Presso l'Università degli studi di Roma "La Sapienza", Facoltà di Ingegneria dell'Informazione, Informatica e Statistica, Corso di Laurea in Statistica, Economia, Finanza ed Assicurazioni, è titolare del corso di *Informatica* (9 CFU, Anno accademico 2013/2014)
- Presso l'Università degli studi di Roma "La Sapienza", Facoltà di Ingegneria dell'Informazione, Informatica e Statistica, Corso di Laurea in Statistica, Economia, Finanza ed Assicurazioni, è titolare del corso di *Informatica* (9 CFU, Anno accademico 2012/2013)
- Presso l'Università degli studi di Roma "La Sapienza", Facoltà di Ingegneria dell'Informazione, Informatica e Statistica, Corso di Laurea in Statistica, Economia, Finanza ed Assicurazioni, è titolare del corso di *Informatica* (9 CFU, Anno accademico 2011/2012)
- Presso l'Università degli studi di Roma "La Sapienza", Facoltà di Ingegneria dell'Informazione, Informatica e Statistica, Corso di Laurea in Informatica, ha tenuto le esercitazioni per il corso di *Sistemi Operativi (II modulo)* (6 CFU, Anno accademico 2010/2011)

#### *Altre Attività Didattiche*

- È membro della Commissione Attività Extrauniversitarie del Consiglio di Area Didattica in Informatica.
- Ha collaborato all'organizzazione della prima edizione del corso "Innovation Lab" (2011) del Consiglio di Area Didattica in Informatica.
- Ha collaborato all'organizzazione del Tirocinio Formativo Attivo in Informatica, A.A. 2012 – 2013.

### **Assegnista di Ricerca e Post-Doc**

- Presso l'Università degli studi di Roma "La Sapienza", Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto le esercitazioni per il corso di *Progettazione di Sistemi Digitali* (canale E-O, Anno accademico 2008/2009 e 2009/2010)
- Presso l'Università degli studi di Roma "Tor Vergata", Facoltà di Scienze MM. FF. NN., Corso di Laurea in Scienze Biologiche, è stato titolare della parte di Informatica all'interno del corso integrato di *Matematica ed Informatica* (Anno accademico 2008/2009)

- Presso l’Università degli studi di Roma “La Sapienza”, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto parte delle esercitazioni per il corso di *Progettazione di Sistemi Digitali* (canale A-D, Anno accademico 2008/2009)
- Presso l’Università degli studi dell’Aquila, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, è stato titolare del corso di *Verifica dei Sistemi Complessi* (Anno accademico 2007/2008)
- Presso l’Università degli studi di Roma “La Sapienza”, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto le esercitazioni per il corso di *Programmazione 1* (Anni accademici 2006/2007 e 2007/2008)
- Presso l’Università degli studi di Roma “La Sapienza”, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, si è occupato del tutoraggio per il corso di *Architettura degli Elaboratori 1* (Anno accademico 2007/2008)
- Presso l’Università degli studi di Roma “La Sapienza”, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto cicli di seminari all’interno del corso di *Programmazione a Oggetti* (Anno accademico 2006/2007)
- Presso la School of Computing della University of Utah ha tenuto cicli di seminari all’interno del corso di *Model Checking* (CS 6964, Fall 2005) e della serie di seminari *AMPS (Programming Languages and Systems Seminar, CS 7931, Fall 2005)*
- Presso l’Università degli studi di Roma “Tor Vergata”, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto cicli di seminari all’interno del corso di *Metodi Formali per la Verifica dei Sistemi Complessi* (Anno accademico 2005/2006)
- Nell’ambito dei suddetti corsi, ha coadiuvato i professori titolari dei corsi nella supervisione dei progetti e dei compiti d’esame
- Ha seguito lo svolgimento di tesi di laurea presso l’Università di Roma “La Sapienza”, in collaborazione col Prof. Enrico Tronci

### **Durante il dottorato**

- Presso l’Università degli studi dell’Aquila, Facoltà di Scienze MM. FF. NN., Corso di Laurea in Informatica, ha tenuto cicli di seminari all’interno dei seguenti corsi:
  - Anno accademico 2004/2005: *Metodi Formali per la Verifica dei Sistemi Complessi*
  - Anno accademico 2003/2004: *Metodi Formali per la Verifica dei Sistemi Complessi*
  - Anno accademico 2003/2004: *Architettura degli Elaboratori*
  - Anno accademico 2002/2003: *Architettura degli Elaboratori*
  - Anno accademico 2001/2002: *Laboratorio di Architettura degli Elaboratori*

- Nell'ambito dei suddetti corsi, ha coadiuvato i professori titolari nella correzione dei compiti d'esame, negli esami orali e nella supervisione dei progetti d'esame
- Ha seguito lo svolgimento di tesi di laurea presso l'Università dell'Aquila, in collaborazione coi Proff. Benedetto Intrigila, Enrico Tronci e Giuseppe Della Penna

## **Scuole internazionali**

- ISCL 2002: Second International Summer School in Computational Logic (Acquafredda di Maratea, 25-30 agosto 2002)

## **Esperienze lavorative**

- Nel maggio 2006 ha tenuto un corso intensivo di programmazione C++, organizzato dalla Regione Lazio (a Tivoli).
- Tra il 2002 ed il 2005 ha tenuto alcuni corsi intensivi di informatica di base (programmazione, web design), organizzati dalla Regione Abruzzo (a L'Aquila ed Avezzano)
- Nel luglio 2002 ha tenuto un corso intensivo di XML organizzato dalla società BitMedia s.r.l. (a Roma).
- Dal luglio 2001 a marzo 2002 ha lavorato presso la Nikesoft s.r.l., azienda informatica dell'Aquila, occupandosi prevalentemente della realizzazione di sistemi di gestione per le PMI (piccole e medie imprese).



## Descrizione dell'attività di ricerca

L'attività di ricerca di Igor Melatti ha come argomento principale il *Model Checking*. Nell'area del Model Checking si fanno ricadere algoritmi e tools, denominati *model checker*, che ricevono in input la specifica formale di un sistema  $\mathcal{S}$  e di una proprietà  $\phi$ , e producono in output *vero* se  $\phi$  è soddisfatta in  $\mathcal{S}$  e *false* altrimenti. Diversamente da altri approcci, come il testing o la simulazione, se un model checker risponde *vero* allora c'è la *certificazione matematica* che  $\mathcal{S}$  soddisfa  $\phi$ . Per sistemi *mission critical* o *safety critical* questo è richiesto già in fase di progettazione, come si può vedere ad esempio negli standard dell'ESA (European Space Agency) e dell'IEC (International Electrotechnical Commission).

Da un punto di vista computazionale, gli algoritmi di Model Checking si articolano generalmente in tre passi:

1. ottenere il *grafo di transizione* del sistema  $\mathcal{S}$ , che specifica come  $\mathcal{S}$  può transire da uno *stato* ad un altro;
2. calcolare l'insieme degli stati *raggiungibili* a partire da un determinato insieme di stati iniziali (*reachability*);
3. verificare la proprietà  $\phi$  su tutti gli stati raggiungibili.

L'ostruzione più significativa alla verifica tramite Model Checking è localizzabile nella *reachability*. Infatti, anche se la descrizione formale di  $\mathcal{S}$  è ragionevolmente limitata, il numero di stati di  $\mathcal{S}$  risulta esponenziale nella misura della descrizione di  $\mathcal{S}$ , il che porta ad esaurire le risorse disponibili per il calcolo, soprattutto per quanto riguarda la memoria. Questo problema viene comunemente detto *state space explosion*.

Nella sua attività di ricerca, Igor Melatti si è concentrato sulla verifica di alcune classi di sistemi concorrenti di grande interesse, ovvero i protocolli di comunicazione e i sistemi ibridi (caratterizzati tanto da variabili discrete quanto da variabili continue), e su una importante classe di proprietà da verificare, ovvero le proprietà di *safety*.

Più in dettaglio, ha lavorato ai seguenti problemi:

### Algoritmi di verifica basati su proprietà statistiche dei sistemi

In questo ambito, il contributo dato consiste nell'ideazione, realizzazione e sperimentazione di nuovi algoritmi per il Model Checking, in grado di ottenere risultati migliori per occupazione di memoria di quelli attualmente state-of-the-art in questo campo per i sistemi e le proprietà di interesse. L'implementazione di questi nuovi algoritmi è stata incorporata nel tool *Caching Murphi* (CMurphi), ottenuto a partire dal model checker *Murphi* di Stanford. CMurphi si è rivelato particolarmente efficace, al punto da essere usato con successo dall'*Intel* (più precisamente, dal *Platform Architecture Research Team*, nei *Microprocessor Technology Labs* del *Corporate Technology Group* dell'*Intel*), per verificare dei *cache coherence protocols* che non era stato possibile verificare usando gli algoritmi precedenti. A questa linea di ricerca fanno riferimento le pubblicazioni (j6), (j12), (c25) e (c29).

### Algoritmi per la verifica di sistemi stocastici

A partire dalla seconda metà degli anni novanta, dal Model Checking si è sviluppata un'altra categoria di tecniche per la verifica formale, ovvero il *Probabilistic Model Checking*. Da un punto di vista algoritmico, gli algoritmi per il Probabilistic Model Checking prendono in input un sistema *stocastico*  $\mathcal{S}$  e una proprietà  $\phi$  che può contenere espressioni probabilistiche.

Nella sua attività di ricerca su questo argomento, Igor Melatti si è focalizzato su alcune importanti classi di sistemi stocastici, ovvero su protocolli di comunicazione probabilistici e su sistemi stocastici ibridi descrivibili tramite catene di Markov a tempo discreto. Per quanto riguarda le proprietà, mi sono focalizzato su quelle ad orizzonte finito, ovvero che considerano un numero finito di transizioni nel grafo di transizione di  $\mathcal{S}$ .

In questo ambito, il contributo dato consiste nell'ideazione, realizzazione e sperimentazione di un tool per il Probabilistic Model Checking (chiamato *Finite Horizon Probabilistic Murphi*, ovvero FHP-Murphi), in grado di ottenere risultati migliori dei tool attualmente state-of-the-art in questo campo per i sistemi e le proprietà di interesse. A questa linea di ricerca fanno riferimento le pubblicazioni (c27), (c28), (c30) e (j10).

### **Algoritmi paralleli per il Model Checking**

Uno dei metodi tradizionalmente usati per contrastare la state explosion negli algoritmi di Model Checking è quello di progettare algoritmi paralleli, ovvero eseguibili su *cluster* di calcolatori. Così facendo, il problema non è più l'occupazione di memoria, dato che si può contare sulla memoria di più macchine; l'ostruzione maggiore la si rileva invece nel tempo di computazione, dato che gran parte dell'esecuzione viene spesa per la comunicazione tra i vari calcolatori.

Il contributo in questo ambito consiste nell'ideazione, realizzazione e sperimentazione di un nuovo schema di parallelizzazione, implementato nel tool *Eddy\_Murphi*, che ha dato ottimi risultati. A questa linea di ricerca fa riferimento le pubblicazioni (c23) e (j7). Ulteriori migliorie sono attualmente allo studio, in collaborazione col gruppo di lavoro del Prof. G. Gopalakrishnan, della University of Utah.

### **Sintesi automatica di programmi reattivi da specifiche formali**

Dato un modello formale dell'ambiente con cui un programma reattivo deve interagire è possibile usare tecniche di Model Checking per sintetizzare automaticamente il codice C che implementa il programma reattivo stesso. Questo può essere fatto considerando il programma reattivo come un *universal plan*. Questo approccio, oltre al vantaggio della generazione automatica di codice corretto per costruzione, permette anche la verifica e validazione dei requisiti a ciclo chiuso sin dai primi stadi della progettazione. Lavori sull'argomento sono stati pubblicati in (c21), (j8) e (c20).

### **Model Checking e sintesi di controllori di sistemi ibridi**

In questa categoria rientrano i lavori che definiscono una metodologia automatica per la verifica e la sintesi di controllori per sistemi ibridi che siano descrivibili come Discrete Time Linear Hybrid Systems, ovvero modellati da un insieme di vincoli lineari. Lavori su questo argomento sono stati pubblicati in (c14), (c4), (c1), (j1), (j3), (j5), (c2), (c3), (c5), (c6), (c8), (c9) e (c10).

### **Technology transfer**

Per valutare l'efficacia degli approcci proposti in un contesto industriale, le seguenti pubblicazioni si occupano di casi di studio ed attività di technology transfer: (c31), (c22), (c16), (c15) e (j6). Più in particolare, le seguenti pubblicazioni riguardano l'applicazione delle tecniche di Model Checking all'ambito delle missioni spaziali: (c13), (c11) e (c7).

## Verifica di protocolli BAR

I protocolli BAR (Byzantine Altruistic Rational) sono dei sistemi cooperativi (come ad esempio i protocolli peer-to-peer) composti da  $n$  agenti, in cui alcuni agenti possono non rispettare il protocollo stesso. Di questi agenti, alcuni (i *razionali*) sono guidati da un meccanismo di premi e punizioni definito dal protocollo stesso, altri (i *bizantini*) modellano agenti difettosi.

In (c18) viene presentato un algoritmo di model checking simbolico per la verifica di equilibri di Nash per meccanismi a stati finiti modellanti protocolli BAR, ovvero sistemi distribuiti *Multiple Administrative Domains* (MAD).

Dato un meccanismo a stati finiti, un *proposed protocol* per ciascun agente e una *soglia di indifferenza* per i premi, il nostro model checker ritorna *PASS* se il proposed protocol è un equilibrio di Nash (a meno della soglia di indifferenza data) per il meccanismo dato, e *FAIL* altrimenti.

Il lavoro in (c17) estende tale lavoro al caso in cui alcuni agenti possano formare delle coalizioni.

## Altre attività di ricerca

La mia attività di ricerca, infine, comprende anche alcuni lavori non strettamente collegati con il Model Checking, frutto di collaborazioni con gruppi di lavoro dell'Università dell'Aquila. Tali lavori sono descritti nelle pubblicazioni (j9), (c26), (j11) e (c32).