

Curriculum Vitae

Aniello Murano

Università degli Studi di Napoli “Federico II”
Dipartimento di Scienze Fisiche
Sezione di Informatica
Via Cinthia, Complesso Monte S. Angelo,
I-80126 Napoli, ITALY.

August 21, 2008

Personal Information

Date of birth: December 1971
Nationality: Italian
URL: <http://people.na.infn.it/~murano>
E-mail: murano@na.infn.it
Phone: +39 081 679279

Current Position

Assistant Professor in Computer Science at the Computer Science Division of the Department of Scienze Fisiche, Università degli Studi di Napoli “Federico II”, Napoli, Italy.

Research Interests

Theory of automata, formal languages, temporal logics, concurrent and real-time systems, game theory, formal verification, control theory, model checking.

Education

- **May 1999 - October 2002:** PhD Student in Computer Science at the Università di Salerno. Thesis title: *Decision Problems on Tree Automata and Synthesis of Open Timed Systems*. Advisors: Prof. Margherita Napoli, Prof. Moshe Y. Vardi, and Dr. Salvatore La Torre. Defense: February 20, 2003. Final Mark: Summa cum laude.

- **September 1998 - May 1999:** Master Post-Lauream in Multimedia methodologies for tele-didactics (Metodologie Telematico-Multimediali per la didattica) at the Università di Salerno. Final Mark: 100/100.
- **November 1997:** Laurea degree in Computer Science at the Università di Salerno. Thesis title: *Automi Finiti su Oggetti Infiniti*. Advisor: Prof. Margherita Napoli. Final Mark: Summa cum laude
- **July 1990:** High School Diploma in Accounting, Business, and Computer Programming at the I.T.C.G A. Masullo, Nola (NA). Final Mark: 60/60.

Post–doctoral Employment

- **October 2003 – June 2004:** Post–doctoral Researcher position at the Hebrew University - School of Computer Science and Engineering, Gerusalemme, Israele under the supervision of Prof. Orna Kupferman (Grant Support 10,000 Euros).
- **November 2002 – February 2005:** Post–doctoral Researcher position at the Università di Salerno, working on the project *formal methods in system verification* under the MIUR program *scientific research, technological development and higher education/training*. Advisor: Prof. Margherita Napoli. Grant Support: 50,000 Euros.

Visiting positions

- **February 18 – March 15, 2006 and July 23 – August 09, 2008 :** *Visiting Researcher* at the School of Computer Science and Engineering, Hebrew University, Jerusalem 91904, Israel, working under the supervision of Prof. Orna Kupferman (Grant support from the University of Naples 6,000 Euros).
- **January – November, 2001 and February 2002:** *Visiting scholar* at the Department of Computer Science – Rice University - Texas (USA), working under the supervision of Prof. Moshe Y. Vardi – (NSF Grant CCR-9988322 and William Marsh Rice University grant, US\$ 8,000).

Summer Schools and Courses Attended

- **July 10–July 23, 2005:** *17th International School for Computer Science Researchers Lipari School on Formal Methods: Theory And Practice*, organized by the Università di Catania and held in Lipari Island, Sicily, Italy.
- **September - December, 2001:** *Computer-Aided Verification* course (Comp607), lecturer Prof. Moshe Y. Vardi, Rice University Houston, Texas (USA).
- **August 13-24, 2001:** *13th European Summer School in Logic, Language and Information*, at the University of Helsinki, Helsinki, Finland.

- **January - May, 2001:** *Logic in Computer Science* course (Comp409), lecturer Prof. Moshe Y. Vardi, Rice University Houston, Texas (USA).
- **August 6-18, 2000:** *12th European Summer School in Logic, Language and Information*, at the University of Birmingham, Birmingham, United Kingdom.
- **September 11–15, 2000:** *Summer School on Quantum Computing*, at the I.I.A.S.S. (International Institute for Advanced Scientific Studies) of Vietri (SA), Italy.

Talks

- *Pushdown Module Checking with Imperfect Information*,
8 September, 2007,
18th International Conference on Concurrency Theory (CONCUR'07), Lisboa - Portugal.
- **Enriched μ -calculus Module Checking**
26 March, 2007,
10th International Conference on Foundations of Software Science and Computation Structures (FOSSACS'07), Braga, Portugal.
- **Pushdown Module Checking for Branching-time Temporal Logics**
9 October, 2006
Schloss Dagstuhl Seminar 06411: Specification, Verification and Test of Open Systems, Dagstuhl, Germany.
- **The Complexity of Enriched μ -Calculi**
16 July, 2006,
33rd International Colloquium on Automata, Languages and Programming (ICALP'06), S. Servolo - Venezia, Italia.
- **Infinite State Systems Module Checking**
28 February, 2006
CS Theory Seminar, Department of Computer Science and Engineering - Hebrew University, Jerusalem, Israel.
- **Pushdown Module Checking**
5 December, 2005
12th International Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR'05), Montego Bay, Jamaica.
- **Typeness for ω -Regular Automata**
3 November 2004
2nd International Symposium on Automated Technology for Verification and Analysis, Taipei, Taiwan.

- **Reasonings on the Complement of Deterministic Büchi Tree Automata**
24 September 2004
Proc. of First International Colloquium on Theoretical Aspects of Computing, Guiyang, China.
- **Weak Muller Tree Automata**
19 May 2004
Workshop Minerva, Kibbutz Shfayim, Israel.
- **Timed Games with Branching-time Winning Conditions,**
28 April 2004
CS Theory Seminar at the Hebrew University, Jerusalem, Israel.
- **Timed Games with Branching-time Winning Conditions**
18 December, 2003
Seminar at the Computer Science section of the Università di Napoli, Italy.
- **Timed games with branching-time winning conditions,**
22 May, 2003
Workshop on Semantics and Verification of Hardware and Software Systems, SVHSS'03, Tel Aviv University, Israel.
- **Weak Muller Acceptance Conditions for Tree Automata**
January 2002
Third International Workshop on Verification, Model Checking and Abstract Interpretation, VMCAI'02, Venezia, Italy.
- **Weak Muller Acceptance Conditions for Tree Automata**
April 2001
Rice University, Houston (TX - USA)

Conferences and Journals Referee

- Reviewed papers for Journals: *Theoretical Computer Science(TCS)*, *Electronic Notes in Theoretical Computer Science(ENTCS)*, *International Journal on Software Tools for Technology Transfer(STTT)*, *Information and Computation(I&C)*, *Formal methods in System Design(FMSD)*.
- Reviewed papers for conferences: *ASIACCS'07*, *POLICY'07*, *FORMATS'06*, *ICTCS'05*, *CAV'03*, *VMCAI'03*, *VMCAI'02*, *FOSSACS'01*, *ICALP'01*.

Participation to Research Projects

- Università di Salerno, ex 60%, 1998-2000: *Metodi Formali per la Specifica dei Sistemi*.
- Università di Salerno, ex 60%, 1999-2001: *Metodi Formali per la Verifica e la Specifica di Sistemi*.

- MURST project anno 1999-2001: *Theory of Concurrency, Higher Order and Types (TOSCA)*.
- Università di Salerno, ex 60%, 2000-2002: *Metodi Formali ed Algoritmi per la Verifica di Sistemi Distribuiti*.
- National Science Foundation (NSF) project, 2000-2003: *Educational Innovation-Integrating Logic in the Computer Science Curriculum*. Grant n. CCR-9988322. Coordinator Prof. Moshe Y. Vardi
- Università di Salerno, ex 60%, 2001-2003: *Verifica di Sistemi Concorrenti e Gerarchici*.
- MURST project, 2001-2003: *Metodi Formali per la Sicurezza e il Tempo (MEFISTO)*.
- Università di Salerno, ex 60%, 2002-2004: *Verifica di Sistemi Gerarchici e di Sistemi Aperti*.
- Università di Salerno, ex 60%, 2003-2005: *Verifica di Sistemi Real-Time, Gerarchici, e Ricorsivi*.
- Università di Salerno, ex 60%, 2004-2006: *Algoritmi su Grafi e Verifica di Sistemi*.
- National Research MIUR project (PRIN), 2005-2007: *Synthesis of deduction-based decision procedures with applications to the automatic formal analysis of software*.
- National Research MIUR project (FIRB), 2005-2007: *Automatic Verification of Internet Security Protocols*.
- Network of Excellence (NoE) project within the "6th Framework Programme" (FP6) of the European Community, 2004-2008: *"REasoning on the WEb with Rules and SEmantics" (REWERSE)* - subarea: "Policy enforcement, composition, and conformance".
- Università di Salerno, ex 60%, 2004-2006: *Algoritmi su Grafi e Verifica di Sistemi*.
- National research MIUR project (PRIN), 2005-2007: *Synthesis of deduction-based decision procedures with applications to the automatic formal analysis of software*.
- National research MIUR project (FIRB), 2005-2007: *Automatic Verification of Internet Security Protocols*.
- Università di Salerno, ex 60% anno 2006-2008: *Metodi per la Verifica di Sistemi Software e Real-Time*.
- European research project ESF Research Networking Programme 2008-2013: *Games for Design and Verification (GAMES)*.

Organizing Activity

- **From May 2007:** Member of the PhD scientific evaluators group within the doctoral program “Scienze Computazionali e Informatiche” at the Dipartimento di Matematica e Applicazioni “Renato Caccioppoli” of the Università di Napoli “Federico II”.
- **From May 2006:** Member of experts evaluators to assist the European Commission’s services for tasks in connection with the Seventh RTD Framework Programme.
- **From September 2005:** Responsible of the Formal Methods Lab, Università di Napoli “Federico II”.
- **September 2002 - April 2003:** Member of the local organizing committee for *First Workshop Mefisto, Metodi Formali per la Sicurezza e il Tempo*, 31 March - 02 April 2003 - Salerno, Italy.
- Current Phd students:
 - Fabio Mogavero
 - Alessandro Bianco
- Past Phd students:
 - Dario Carotenuto

Teaching Experience

- **From May 2007:** Teaching in *History of the Computer Science* course, at “Corsi Abilitanti” for new high school teachers to be inserted in school as permanent staff, Università di Napoli “Federico II”
- **From May 2006:** Teaching in *Formal methods in open system verification* course, Computer Science Phd Program, Università di Salerno.
- **From September 2005:** Teaching in *Semantics* course, Computer Science Master Degree, Università di Napoli “Federico II”. Starting from September 2008 this course has been also provided in *e-learning*.
- **From March 2005:** Teaching in *Laboratory of Algorithms and Data Structures* course, Computer Science Degree, Università di Napoli “Federico II”. Starting from September 2007 this course has been also provided in *e-learning*.
- **March 2002 - February 2005:** Teaching assistant in *Theory of Computation* and *Automata and Formal Languages* courses. References Prof. Margherita Napoli (lecturer of the course).

Language

Italian (mother tongue), English (fluent), and Spanish (scholastic).

References

- Prof. Margherita Napoli, Università di Salerno, Baronissi(SA) - Italy
- Prof. Orna Kupferman, Hebrew University, Jerusalem, Israel
- Prof. Moshe Y. Vardi, Rice University, Houston - TX (USA)
- Prof. Adriano Peron, Università degli Studi di Napoli “Federico II”, Napoli - Italy

Research Activity

Releasing reliable systems has never been an easy task and the complexity of modern digital systems makes it even harder. In the last years several techniques have been implemented to check for system dependability, and those based on formal methods have received great attention. The key idea is to construct, as intermediate step, an abstract *model* of the system and then formally check it automatically with respect to a formal *specification* (desired properties). The formal analysis of systems relies on the solution of some decision problems such as *model-checking*, *satisfiability*, *games*, *module checking* and *control synthesis*. In the literature, these problems have been studied in a variety of formalisms, on different classes of system models (such as *finite-state systems*, *pushdown systems*, *real time systems*), and with respect to several kinds of properties (such as *automata*, *temporal and real-time logics*).

Automata

Finite Automata turn out to be a very useful formalism in formal verification where they can be used as models for both systems and specifications, as well as tools to solve decision problems. Indeed, many decision and synthesis problems have automata-based solutions and no other solution for them is known. By translating systems and specification to automata, questions about systems and their specifications can be reduced to decision problems in the automata theory. In more details, given a system S and a specification φ , by constructing an automaton A_S corresponding to S and an automaton $A_{\neg\varphi}$ corresponding to the negation of φ , S meets the specification if the language accepted by the intersection of A_S and $A_{\neg\varphi}$ is not empty.

In [J5], [C14], and [I2], we introduce and investigate two new acceptance conditions for automata on infinite trees, i.e., the *Landweber* and the *Muller-Superset* conditions. We prove that the latter induces automata expressively equivalent to Bchi tree automata, but exponentially more succinct. The former, instead, induces automata that are not comparable with Bchi tree automata, but the emptiness problem for them is still decidable in polynomial time. In [C10], we consider *co-Büchi tree automata* along with both *alternating*

and *generalized* paradigms, as a characterization of the class of languages whose complement is accepted by generalized Büchi tree automata. For the considered automata, we prove that the simulation theorem does not hold and the emptiness problem is ExpTime-complete. For the deterministic subclass, instead, we show a polynomial-time algorithm to solve the emptiness problem.

In [C5], we introduce and study the new class of *Visibly Pushdown Automata with two stacks* as a useful model to effectively describe concurrent pushdown systems using a simple communication mechanism between stacks. We show that this automata accept some context-sensitive languages that are not context-free and some context-free languages that are not accepted by any visibly automaton with one stack. We also show that the emptiness problem for this automata is undecidable, while it become solvable in PTime by adding an ordering constraint on stacks.

In [J4] and [C9], we introduce and study several notions of *typeness* for automata on infinite words, which are helpful in investigating the complexity and complication of translations between various classes of automata. For example, we prove that deterministic Büchi automata are co-Büchi-type, that is, the translation (when exists) from a Büchi to a co-Büchi automaton can be performed just redefining the acceptance condition (i.e., the set of accepting states), and thus involving no blow-up. Also, we prove that nondeterministic Büchi automata are not co-Büchi-type and nondeterministic co-Büchi automata are Büchi-powerset type. From the latter, we obtain that a translation of nondeterministic co-Büchi to deterministic Büchi automata, can be done (when it exists) by applying the subset construction.

In [C7] we consider the *fully enriched μ -calculus* logic, which is an extension of the propositional μ -calculus with inverse programs, graded modalities, and nominals, and study the complexity of its fragments obtained by dropping at least one of the additional constructs. We show that, in all fragments obtained in this way, satisfiability is decidable and ExpTime-complete. Our results are obtained by introducing two-way graded alternating parity automata over infinite forests and by reducing the satisfiability to the emptiness problem for this automata. In fact, we show that the emptiness problem is solvable in ExpTime.

Module Checking and Game Theory

In the context of verification of open systems, *games* and *module checking* provide a suitable framework. The interaction of an open system with its environment can be modeled as a game of the system (*protagonist*) against the environment (*antagonist*). Such a game is usually formalized as a game graph, modeling the system, and a winning condition φ , specifying the requirements. Thus, we can perform system verification by deciding if there exists a strategy of the system that selects only behaviors fulfilling φ , independently from the behavior of the environment.

In [C12, C13] we consider timed games where the system is modeled by a timed automaton and the winning condition is given by a discrete [C12] or dense real-time [C13] temporal logic formula. In both cases, we prove that the module checking problem is equivalent to the satisfiability problem, for the corresponding logic considered. In particular, for the branching dense real-time case, we show that the module checking problem is undecidable

and that the undecidability relies on the ability of express punctuality. Indeed, by removing this feature, we show that the problem becomes not harder than the discrete corresponding one.

In many present-day computational systems, the correct functioning of the system depends on the environment in which it is operating. For example, a web service designed for the intranet may not function correctly for the internet, and a car tuned for the Sahara desert may not start in Alaska. Usually, in order to verify or test such a system, one considers the composition of the system with its maximal or worst case environment. To distinguish those system that do not interact with any environment from them interacting with an external environment and whose behavior depend on this interaction, we call the former *closed systems* and the latter *open systems*. For example, an embedded control circuit can be seen as an open system, since it connects with the plant via sensors and actuators. The loop consisting of controller and plant is instead a closed system. The task of verifying an open system (without closing it by its maximal environment) is called *module checking*. Thus, one may think to the module checking problem as the problem of finding winning strategies in a particular two-player game.

In [C6, C8, T1] we consider the module checking problem with respect to systems modeled as open pushdown systems and requirements defined as CTL, CTL* and fully enriched μ -calculus formulas.

Regarding Decision Problems

In [C15] we consider optimization problems for weighted automata. Weighted timed automata extend timed automata with costs on both locations and transitions. In this framework we study the *optimal reachability* and the *optimal control synthesis* problems for automata with acyclic control graphs. This class of automata is relevant for some practical problems such as some static scheduling problems or air-traffic control problems. We prove that the optimal reachability problem is NP-complete, while the control-synthesis problem is solvable in exponential space.

Finally, in [J6] we apply the main idea present in [C15] to model-checking the secure release of a time-locked secret over a network.

Journals

- [J1] P.A. Bonatti, C. Lutz, A. Murano, and M.Y. Vardi. The complexity of enriched μ -calculi. *Logical Methods in Computer science (LMCS 2008)*, To appear:26 pages, 2008.
- [J2] A. Ferrante, A. Murano, and M. Parente. Enriched μ -calculi module checking. *Logical Methods in Computer science (LMCS 2008)*, 4(3:1):1–21, 2008.
- [J3] L. Bozzelli, A. Murano, and A. Peron. Pushdown module checking. *Formal Methods in System Design (FMSD 2008)*, To appear:33 pages, 2008.

- [J4] O. Kupferman, G. Morgenstern, and A. Murano. Typeness for ω -regular automata. *International Journal of Foundations of Computer Science (IJFCS 2006)*, 17(4):869–883, 2006.
- [J5] S. La Torre, A. Murano, and M. Napoli. Weak muller acceptance condition for tree automata. *Theoretical Computer Science (TCS'05)*, 332(1–3):233–250, 2005.
- [J6] S. La Torre, A. Murano, and M. Parente. Model-checking the secure release of a time-locked secret over a network. *Electronic Notes in Theoretical Computer Science (ENTCS 2004)*, 99:229–243, 2004.

Conferences

- [C1] A. Murano. The pushdown module checking saga. In *1th Annual Workshop of the ESF Networking Programme on Games for Design and Verification, (GAMES'08)*. Warsaw, Poland, 8-11 September, 2008.
- [C2] B. Aminof, A. Legay, A. Murano, and O. Serre. μ -calculus pushdown module checking with imperfect state information. In *5th IFIP International Conference on Theoretical Computer Science, (IFIP TCS'08)*, volume 273 of *IFIP*, pages 333–348. Springer-Verlag, 2008. Milano, Italy , 7-10 September, 2008.
- [C3] B. Aminof, A. Murano, and M. Y. Vardi. Pushdown module checking with imperfect information. In *18th International Conference on Concurrency Theory, (CONCUR'07)*, volume 4703 of *LNCS*, pages 461–476. Springer-Verlag, 2007. Lisboa - Portugal, 38 September, 2007.
- [C4] A. Ferrante, A. Murano, and M. Parente. Enriched μ -calculus pushdown module checking. In *14th International Conference on Logic for Programming Artificial Intelligence and Reasoning, (LPAR'07)*, volume 4790 of *LNCS*, pages 438–453. Springer-Verlag, 2007. Yerevan, Armenia, 15-19 October, 2007.
- [C5] D. Carotenuto, A. Murano, and A. Peron. 2-visibly pushdown automata. In *11th International Conference on Development in Language Theory (DLT'07)*, volume 4588 of *LNCS*, pages 132–144. Springer-Verlag, 2007. Turku - Finland, 36 July, 2007.
- [C6] A. Ferrante and A. Murano. Enriched μ -calculus module checking. In *10th International Conference on Foundations of Software Science and Computation Structures, (FOSSACS'07)*, volume 4423 of *LNCS*, pages 183–197. Springer-Verlag, 2007. Braga - Portugal, 24-31 March, 2007.
- [C7] P.A. Bonatti, C. Lutz, A. Murano, and M.Y. Vardi. The complexity of enriched μ -calculi. In *33rd International Colloquium on Automata, Languages and Programming (ICALP'06)*, volume 4052 of *LNCS*, pages 540–551. Springer-Verlag, 2006. S. Servolo, Venice - Italy, July 9 - 16, 2006.

- [C8] L. Bozzelli, A. Murano, and A. Peron. Pushdown module checking. In *12th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR'05)*, pages 504–518, 2005. Montego Bay, Jamaica, December 2-6, 2005.
- [C9] O. Kupferman, G. Morgenstern, and A. Murano. Typeness for ω -regular automata. In *2nd International Symposium on Automated Technology for Verification and Analysis (ATVA'04)*, volume 3299 of *LNCS*, pages 324–333. Springer-Verlag, 2004. Taipei, Taiwan, ROC, October 31-November 3, 2004.
- [C10] S. La Torre and A. Murano. Reasoning about co-büchi tree automata. In *First International Colloquium on Theoretical Aspects of Computing (ICTAC'04)*, volume 3407 of *LNCS*, pages 527–542. Springer-Verlag, 2005. Guiyang, China, September 20-24, 2004.
- [C11] A. Murano. Timed games with branching-time winning conditions. In *1st International Workshop on Semantics and Verification of Hardware and Software Systems (SVHSS'03)*, 2003. Tel Aviv University, Israel, May 20-22, 2003.
- [C12] M. Faella, S. La Torre, and A. Murano. Automata-theoretic decision of timed games. In *Third Workshop in Verification, Model Checking, and Abstract Interpretation (VMCAI'02)*, volume 2294 of *LNCS*, pages 94–108. Springer-Verlag, 2002. Venezia, Italy, January 21 - 22, 2002.
- [C13] M. Faella, S. La Torre, and A. Murano. Dense real-time games. In *Proc. of the Seventeenth Annual IEEE Symposium on Logic in Computer Science (LICS'02)*, pages 167–176. IEEE Computer Society Press, 2002. Copenhagen, Denmark, July 22-25, 2002.
- [C14] S. La Torre, A. Murano, and M. Napoli. Weak muller acceptance condition for tree automata. In *Third Workshop in Verification, Model Checking, and Abstract Interpretation (VMCAI'02)*, volume 2294 of *LNCS*, pages 240–254. Springer-Verlag, 2002. Venezia, Italy, January 21 - 22, 2002.
- [C15] S. La Torre, A. Murano, and S. Mukhopadhyay. Optimal-reachability and control for acyclic weighted timed automata. In *2nd IFIP International Conference on Theoretical Computer Science (IFIP TCS'02)*, volume 223, pages 485–497. Kluwer Academic Publishers, 2002. Montreal, Canada, August 25-30, 2002.
- [C16] A. Murano. Weak muller tree automata. In *International Workshop Minerva on Formal-Verification*, 2004. Kibbutz Shfayim, Israel, May 16-19, 2004.

Invited Talks

- [I1] A. Murano. Pushdown module checking for branching-time temporal logics. Dagstuhl School- Germany, Preprint Ottobre 2006.
- [I2] A. Murano. Weak muller tree automata. In *International Workshop Minerva on Formal-Verification*, 2004. Kibbutz Shfayim, Israel, May 16-19, 2004.

Book Chapters

- [B1] Antonio Iorio and A. Murano. Studio di fattibilità per la realizzazione di una rete informatica oncologica pediatrica. In V.Poggi and M. Fusco, S. Lodato, N. Nante, and A.Murano, editors, *Un network in Oncologia pediatrica: sperimentazione clinica, organizzativa e gestionale di una rete regionale*, pages 40–78, 2008.

Miscellaneous

- [M1] A. Murano, G. Menna, R. Parasole, F. Petruzzello, and V. Poggi. Creazione di una struttura informatica che sottenda una rete oncologica pediatrica regionale. In *XXXV Congresso nazionale AIEOP.*, In stampa su supplemento di un volume della rivista *Haematologica*, Ottobre 2008.

Theses

- [T1] Aniello Murano. *Tecision Problems on Tree Automata and Synthesis of Open Timed Systems*. PhD thesis, Università degli Studi di Salerno, February 2003. Supervisor: M. Napoli, M.Y.Vardi e S. La Torre.
- [T2] Aniello Murano. *Classi di Automi su Alberi Infiniti*. Master's thesis, Università degli Studi di Salerno - Facoltà di Scienze MM.FF.NN - Corso di Laurea in Scienze dell'Informazione, Novembre 1997. Supervisore: M. Napoli.

Technical Reports

- [TR1] A. Murano, M. Napoli, and M. Parente. The program complexity of hierarchical module checking (sottomesso a conferenza). Luglio 2008.
- [TR2] A. Bianco, F. Mogavero, and A. Murano. Graded computation tree logic (sottomesso a conferenza). Report n 21, April 2008.
- [TR3] F. Mogavero and A. Murano. Branching-time temporal logics with minimal model quantifiers (sottomesso a conferenza). Luglio 2008.
- [TR4] D. Carotenuto, A. Murano, and A. Peron. 2-visibly pushdown automata (sottomesso a rivista). Preprint Dicembre 2006.