

UNIVERSITÀ DEGLI STUDI DI MILANO

Procedura di valutazione per la chiamata a professore di II fascia da ricoprire ai sensi dell'art. 24, comma 6, della Legge n. 240/2010 per il settore concorsuale 01/B1 – Informatica settore scientifico-disciplinare INF/01 – Informatica presso il Dipartimento di Informatica “Giovanni Degli Antoni”, Codice concorso 4413.

Alberto Momigliano

Curriculum Vitæ

Informazioni Personali

Cognome Momigliano

Nome Alberto

Data di nascita 29 Ottobre 1960

Posizione attuale ed impieghi precedenti

- *Ricercatore confermato INF/01*, presso il Dipartimento di Informatica, Università degli studi di Milano.
- *Collaboratore alla ricerca* (assegnista) presso il Dipartimento di Scienze dell'Informazione, Università degli studi di Milano nell'ambito del progetto coordinato dal Prof. Mario Ornaghi, Novembre 2003 – Novembre 2007.
- *Part-time Research Fellow* presso il Laboratory for Foundations of Computer Science, School of Informatics, University of Edinburgh, Dicembre 2003 – Dicembre 2007, parzialmente finanziato dai progetti europei *MRG* e *Mobius*.
- *Research Fellow* presso il Laboratory for Foundations of Computer Science, School of Informatics, University of Edinburgh, Luglio 2003 – Novembre 2003, finanziato da *MRG*.
- *Research Assistant* presso il Department of Mathematics and Computer Science, University of Leicester (UK), Luglio 2000–Luglio 2003, finanziata dal grant *EPSRC* “Mechanized Operational Semantics”.
- *Ricercatore* al CEFRIEL, Milano: disegno e sviluppo di un sistema di *semantic-based information retrieval*, coordinato dal Prof. Augusto Celentano, Politecnico di Milano, 1990.
- *Programmatore di ricerca* nel gruppo di Sistemi Esperti, Honeywell Bull, Pregnana Milanese: disegno e sviluppo del sistema di schedulazione di aeromobili *OMAR* (Operative Management of Aircraft Routing) per Alitalia, 1988–1990.

ASN

- Settore Concorsuale 01/B1 Informatica: Seconda Fascia, Bando 2016 (D.D. 1532/2016).
- Settore Concorsuale 01/A1 Logica Matematica e Matematiche Complementari: Prima Fascia, Bando 2018 (DD n. 1532/2016).
- Settore Concorsuale 01/A1 Logica Matematica e Matematiche Complementari: Seconda Fascia, Bando 2012 (DD n. 222/2012).

Istruzione

- Ph.D. in Pure and Applied Logic, Carnegie Mellon University. Titolo della tesi: “Elimination of Negation in a Logical Framework”. Supervisore Frank Pfenning, membri del comitato di tesi: Dale Miller, Dana Scott.
- M.S. in Logic and Computation, Carnegie Mellon University. Titolo della tesi: “Some Remarks on Uniform Proofs and Constructive Negation”. Supervisore Wilfried Sieg.
- M.Sc. in Computation, Oxford University. Titolo della tesi: “Prolog and Negation”, supervisore C.A.R. Hoare.
- Laurea in Filosofia, Università degli Studi di Milano con votazione 110/110 e lode. Titolo della tesi: “Implicazione e Deducibilità nei Sistemi Formali”. Relatore Corrado Mangione, correlatore Andrea Bonomi.

Partecipazione a progetti di ricerca

- Responsabile unico:
 - “Estensioni del Property-based Testing di e con linguaggi di programmazione dichiarativa”. Progetto di ricerca GNCS 2020.
 - “Certificazione di verificatori automatici del software basati su clausole di Horn con vincoli”. Progetto di ricerca GNCS 2017.
- Partecipante:
 - “METALLIC #2: METodi di prova per il ragionamento Automatico per Logiche non-classiche”. Progetto di ricerca GNCS 2019.
 - “Metodi di prova orientati al ragionamento automatico per logiche non-classiche”. Progetto di ricerca GNCS 2018.
 - PRIN 2012: “Metodi logici per il trattamento dell’informazione”. Unità di Milano.
 - MOBIUS (mobius.inria.fr): “Mobility, Ubiquity and Security. Enabling proof-carrying code for Java on mobile devices”. Mobius is a European Integrated Project developing novel technologies for trustworthy global computing, using proof-carrying code to give users independent guarantees of the safety and security of Java applications for their mobile phones and PDAs (Research fellow; 2005–2007).
 - PRIN 2006. “Potenziamento e Applicazioni della Programmazione Logica Disgiuntiva” Unità della Università degli Studi di Messina.
 - MRG (groups.inf.ed.ac.uk/mrg): “Mobile Resource Guarantees”. The project has developed the infrastructure needed to endow mobile code with independently verifiable certificates describing its resource behaviour (space, time, etc.). Funded under the Global Computing pro-active initiative of the Future and Emerging Technologies part of the Information Society Technologies programme of the European Commission’s Fifth Framework Programme (Research fellow; 2003–2005).
 - MOS: “Mechanized Operational Semantics”, EPSRC, PI: Roy Crole and Simon Ambler (Research assistant, 2000–2003).
 - TWELF (<http://twelf.plparty.org/>): a research project concerned with the design, implementation, and application of logical frameworks funded by the NSG and DARPA under grants Meta-Logical Frameworks, Efficient Logical Frameworks, PI: Frank Pfenning (Ph.D student; 1994–2000).
- Partecipazione non-strutturata:

- “A Higher-Order Abstract Syntax Approach to Reasoning about Programs and Programming Languages”, NSERC (Natural Sciences and Engineering Research Council of Canada) Discovery Grant. PI: Amy Felty, April 2015 – ongoing. (hybrid.di.unimi.it).

Attività didattiche

- Corsi triennali al DI:
 - Programmazione Dichiarativa, 2020 (6 CFU).
 - Laboratorio di Logica: 2018 e 2019 (3 CFU).
 - Programmazione Funzionale: 2012–2019 (6 CFU). (3 CFU, triennale).
- Corsi magistrali al DI:
 - Metodi Formali: 2014, 2015, 2016 e 2019 (6 CFU).
 - Verifica e Convalida del Software: 2017 (1.5 CFU).
- Corsi di servizio: Laboratorio di Informatica per le Scienze Biologiche: 2011–2013 (3 CFU, triennale).
- Corsi di dottorato al DI:
 - *From Lightweight Validation to Formal Certification* (12 ore), Novembre 2015.
 - *Parallel Haskell on Multi-Cores and Clusters* (4 ore), Giugno 2015.
 - *Linear Logic* (6 ore), Aprile 2011.
 - *Proof Search and Computation* (10 ore), Marzo 2010.
- Altre lezioni:
 - D(S)I: Assistenza alle ore di laboratorio del corso di *Intelligenza Artificiale 1* e lezioni nel corso di *Metodi Formali dell'Informatica*, in collaborazione col Prof. Mario Ornaghi.
 - Lezioni nel corso *Fondamenti logico-matematici dell'informatica* in collaborazione con Prof. Ugo Moscato, Università degli Studi di Milano-Bicocca, 2008.
 - Coordinatore del *Semantics Seminar*, University of Leicester (2001–2003):
 - * *Co-inductive Techniques in Relational Semantics*.
 - * *Operational and Denotational Semantics for Languages with Variable Bindings*.
 - Teaching assistant del Prof. Bob Harper per il corso *Principles of Programming Languages*, Carnegie Mellon University, 2000.
 - Docente del corso fondamentale del primo anno *Introduction to Logic*, Carnegie Mellon University, 1996–99.
- Sono stato relatore di circa due dozzine di tesi triennali, la metà della quali come tirocinio interno, nonché di quattro tesi magistrali (Pessina, Fachini, Komauli, Marabelli), tutte valutate con il massimo dei voti.
- Sono stato membro del Collegio di Dottorato in Informatica dalla costituzione fino al 2017.

Seminari ed interventi su invito¹

- “An overview of PBT for the working semanticist”. King’s College London and University of Sussex, 6/2018.
- “Property-Based Testing PL Artifacts: An experience report”. Department of Computer Science and Engineering, Chalmers University, 5/2017.
- “Benchmarks for mechanized meta-theory: A very personal and partial view”. Dagstuhl Seminar on *Universality of proofs*, 10/2016.
- “Toward a Theory of Contexts of Assumptions in Logical Frameworks.” Meeting on *Theory and Application of Formal Proof*, Laboratoire d’Informatique de École Polytechnique, 11/2013.
- “A HOAS Encoding of Howe’s Method”. Department of Theoretical Computer Science, IT University of Copenhagen, 3/2012, École Polytechnique, Parigi, 7/2012.
- “Mechanized metatheory model-checking”. Department of Theoretical Computer Science, IT University of Copenhagen, 8/2007.
- “A Practical Approach to Co-induction in Twelf”. Comète-Parsifal Seminar, École Polytechnique, Paris, 6/2006.
- “A Program Logic for Resources and its Application to Optimisation Validation”. Queen Mary’s College, University of London, 12/2005.
- “Automatic Certification of Resource Consumption”. Comète-Parsifal Seminar, École Polytechnique, Paris, 5/2005.
- “Induction and Co-Induction in Sequent Calculus”. Heriot-Watt University, Edinburgh, 5/2003.
- “A Hybrid Logical Framework”. INRIA Futurs, Orsay Cedex, 3/2003.
- “Simple Compiler Verification in a Hybrid Logical Framework”. University of Nottingham Theory Seminar, 2/2002.
- “A Definitional Approach to Higher-Order Abstract Syntax”. Yale University, Carnegie Mellon University & Penn State University, 5/2002.
- “A Fresh Look to Uniform Proof Search”. Dagstuhl Seminar on *Semantics Foundations of Proof-search*, 4/2001.
- “Think Positive! Or Elimination of Negation in a Logical Framework”. Joint Theory Seminar, Queen Mary’s College, University of London, 11/2000.

Servizi alla comunità scientifica

- (Co)Chair:
 - *Trentunesimo Convegno Italiano di Logica Computazionale*, Milano, 2016.
 - *LFMTP 2013: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *ICFP’13*, ACM SIGPLAN International Conference on Functional Programming, Boston, 2013.
 - *LFMTP 2006: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *FLOC 2006*, Federated Logic Conference, Seattle, Washington, 2006.

¹Escludendo presentazioni a convegni

- *MERLIN 2005: MEchanized Reasoning about Languages with variable bInding and Names*, in associazione con *ICFP'05*, ACM SIGPLAN International Conference on Functional Programming, Tallin, Estonia, 2005.
- Tutorials Chair di *ETAPS 2005*, European Joint Conferences on Theory and Practice of Software, Edinburgh, 2005.
- *MERLIN 2003: MEchanized Reasoning about Languages with variable bInding and Names*, in associazione con *PLI'03*, Principles, Logics, and Implementations of High-Level Programming Languages, Uppsala, Svezia, 2003.
- *MERLIN 2001: MEchanized Reasoning about Languages with variable bInding and Names*, in associazione con *IJCAR 2001*, International Joint Conference on Automated Reasoning, Siena, 2001.
- *Post-Conference Workshop on Proof-Theoretical Extensions of Logic Programming*, in associazione con *ICLP'94*, International Conference on Logic Programming, Santa Margherita Ligure, 1994.
- *Membro del comitato di programma di:*
 - *LSFA 2020: 15th International Workshop on Logical and Semantic Frameworks, with Applications*, Salvador, Brasile, 2020.
 - *LSFA 2019: 14th International Workshop on Logical and Semantic Frameworks, with Applications*, in associazione con *CADE-27*, Natal, Brasile, 2019.
 - *CCP 2018: 7th International Conference on Certified Programs and Proofs*, Los Angeles, USA, 2018.
 - *LFMTP 2015: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *CADE-25*, Berlin, 2015.
 - *ITP 2012: Interactive Theorem Proving*, Princeton, USA, 2012.
 - *LFMTP 2009: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *CADE-22*, Montreal, Canada, 2009.
 - *LFMTP 2008: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *LICS'08*, Pittsburgh, USA, 2008.
 - *ICTCS 2007: Italian Conference in Theoretical Computer Science*, Roma, 2007.
 - *LFMTP 2006: Logical Frameworks and Metalanguages. Theory and Applications*, in associazione con *FLOC'06*, Seattle, 2006.
 - *MERLIN 2005: MEchanized Reasoning about Languages with variable bInding and Names*, in associazione con *ICFP 2005*, Tallin, Estonia, 2005.
 - *PPDP 2003, 5th International Conference on Principles and Practices of Declarative Programming Languages*, Uppsala, Svezia, 2003.
- *Steering Committee: LFMTP, Logical Frameworks and Meta Languages: Theory and Practice*. 2015–ongoing.
- *Peer reviews:*
 - Riviste: Theoretical Computer Science, Journal of Automated Reasoning, Journal of Higher Order and Symbolic Computation, Journal of Functional Programming, Annals of Pure and Applied Logic, Journal of Formalized Reasoning, Information Processing Letters, Data & Knowledge Engineering Journal. . .
 - Conferenze: POPL, LICS, ICFP, CSL, ICALP, APLAS, ESOP, PPDP, ICLP, LOPSTR, TABLEAUX, . . .

Riconoscimenti

- Vincitore della borsa di studio del CNR per corsi di perfezionamento all'estero in Scienze Matematiche, bando 203.01.62 del 18/5/93, secondo in graduatoria con votazione 49.90/50.
- Vincitore del “Best Paper Award”, 1st International Conference on Practical Applications of Prolog, London, 1992.
- “Invited panelist” alla discussione su “Prolog in the real world” concludente la 20th *Joint International Conference on Logic Programming (ICLP'92)*, Washington, 1992.

Publicazioni

Articoli su invito

3. June Andronick, Gilles Barthe, Gerwin Klein, Robbert Krebbers, Alberto Momigliano, Brigitte Pientka, Natarajan Shankar & René Thiemann. Applications in Computer Science. In: Jasmin Blanchette & Assia Mahboubi (Eds.), *Proof Assistants and Their Applications in Mathematics and Computer Science*. Progress in Computer Science and Applied Logic, Springer. Forthcoming.
2. Alberto Momigliano & Mario Ornaghi. Proof-theoretic and Higher-order Extensions of Logic Programming. In: *A 25-Year Perspective on Logic Programming: Achievements of the Italian Association for Logic Programming, GULP*. Lecture Notes in Computer Science 6125 Springer 2010.
1. Alberto Momigliano & Lennart Beringer. Certification of Resource Consumption: from Types to Logic (Programming). *The Association for Logic Programming Newsletter*, Vol. 18, No. 2, May 2005.

Articoli in riviste internazionali

13. Andreas Abel, Guillam Allais, Aliya Hameer, Alberto Momigliano, Brigitte Pientka, Steven Schaefer & Kathrin Stark. POPLMark Reloaded: Mechanizing Proofs by Logical Relations. *Journal of Functional Programming* 29: e19 (2019).
12. Alberto Momigliano, Brigitte Pientka & David Thibodeau. A Case Study in Programming Coinductive Proofs in Beluga: Howe’s Method. *Mathematical Structures in Computer Science*, 29(8): 1309–1343 (2019).
11. Mauro Ferrari, Camillo Fiorentini & Alberto Momigliano. From Constructivism to Logic Programming: an Homage to Mario Ornaghi. *Fundamenta Informaticae*, 161 (1–2): 1–7 (2018).
10. Amy Felty, Alberto Momigliano & Brigitte Pientka. Benchmarks for Reasoning with Syntax Trees Containing Binders and Contexts of Assumptions. *Mathematical Structures in Computer Science* 28(9): 1507–1540 (2018).
9. James Cheney & Alberto Momigliano. α Check: a mechanized metatheory model-checker. *Theory and Practice of Logic Programming*, 17(3): 311–352 (2017).
8. Alessandro Avellone, Camillo Fiorentini & Alberto Momigliano. A Semantical Analysis of Focusing and Contraction in Intuitionistic Logic. *Fundamenta Informaticæ*, 140(3–4): 247–262 (2015).
7. Amy Felty, Alberto Momigliano & Brigitte Pientka. The Next 700 Challenge Problems for Reasoning with Higher-Order Abstract Syntax Representations. Part 2: A Survey. *Journal of Automated Reasoning*, 55(4): 307–372 (2015).
6. Alwen Tiu & Alberto Momigliano. Cut Elimination for a Logic with Induction and Co-induction. *Journal of Applied Logic*, 10(4): 330–367 (2012).

5. Amy Felty & Alberto Momigliano. Hybrid: A Definitional Two-Level Approach to Reasoning with Higher-Order Abstract Syntax. *Journal of Automated Reasoning* 48(1): 43–105 (2012).
4. David Aspinall, Lennart Beringer, Martin Hofmann, Hans-Wolfgang Loidl, & Alberto Momigliano. A Program Logic for Resourcees, *Theoretical Computer Science*, 389(3): 411–445 (2007).
3. Alberto Momigliano & Frank Pfenning. Higher-Order Pattern Complement and the Strict Lambda-Calculus. *ACM Transactions on Computational Logic*, 4(4): 493–529 (2003).
2. Alberto Momigliano & Mario Ornaghi. Regular Search Spaces and Constructive Negation. *Journal of Logic and Computation*, 7(3): 367–403 (1997).
1. Massimo Paltrinieri, Alberto Momigliano & Franco Torquati. Aircraft Routing as Constraints Satisfaction. *International Journal of Expert Systems: Research & Applications*, 8(4): 349–373 (1995).

Articoli in conferenze internazionali²

27. Roberto Blanco, Dale Miller & Alberto Momigliano. Property-Based Testing via Proof Reconstruction. In: *ACM-SIGPLAN International Conference on Principles and Practice of Declarative Programming (PPDP 2019)*, ACM Press, 1–13,
26. Guglielmo Fachini & Alberto Momigliano. Validating the Meta-Theory of Programming Languages. In: *Software Engineering and Formal Methods 15th International Conference (SEFM 2017)*. Lecture Notes in Computer Science, 10469, 2017: 367–374.
25. James Cheney, Alberto Momigliano & Matteo Pessina: Advances in Property-Based Testing for α Prolog \checkmark . In: *Tests and Proofs 10th International Conference (TAP 2016)*, Lecture Notes in Computer Science, 9762, 2016: 37–56
24. Camillo Fiorentini, Alberto Momigliano, Mario Ornaghi & Iman Poernomo: A Constructive Approach to Testing Model Transformations. In: *Theory and Practice of Model Transformations, Third International Conference (ICMT 2010)*, Lecture Notes in Computer Science 6142, Springer 2010, 77–92.
23. Amy Felty & Alberto Momigliano. Reasoning with Hypothetical Judgments and Open Terms in Hybrid. In: *Proceedings of the 11th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP 2009)*, ACM Press, Sept. 2009, 83–92.
22. Camillo Fiorentini, Mario Ornaghi, Alberto Momigliano & Francesco Pagano. Applying ASP to UML Model Validation. In: *Tenth International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'09)*, Lecture Notes in Computer Science, 5753, Springer-Verlag, 2009, 457–463.
21. Camillo Fiorentini, Mario Ornaghi & Alberto Momigliano. Towards a type discipline for Answer Set Programming \checkmark . In: *Types for Proofs and Programs (TYPES 2008)*, Series: Lecture Notes in Computer Science, Vol 5497, 2009, 117–135.
20. Mauro Ferrari, Camillo Fiorentini, Mario Ornaghi & Alberto Momigliano. Snapshot generation in a constructive object-oriented modeling language. In: *Proceedings of of the Logic Based Program Synthesis and Transformation 14th International Symposium (LOPSTR 2007)*, Revised Selected Papers. Series: Lecture Notes in Computer Science, 4915, Springer-Verlag, 2008, 169–184.
19. James Cheney & Alberto Momigliano. Mechanized MetaTheory Model Checking. In: *Proceedings of the 9th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP 2007)*, ACM Press, 2007, 75–86.

²Presentazioni annotate da \checkmark

18. Donald Sannella, Martin Hofmann, David Aspinall, Stephen Gilmore, Ian Stark, Lennart Beringer, Hans-Wolfgang Loidl, Kenneth MacKenzie, Alberto Momigliano, Olha Shkaravska: Mobile Resource Guarantees (project evaluation paper). In: *Revised Selected Papers from the Sixth Symposium on Trends in Functional Programming (TFP 2005)*, Trends in Functional Programming 6 Intellect 2007, 211–226.
17. Kung-Kiu Lau, Alberto Momigliano & Mario Ornaghi. Constructive Specifications for Compositional Units. In: *Proceedings of the Logic Based Program Synthesis and Transformation 14th International Symposium (LOPSTR 2004)*, Series: Lecture Notes in Computer Science, 3618, Springer-Verlag, 2005, 198–214.
16. Lennart Beringer, Martin Hofmann, Alberto Momigliano & Olha. Shkaravska. Automatic Certification of Heap Consumption \checkmark . In: *Proceedings of the 11th International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR2004)*, Lecture Notes in Artificial Intelligence 3452, Springer-Verlag, 2005, 347–362.
15. D. Aspinall, L. Beringer, M. Hofmann, H-W. Loidl & Alberto Momigliano. A Program Logic for Resource Verification. In: *Proceedings of the 17th International Conference on Theorem Proving in Higher Order Logics (TPHOLs2004)*, Lecture Notes in Computer Science 3223, Springer-Verlag, 34–49.
14. Alberto Momigliano & Alwen Tiu. Induction and Co-induction in Sequent Calculus \checkmark . In: *Types for Proofs and Programs International Workshop (TYPES 2003)*, Revised Selected Papers Series, Lecture Notes in Computer Science, 3085, 2004, 293–308
13. Simon Ambler, Roy Crole & Alberto Momigliano. A Combinator and Presheaf Topos Model for Primitive Recursion over Higher Order Abstract Syntax. In: *Collegium Logicum (Proceedings of the Kurt Godel Society) vol. VIII*, Computer Science Logic/8th Kurt Godel Colloquium, 2003, 83–91.
12. Alberto Momigliano & Simon Ambler. Multi-Level Meta-Reasoning with Higher Order Abstract Syntax \checkmark . In: *Foundations of Software Science and Computational Structures 6th International Conference (FOSSACS 2003)*, Lecture Notes in Computer Science, 2620, 2003, 375–391.
11. Simon Ambler, Roy Crole & Alberto Momigliano. Combining Higher Order Abstract Syntax with Tactical Theorem Proving and (Co)Induction. In: *15th International Conference on Theorem Proving in Higher Order Logics (TPHOLs02)*, Springer Verlag, Lecture Notes in Computer Science 2342, 2002, 327–343,
10. Alberto Momigliano, Simon Ambler & Roy Crole. A Comparison of Formalizations of the Meta-Theory of a Language with Variable Binding in Isabelle \checkmark . In: *14th International Conference on Theorem Proving in Higher Order Logics (TPHOLs01)*, Supplemental Proceedings, 3–6 September 2001, Edinburgh, Scotland, 2001, 267–282.
9. Alberto Momigliano. Elimination of Negation in a Logical Framework \checkmark . In: *CSL 2000, 14th Annual Conference of the European Association for Computer Science Logic (EACSL)*, Springer Verlag, Lecture Notes in Computer Science 1852, 2000, 411–426.
8. Alberto Momigliano & Frank Pfenning. The Relative Complement Problem for Higher-Order Patterns \checkmark . In: *Logic Programming, Proceedings of the 1999 International Conference on Logic Programming (ICLP'99)*, The MIT Press, Cambridge, MT, 1999, 380–394.
7. Alberto Momigliano & Mario Ornaghi. Towards a Logic for Reasoning about Logic Programs Transformation. In: *Logic Programming Synthesis and Transformation, 7th International Workshop (LOPSTR'97)*, Springer Verlag, Lecture Notes in Computer Science 1463, 1998, 226–244.
6. Alberto Momigliano & Mario Ornaghi. An Introduction to Regular Search Spaces \checkmark . In: *Proceedings of the Joint Conference on Declarative Programming (GULP-PRODE'95)*, 1995, 183–194.

5. Alberto Momigliano & Mario Ornaghi. Regular Search Spaces as a Foundation of Logic Programming. In: *Extensions of Logic Programming, Proceedings of the 4th International Workshop (ELP'93)*, Springer Verlag, Lecture Notes in Artificial Intelligence 798, 1994, 222–254.
4. Alberto Momigliano. Minimal Negation and Hereditary Harrop Formulae \checkmark . In: *Logical Foundations of Computer Science (Tver '92): Proceedings of Second International Symposium*, Springer-Verlag, Lecture Notes in Computer Science 620, 1992, 326–335.
3. Massimo Paltrinieri, Alberto Momigliano & Franco Torquati. Scheduling of an Aircraft Fleet. In: *Practical Approaches to Scheduling and Planning: Papers from the 1992 Spring Symposium*, AAAI Press, 1992, 25–29.
2. Alberto Momigliano, Massimo Paltrinieri & Franco Torquati. A Scheduling System for an Aircraft Fleet. In: *Practical Application of Prolog, Proceedings of the International Conference & Exhibition*, 1992, 13 pages.
1. Franco Torquati, Massimo Paltrinieri & Alberto Momigliano. A Constraint Satisfaction Approach to Operative Management of Aircraft Routing. *Proceedings of the Third International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems (IEA/AIE 90)*, ACM Press, 1990, 1140–1146.

Articoli in riviste elettroniche con *peer reviewing*

4. Alberto Momigliano, Alan Martin & Amy Felty. Two-Level Hybrid: A System for Reasoning Using Higher-Order Abstract Syntax. *Electr. Notes Theor. Comput. Sci. (ENTCS)* 196:85–93 (2008).
3. Alberto Momigliano, David Aspinall & Lennart Berlinger. Optimisation Validation. *Electr. Notes Theor. Comput. Sci. (ENTCS)* 176(3):37–59 (2007).
2. Mario Ornaghi, Marco Benini, Mauro Ferrari, Camillo Fiorentini & Alberto Momigliano. A Constructive Modeling Language for Object Oriented Information Systems. *Electr. Notes Theor. Comput. Sci. (ENTCS)* 153(1):55–75 (2006).
1. Alberto Momigliano, Simon Ambler & Roy Crole. A Hybrid Encoding of Howe's Method for Establishing Congruence of Bisimilarity. *Electr. Notes Theor. Comput. Sci. (ENTCS)* 70(2) (2002).

Articoli in librerie elettroniche con *peer reviewing*

4. Amy Felty, Alberto Momigliano & Brigitte Pientka. An Open Challenge Problem Repository for Systems Supporting Binders. *Proceedings of the seventh international workshop on Logical frameworks and meta-languages, theory and practice (LFMTP 2015)*, EPTCS, 18–32,
3. Alberto Momigliano. A supposedly fun thing I may have to do again: a HOAS encoding of Howe's method \checkmark . In *Proceedings of the seventh international workshop on Logical frameworks and meta-languages, theory and practice (LFMTP 2012)*. ACM DL, 33–42.
2. A. Momigliano & Jeff Polakow. A Formalisation of an Ordered Logical Framework in Hybrid with Applications to Continuation Machines \checkmark . *Proceedings of Eighth ACM SIGPLAN International Conference on Functional Programming, Workshop on Mechanized reasoning about languages with variable binding, (MERLIN 2003)*. ACM DL, 1–9,
1. Simon Ambler, Roy Crole & A. Momigliano. A Definitional Approach to Primitive Recursion over Higher Order Abstract Syntax \checkmark . *Proceedings of Eighth ACM SIGPLAN International Conference on Functional Programming, Workshop on Mechanized reasoning about languages with variable binding, (MERLIN 2003)*. ACM DL, 1–11.

Articoli in conferenze nazionali con *peer reviewing*

9. Alberto Momigliano. Why Proof-Theory Matters in Specification-based Testing ✓. *ICTCS'20*, to appear in CEUR.
8. Giorgio Marabelli & Alberto Momigliano. Formalizing the Meta-theory of Program Equivalences in Coq ✓. *ICTCS'19*:204–209. CEUR Workshop Proceedings Volume 2504.
7. Alberto Momigliano & Mario Ornaghi. The Blame Game for Property-based Testing ✓. *CILC 2019*:4–13. CEUR Workshop Proceedings Volume 2396.
6. Francesco Komauli & Alberto Momigliano. Property-testing of Abstract Machines: an Experience Report. *CILC 2018*:22–39. CEUR Workshop Proceedings 2214.
5. Mario Ornaghi, Camillo Fiorentini & Alberto Momigliano. LOGI: A Didactic Tool for a Beginners' Course in Logic (System Description). *ICTCS-CILC*: 341–345, CEUR Workshop Proceedings 1949.
4. Alessandro Avellone, Camillo Fiorentini & Alberto Momigliano: Focusing on Contraction. *CILC 2013*: 65–81, CEUR Workshop Proceedings 1068.
3. Camillo Fiorentini, Alberto Momigliano & Mario Ornaghi. Towards Introducing Types in DLV*. *CILC 2008*.
2. Mauro Ferrari, Camillo Fiorentini, Alberto Momigliano & Mario Ornaghi. Towards Proof-theoretic Model Generation. *CILC 2007*.
1. Alberto Momigliano. Uniform Proofs Systems with Constructive Negation ✓. *GULP'92*:3–17, Città Studi Editore, Milano, 1992.

Articoli in workshop con *peer reviewing* e *proceedings* informali

3. Mario Ornaghi, Camillo Fiorentini and Alberto Momigliano. Snapshots Gneration via Constructive Logic. In *MoVeLog'05, Mobile Code Safety and Program Verification Using Computational Logic Tools*. An *ICLP* Workshop, Sitges, Spain, 10/2005, <http://babel.lis.fi.upm.es/movelog05>.
2. L. Beringer, M. Hofmann, A. Momigliano & O. Shkaravska. Towards Certification Generation for Linear Heap Consumption. In: Didier Galmiche & David Pym (Eds.): *Proceedings of the ICALP Workshop "Logics for Resources, Processes, and Programs" (LRPP 2004)*, 7/2004.
1. Alberto Momigliano & Mario Ornaghi. A Proof-Theoretic Reconstruction of Logic Programming. In: F. Pfenning, J. Harland & D. Pym (Eds.), *Proceeding of the Post-Conference Workshop on Proofs and Types*, in association with *1992 Joint International Conference and Symposium on Logic Programming*, 11/1992, Washington DC, 1992.

Edizioni di numeri di riviste e atti di convegni

8. Camillo Fiorentini, Alberto Momigliano & Alberto Pettorossi. *Special Issue of the 31th Italian Conference on Computational Logic (CILC 2016)*. Fundamenta Informaticæ, vol. 161 (2018).
7. Camillo Fiorentini & Alberto Momigliano: *Proceedings of the 31st Italian Conference on Computational Logic*, Milano, Italy, 2016. CEUR, Vol-1645.
6. Alberto Momigliano & Brigitte Pientka & Randy Pollack. *Proceedings of LFMTTP 2013: Logical Frameworks and Metalanguages. Theory and Applications*. ACM Press & ACM Digital Library.
5. Alberto Momigliano & Brigitte Pientka. *Proceedings of LFMTTP 2006: Logical Frameworks and Metalanguages. Theory and Applications*. Electr. Notes Theor. Comput. Sci. (ENTCS) 174(5):1–2 (2007).

4. Alberto Momigliano & Randy Pollack. *Proceedings of MERLIN 2005: MEchanized Reasoning about Languages with variable bINDing*, ACM Press & ACM Digital Library.
3. Marino Miculan, Furio Honsell & Alberto Momigliano. *Proceedings of MERLIN 2003: MEchanized Reasoning about Languages with variable bINDing*, ACM Press & ACM Digital Library, 2003.
2. Simon Ambler, Roy Crole & Alberto Momigliano. *Proceedings of MERLIN 2001: MEchanized Reasoning about Languages with variable bINDing*, Electronic Notes in Theoretical Computer Science, (ENTCS), Volume 58, Issue 1, November 2001.
1. Alberto Momigliano & Mario Ornaghi. *Proceedings of the Post-Conference Workshop on Proof-Theoretical Extensions of Logic Programming*, held in connection with *1994 International Conference on Logic Programming (ICLP'94)*, Santa Margherita Ligure, Giugno 1994.

Tesi di dottorato

A. Momigliano. Elimination of Negation in a Logical Framework. Ph.D. thesis, disponibile come *Technical Report CMU-CS-00-175*, School of Computer Science, Carnegie Mellon University, August 2000, 160 pagine.

Itinerario scientifico

La mia ricerca si colloca all'intersezione di programmazione logica e teoria dei tipi, in relazione al loro utilizzo per:

- lo studio dei principi di base della progettazione e implementazione di linguaggi di programmazione;
- la specifica, validazione e verifica formale di sistemi software.

Dal secondo anno di università mi sono concentrato sulla logica. La mia tesi di laurea verteva su argomenti di logica delle rilevanza, poi ripresa in modo alquanto diverso nella tesi di dottorato.

Mi sono poi indirizzato verso l'applicabilità della logica in informatica teorica. Durante il "Master of Science" alla Oxford University ho iniziato ad occuparmi di programmazione logica. Sotto la supervisione di Tony Hoare, la mia tesi di Master discuteva varie semantiche per il Prolog con negazione. Questo argomento ha costituito un tema ricorrente della mia ricerca per gli anni a venire.

Ho acquisito una esperienza di programmazione a livello industriale nel gruppo di sistemi esperti della Honeywell Bull di Pregnana, dove ho collaborato al disegno e sviluppo di un sistema di schedulazione di aeromobili e manutenzioni per Alitalia. Si trattava, ai tempi, del più sostanziale esempio di sistema "knowledge-based", scritto in Prolog e basato su una architettura di "constraints satisfaction".

Questa esperienza mi ha ispirato un più vivo interesse per la teoria dei linguaggi di programmazione perseguito nel dottorato in logica pura ed applicata alla Carnegie Mellon University. Da allora ho adottato la teoria della dimostrazione come strumento principale. Il mio primo risultato è stato l'immersione dell'intera logica classica in una estensione del Prolog nota come *sistema di prove uniformi*. Successivamente, in collaborazione con Mario Ornaghi, abbiamo introdotto la nozione di *spazio di ricerca regolare* come una fondazione "proof-theoretic" della programmazione logica.

Nella mia tesi di dottorato ho dato una soluzione al problema, aperto da quasi un ventennio, della negazione in linguaggi di programmazione logica di ordine superiore basati sulle formule Harrop ereditarie.

La nozione di negazione in programmazione logica ha stretti collegamenti con la teoria delle *definizioni (co)induttive*, in particolare alla questione della compatibilità tra (co)induzione e la *sintassi astratta di ordine superiore* (SAOS). Questa è una teoria dichiarativa degli alberi sintattici in presenza di operatori di legame quali astrazione e quantificatori. SAOS porta in teoria dei tipi a definizioni ricorsive non-monotone, rendendo problematica la formulazione di opportuni principi di (co)induzione e quindi il loro utilizzo nella verifica formale. Questa ricerca è iniziata nel progetto EPSRC "Mechanized Operational Semantics" ed è confluita nel successivo assegno di ricerca presso il DSI. Un primo risultato è *Hybrid*, un sistema che permette

di ragionare (co)induttivamente con SAOS in “proof assistants” come Isabelle/HOL e Coq, con applicazioni nella verifica di compilatori e di macchine astratte. *Hybrid* è tuttora in utilizzo e ha influenzato la teoria e la pratica della verifica formale di modelli di linguaggi di programmazione. Nello stesso ambito, abbiamo presentato un calcolo a sequenti dotato di (co)induzione compatibile con SAOS che è ora alla base di sistemi quali il model-checker *Bedwyr* e il proof assistant *Abella*. Questo temi sono alla base della serie di workshop MERLIN, poi esteso in ambito e rinominato *LFMTP*, di cui sono stato uno dei fondatori ed organizzatori .

Un’applicazione particolarmente interessante della teoria dei tipi è la certificazione del codice mobile, in particolare in architetture di *proof-carrying code* (PCC). In qualità di Research Fellow a LFCS Edinburgh ho collaborato ai progetti europei *MRG* e *Mobius*, che si proponevano di applicare PCC alla questione della certificazione dell’uso di risorse, quali, per esempio, “heap space”. Ho collaborato allo sviluppo e verifica (in Isabelle/HOL) di una nuova logica di programma alla Hoare per la Java virtual machine dedicata alla verifica di risorse, nonché a una metodologia per l’inferenza automatica di certificati di consumo di risorse da tipi opportuni.

Negli ultimi anni i miei interessi si sono concentrati su due temi principali:

1. La formulazione di “benchmarks” nel campo dei modelli di linguaggi di programmazione, in modo da favorire la valutazione e lo sviluppo di nuovi strumenti per la verifica formale di proprietà degli stessi, quali la normalizzazione (forte), la bisimulazione e più in generale la meta-teoria dell’equivalenza tra programmi.
2. Il passaggio dalla richiesta monolitica di verifica formale delle proprietà di un artefatto, con i ben noti alti costi che detta verifica comporta, a quella della sua *validazione*, in particolare con tecniche di generazione di contro-modelli e di *property-based testing* (PBT). Questi approcci sono, almeno in principio, “push-button” e possono essere utilizzati nella fase iniziale di design per individuare con sforzo minore i banchi più superficiali. Segnalo qui il disegno, lo sviluppo e la sperimentazione di *α Check*, uno strumento di PBT basato sulla logica nominale.

La teoria della dimostrazione si è rivelata uno strumento fondamentale anche nel campo del testing e più in generale del model-checking; essa fornisce un approccio unificante a nozioni provenienti da ambiti alquanto diversi quali la programmazione funzionale e l’ingegneria del software, e apre la strada a generalizzazioni che permettono di gestire la validazione di proprietà di sistemi infiniti, concorrenti e “stateful”.

Milano, 14/09/2020.