GENNARO PARLATO   CURRICULUM VITAE   (August 15, 2021)

CONTACT INFORMATION

Computer Science Division
Department of Bioscience and Territory
University of Molise
Contrada Fonte Lappone
86090 - Pesche (IS), Italy

Work: +39 0874 404 176
Italy mobile: +39 338 90 100 62

E-mail: gennaro.parlato@unimol.it
Web: http://gennaro-parlato.github.io/

PERSONAL

Name:	Gennaro
Last Name:	Parlato
Citizenship:	British and Italian

EDUCATION

Doctor of Philosophy (Ph.D.) in Computer Science   *11 April 2006*
Università degli Studi di Salerno, Italy.
Advisor: Prof. Salvatore La Torre
Thesis title: *On the Model-Checking of Hierarchical and Recursive State Machines*

Laurea Degree (M.Sc. equivalent) in Computer Science   *21 March 2002*
Università degli Studi di Salerno, Italy.
Thesis title: *Raggiungibilitá e Scoperta dei Cicli in Macchine a Stati Gerarchiche (in Italian)*
Advisor: Prof. Margherita Napoli
Grade: 110/110 summa cum laude

ACADEMIC POSITIONS

Associate Professor, University of Molise   *Oct 2018–present*
Computer Science division,
Department of Biosciences and Territory,
University of Molise, Italy.

Associate Professor, University of Southampton   *Oct 2016–Sept 2018*
Cyber Physical Systems Research Group,
School of Electronics and Computer Science,
University of Southampton, UK.

Lecturer in Automated Verification, University of Southampton   *May 2011–Sept 2016*
Electronic & Software Systems Research Group,
School of Electronics and Computer Science,
University of Southampton, UK.

Postdoctoral Researcher, LIAFA, CNRS, University of Paris 7   *Apr 2010–Apr 2011*
Laboratoire d'Informatique Algorithmique: Fondements et Applications
(with Prof. Ahmed Bouajjani)
Paris, France.

Postdoctoral Researcher, University of Illinois at Urbana-Champaign   *May 2006–Mar 2010*
Dept. of Computer Science
(with Prof. P. Madhusudan)
Urbana, USA.

Postdoctoral Researcher, Università degli Studi di Salerno   *Jan 2007–Dec 2009*
Dept. of Computer Science
(with Prof. Margherita Napoli)
Fisciano, Italy.

Postdoctoral Researcher, Università degli Studi di Salerno   *Feb 2006–Aug 2006*
Dept. of Computer Science
(with Prof. Margherita Napoli)
Fisciano, Italy.

AREAS OF RESEARCH

- Program analysis, Testing and Verification
- Model Checking
- Analysis of Concurrent Software
- Learning
- Logics, and Automata Theory
- Computer Security and Access Control

CURRENT RESEARCH PROJECTS

- Model checking abstractions of concurrent programs using fixed-point (GETAFIX)
- Decidable automata with auxiliary storage
- Sequentialization for the analysis of concurrent programs (CSEQ)
- Verification of programs running under weak memory models
- Decidable logics for programs with data structures and data (STRAND)
- Verification of access control models (VAC)
- Distributed algorithms for automatic analysis and verification (SWARM)
- Automatic verification of heap-manipulating programs (BLOB)

SOFTWARE/TOOLS DEVELOPED

- CSEQ – Sequentialization Tools for Concurrent C Programs
  https://github.com/CSeq/Overview
- GETAFIX – A Symbolic Model-checker for Recursive Programs.
  http://madhu.cs.illinois.edu/getafix/
- STRAND – Decidable logics/SMT solver for reasoning with heaps.
  https://engineering.purdue.edu/~xqiu/strand/index.html
- VAC – Verifier of Access Control
  http://users.ecs.soton.ac.uk/gp4/VAC.html
- PAC – Program Analysis in the Clouds

HONORS AND AWARDS

- Gold medal at "Competition on Software Verification" (SV-COMP'21), concurrency category
  Tool: Lazy-CSeq
- Gold medal at "Competition on Software Verification" (SV-COMP'20), concurrency category
  Tool: Lazy-CSeq
- Silver medal at "Competition on Software Verification" (SV-COMP'18), concurrency category
  Tool: Lazy-CSeq
- Silver medal at "Competition on Software Verification" (SV-COMP'17), concurrency category
  Tool: Lazy-CSeq-Abs

- Bronze medal at "Competition on Software Verification" (SV-COMP'17), concurrency category
  Tool: Lazy-CSeq-Swarm
- Gold medal at "Competition on Software Verification" (SV-COMP'16), concurrency category
  Tool: MU-CSeq-0.4
- Silver medal at "Competition on Software Verification" (SV-COMP'16), concurrency category
  Tool: Lazy-CSeq-1.0
- Gold medal at "Competition on Software Verification" (SV-COMP'15), concurrency category
  Tool: Lazy-CSeq
- Silver medal at "Competition on Software Verification" (SV-COMP'15), concurrency category
  Tool: MU-CSeq
- Gold medal at "Competition on Software Verification" (SV-COMP'14), concurrency category
  Tool: Lazy-CSeq
- Silver medal at "Competition on Software Verification" (SV-COMP'14), concurrency category
  Tool: MU-CSeq
- Silver medal at "Competition on Software Verification" (SV-COMP'13), concurrency category
  Tool: CSeq
- Fellow of The Higher Education Academy (UKSPF Descriptor 2) [March 2014]
- Teaching qualification in France for Maitre de Conferences in Computer Science [February 2011]

PUBLICATIONS

Citations: 1693 (Google Scholar, **h-index**: 24)
Publications are available online at `http://gennaro-parlato.github.io/`

THESES

[T1] **On the Model Checking of Hierarchical and Recursive State Machines.** Dissertation (Ph.D.), Università degli Studi di Salerno, Italy, April 2006.

[T2] **Raggiungibiltá e Scoperta dei Cicli in Macchine a Stati Gerarchiche (in italian).** Laurea degree's thesis, Università degli Studi di Salerno, Italy, March 2002.

CONFERENCE PUBLICATIONS

[1] **On the Model-Checking of Branching-time Temporal Logic with BDI Modalities.** *In Proc. 19th International Conference on Autonomous Agents and Multiagent Systems (**AAMAS'20**), pages 681–689, Auckland, New Zealand, May 2020* (S. La Torre, and G. Parlato).

[2] **A Fixed-point Model-checker for BDI Logics over Finite-state Worlds.** *In Proc. 2nd Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis (**OVERLAY'20**), volume 2785 of CEUR Workshop Proceedings, pages 11–16, Bolzano, Italy, Sept 2020* (S. La Torre, and G. Parlato).

[3] **VeriSmart 2.0: Swarm-Based Bug-Finding for Multi-threaded Programs with Lazy-CSeq.** *In Proc. 34th IEEE/ACM International Conference on Automated Software Engineering (**ASE'19**), pages 1150–1153, San Diego, CA, USA, Nov 2019* (B. Fischer, S. La Torre, and G. Parlato).

[4]   **Model Checking BDI Logics over Finite-state Worlds.** *In Proc. 1st Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis (***OVERLAY'19***), volume 2509 of CEUR Workshop Proceedings*, pages 11–16, Rende, Italy, Nov 2019 (S. La Torre, and G. Parlato).

[5]   **Parallel Bug-finding in Concurrent Programs via Reduced Interleaving Instances.** *In Proc. 32nd IEEE/ACM International Conference on Automated Software Engineering (***ASE'17***)*, pages 753–764, to appear, Urbana-Champaign, IL, USA, Oct-Nov 2017 (T. L. Nguyen, P. Schrammel, B. Fischer, S. La Torre, and G. Parlato).

[6]   **Using Shared Memory Abstractions to Design Eager Sequentializations for Weak Memory Models.** *In Proc. 15rd International Conference on Software Engineering and Formal Methods (***SEFM'17***), volume 10469 of LNCS*, pages 185–202, Trento, Italy, September 2017 (E. Tomasco, T. L. Nguyen, B. Fischer, S. La Torre, and G. Parlato).

[7]   **Preventing Unauthorized Data Flows.** *In Proc. Annual IFIP WG 11.3 Conference on Data and Applications Security and Privacy (***DBSec'17***), volume 10359 of* pages 41–62, Philadelphia, PA, USA, July 2017 (E. Uzun, G. Parlato, V. Atluri, A.L. Ferrara, J. Vaidya, S. Sural, and D. Lorenzi).

[8]   **Lazy-CSeq 2.0: Combining Lazy Sequentialization with Abstract Interpretation - (Competition Contribution).** *In Proc. 23rd International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'17***), volume 10206 of LNCS*, pages 375–379, Uppsala, Sweden, April 2017 (T. L. Nguyen, O. Inverso, B. Fischer, S. La Torre, and G. Parlato).

[9]   **Concurrent Program Verification with Lazy Sequentialization and Interval Analysis.** *In Proc. 5th International Conference on Networked Systems (***NETYS'17***), volume 10299 of LNCS*, pages 255–271, Marrakech, Morocco, May 2017 (T. L. Nguyen, B. Fischer, S. La Torre, and G. Parlato).

[10]   **Lazy Sequentialization for TSO and PSO via Shared Memory Abstractions.** *In Proc. 16th International Conference on Formal Methods in Computer-Aided Design (***FMCAD'16***)*, pages 193–200, IEEE, Mountain View, CA, USA, Oct. 2016 (E. Tomasco, T. L. Nguyen, O. Inverso, B. Fischer, S. La Torre, and G. Parlato).

[11]   **Lazy Sequentialization for the Safety Verification of Unbounded Concurrent Programs.** *In Proc. 14th International Symposium on Automated Technology for Verification and Analysis (***ATVA'16***), volume 9938 of LNCS*, pages 174–191, Chiba, Japan, Oct. 2016 (T. L. Nguyen, B. Fischer, S. La Torre, and G. Parlato).

[12]   **MU-CSeq 0.4: Individual Memory Location Unwindings (Competition Contribution).** *In Proc. 22nd International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'16***), volume 9636 of LNCS*, pages 938–941, Eindhoven, The Netherlands, April 2016 (E. Tomasco, T. L. Nguyen, O. Inverso, B. Fischer, S. La Torre, and G. Parlato).

[13]   **Lazy-CSeq: A Context-Bounded Model Checking Tool for Multi-Threaded C-Programs (Tool Demonstration).** *In Proc. 30th IEEE/ACM International Conference on Automated Software Engineering (***ASE'15***)*, pages 807–812, Lincoln, Nebraska, USA, November 2015 (O. Inverso, T. L. Nguyen, B. Fischer, S. La Torre, and G. Parlato).

[14]   **Verifying Concurrent Programs by Memory Unwinding.** *In Proc. 21st International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'15***), volume 9035 of LNCS*, pages 551–565, London, UK, April 2015 (E. Tomasco, O. Inverso, B. Fischer, S. La Torre, and G. Parlato).

[15]   **MU-CSeq 0.3: Sequentialization by Read-implicit and Coarse-grained Memory Unwindings (Competition Contribution).** *In Proc. 21st International Conference on Tools and Algorithms for the*

Construction and Analysis of Systems (**TACAS'15**)*, volume 9035 of *LNCS*, pages 436–438, London, UK, April 2015 (E. Tomasco, O. Inverso, B. Fischer, S. La Torre, and G. Parlato).

[16]   **Unbounded Lazy-CSeq: A Lazy Sequentialization Tool for C Programs with Unbounded Context Switches (Competition Contribution).** *In Proc. 21st International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'15**)*, volume 9035 of *LNCS*, pages 461–463, London, UK, April 2015 (T. L. Nguyen, B. Fischer, S. La Torre, and G. Parlato).

[17]   **Bounded Model Checking of Multi-Threaded C Programs via Lazy Sequentialization.** *In Proc. 26th International Conference on Computer Aided Verification (***CAV'14**)*, volume 8559 of *LNCS*, pages 584–601, Vienna, Austria, July 2014 (O. Inverso, E. Tomasco, S. La Torre, B. Fischer, and G. Parlato).

[18]   **VAC – Verifier of Administrative Role-based Access Control Policies.** *In Proc. 26th International Conference on Computer Aided Verification (***CAV'14**)*, volume 8559 of *LNCS*, pages 182–190, Vienna, Austria, July 2014 (A. L. Ferrara, P. Madhusudan, T. L. Nguyen, and G. Parlato).

[19]   **Context-Bounded Analysis of TSO Systems.** *In Special issue of LNCS dedicated to Professor Joseph Sifakis*, volume 8415 of *LNCS*, Grenoble, France, April 2014 (F. M. Atig, A. Bouajjani, and G. Parlato).

[20]   **Lazy-CSeq: A Lazy Sequentialization Tool for C (Competition Contribution).** *In Proc. 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'14**)*, volume 8413 of *LNCS*, pages 398–401, Grenoble, France, April 2014 (O. Inverso, E. Tomasco, S. La Torre, B. Fischer, and G. Parlato).

[21]   **MU-CSeq: Sequentialization of C Programs by Shared Memory Unwindings (Competition Contribution).** *In Proc. 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'14**)*, volume 8413 of *LNCS*, pages 398–401, Grenoble, France, April 2014 (E. Tomasco, O. Inverso, S. La Torre, B. Fischer, and G. Parlato).

[22]   **A Unifying Approach for Multistack Pushdown Automata.** *In Proc. 39th International Symposium on Mathematical Foundations of Computer Science (***MFCS'14**)*, volume 8634 of *LNCS*, pages 377–389, Budapest, Hungary, August 2014 (S. La Torre, M. Napoli, and G. Parlato).

[23]   **Scope-Bounded Pushdown Languages.** *In Proc. 18th International Conference on Developments in Language Theory (***DLT'14**)*, volume 8633 of *LNCS*, pages 116–128, Ekaterinburg, Russia, July 2014 (S. La Torre, M. Napoli, and G. Parlato).

[24]   **On the Path-Width of Integer Linear Programming.** *In Proc. 5th International Symposium on Games, Automata, Logics and Formal Verification (***GandALF'14**)*, volume 8633 of *EPTCS*, pages 74–87, Verona, Italy, September 2014 (C. Enea, P. Habermehl, O. Inverso, and G. Parlato).

[25]   **CSeq: A Concurrency Pre-Processor for Sequential C Verification Tools.** *In Proc. 28th IEEE/ACM International Conference on Automated Software Engineering (***ASE'13**)*, IEEE, pages 710-713, Palo Alto, CA, USA, November 2013 (B. Fischer, O. Inverso, and G. Parlato).

[26]   **Quantified Data Automata on Skinny Trees: An Abstract Domain for Lists.** *In Proc. 20th Static Analysis Symposium (***SAS'13**)*, volume 7935 of *LNCS*, pages 172–193, Seattle, WA, USA, June 2013 (P. Garg, P. Madhusudan, and G. Parlato.).

[27]   **Policy Analysis for Self-Administrated Role-Based Access Control.** *In Proc. 19th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (***TACAS'13**)*, volume 7795 of *LNCS*, pages 432–447, Rome, Italy, March 2013 (A. L. Ferrara, P. Madhusudan, and G. Parlato).

[28]   **CSeq: A Sequentialization Tool for C (Competition Contribution).** *In Proc. 19th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (**TACAS'13**), volume 7795 of LNCS, pages 616–618, Rome, Italy, March 2013 (B. Fischer, O. Inverso, and G. Parlato).*

[29]   **Scope-bounded Multistack Pushdown Systems: Fixed-Point, Sequentialization, and Tree-Width.** *In Proc. 32nd Foundations of Software Technology and Theoretical Computer Science conference (**FSTTCS'12**), IIIT Hyderabad, India, December 2012 (S. La Torre, and G. Parlato).*

[30]   **Security Analysis of Access Control Policies through Program Verification.** *In Proc. 25th IEEE Computer Security Foundations Symposium (**CSF'12**), pages 113–125, Harvard University, Cambridge MA, USA, June 2012 (A. L. Ferrara, P. Madhusudan, and G. Parlato).*

[31]   **Analyzing Temporal Role Based Access Control Models.** *In Proc. 17th ACM Symposium on Access Control Models and Technologies (**SACMAT'12**), pages 177–186, Rutgers University, Newark NJ, USA, June 2012 (E. Uzun, V. Atluri, S. Sural, J. Vaidya, G. Parlato, A. L. Ferrara, P. Madhusudan).*

[32]   **Sequentializing Parameterized Programs.** *In Proc. 4th International Workshop on Foundations of Interface Technologies (**FIT'12**), Tallinn, Estonia, March 2012 (S. La Torre, P. Madhusudan, and G. Parlato).*

[33]   **On Sequentializing Concurrent Programs.** *In Proc. 18th International Static Analysis Symposium (**SAS'11**), volume 6887 of LNCS, pages 129–145, Venice, Italy, September 2011 (A. Bouajjani, M. Emmi, and G. Parlato).*

[34]   **Getting Rid of Store-Buffers in the Analysis of Weak Memory Models.** *In Proc. 23rd International Conference on Computer Aided Verification (**CAV'11**), volume 6806 of LNCS, pages 99–115, Cliff Lodge, Snowbird, Utah, USA, July 2011 (M. F. Atig, A. Bouajjani, and G. Parlato).*

[35]   **Decidable Logics Combining Heap Structures and Data.** *In Proc. 38th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (**POPL'11**), pages 611–622, Austin, USA, January 2011 (X. Qiu, G. Parlato, and P. Madhusudan).*

[36]   **The Tree Width of Automata with Auxiliary Storage.** *In Proc. 38th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (**POPL'11**), pages 283–294, Austin, USA, January 2011 (P. Madhusudan, and G. Parlato).*

[37]   **A Tabu Search Heuristic Based on k-Diamonds for the Weighted Feedback Vertex Set Problem.** *In Proc. Network Optimization - 5th International Conference (**INOC'11**), volume 6701 of LNCS, pages 589–602, Hamburg, Germany, June 2011 (F. Carrabs, R. Cerulli, M. Gentili, and G. Parlato).*

[38]   **Model-Checking Parameterized Concurrent Programs Using Linear Interfaces.** *In Proc. 22nd International Conference on Computer Aided Verification (**CAV'10**), volume 6174 of LNCS, pages 629–644, Edinburgh, UK, June 2010 (S. La Torre, P. Madhusudan, and G. Parlato).*

[39]   **The Language Theory of Bounded Context-Switching.** *In Proc. 9th Latin American Theoretical Informatics Symposium (**LATIN'10**), volume 6034 of LNCS, pages 96–107, Oaxaca, Mexico, April 2010 (S. La Torre, P. Madhusudan, and G. Parlato).*

[40]   **Reducing Context-bounded Concurrent Reachability to Sequential Reachability.** *In Proc. 21st International Conference on Computer Aided Verification (**CAV'09**), volume 5643 of LNCS, pages 477–492, Grenoble, France, June 2009 (S. La Torre, P. Madhusudan, and G. Parlato).*

[41]   **Analyzing Recursive Programs using Fixed-point Calculus.** *In Proc. International 30th Conference on Programming Language Design and Implementation (**PLDI'09**), ACM, pages 211–222, Dublin, Ireland, June 2009 (S. La Torre, P. Madhusudan, and G. Parlato).*

[42] **An Infinite Automaton Characterization of Double Exponential Time.** *In Proc. 17th EACSL Annual Conference on Computer Science Logic (*CSL'08*), volume 5213 of LNCS, pages 33–48, Bertinoro, Italy, September 2008 (S. La Torre, P. Madhusudan, and G. Parlato).*

[43] **Context-Bounded Analysis of Concurrent Queue Systems.** *In Proc. 14th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (*TACAS'08*), volume 4963 of LNCS, pages 299–314, Budapest, Hungary, March 2008 (S. La Torre, P. Madhusudan, and G. Parlato).*

[44] **A Robust Class of Context-Sensitive Languages.** *In Proc. 18th Annual IEEE Symp. on Logic in Computer Science, (*LICS '07*), pages 161–170, IEEE, Wroclaw, Poland, July 2007 (S. La Torre, P. Madhusudan, and G. Parlato).*

[45] **On the Complexity of LTL Model-Checking of Recursive State Machines.** *In Proc. 34th Int'l Coll. on Automata, Languages and Programming (*ICALP'07*), volume 4596 of LNCS, pages 937–948, Wroclaw, Poland, July 2007 (S. La Torre, and G. Parlato).*

[46] **Verification of Succinct Hierarchical State Machines.** *1st Int'l Conference on Language and Automata Theory and Applications (*LATA'07*), Tarragona, Spain, March-April 2007 (S. La Torre, M. Napoli, M. Parente, and G. Parlato).*

[47] **Improvements for Truthful Mechanisms with Verifiable One-Parameter Selfish Agents.** *In Proc. 3rd Workshop on Approximation and Online Algorithms (*WAOA'05*), volume 3879 of LNCS, pages 147–160, Palma de Mallorca, Spain, October 2005 (A. Ferrante, G. Parlato, F. Sorrentino, and C. Ventre).*

[48] **A Linear Time Algorithm for the Minimum Weighted Feedback Vertex Set on Diamonds.** *3rd Cologne Twente Workshop on Graphs and Combinatorial Optimization (*CTW'04*), Loveno di Menaggio, Como, Italy, May-June 2004 (F. Carrabs, R. Cerulli, M. Gentili, and G. Parlato).*

[49] **Hierarchical and Recursive State Machines with Context-Dependent Properties.** *In Proc. 30th Int'l Coll. on Automata, Languages and Programming (*ICALP'03*), volume 2719 of LNCS, pages 776–789, Eindhoven, The Netherlands, June-July 2003 (S. La Torre, M. Napoli, M. Parente, and G. Parlato).*

JOURNAL ARTICLES

[50] **Bounded Verification of Multi-Threaded Programs via Lazy Sequentialization.** *ACM Transactions on Programming Languages and Systems (TOPLAS), to appear* (O. Inverso, E. Tomasco, S. La Torre, B. Fischer, and G. Parlato).

[51] **Reachability of scope-bounded multistack pushdown systems.** *Information and Computation, vol. 275, 2020, pages 1–25.* (S. La Torre, M. Napoli, and G. Parlato).

[52] **On the Path-Width of Integer Linear Programming.** *Information and Computation, vol. 253, Apr. 2017, pages 257–271.* (C. Enea, P. Habermehl, O. Inverso, and G. Parlato).

[53] **Scope-Bounded Pushdown Languages.** *International Journal of Foundations of Computer Science (IJFCS), vol. 27, n.2, Feb. 2016, pages 215-234.* (S. La Torre, M. Napoli, and G. Parlato).

[54] **Security Analysis for Temporal Role Based Access Control.** *Journal of Computer Security (JCS), vol. 22, n.6, 22 Dec. 2014, pages 961-996.* (E. Uzun, V. Atluri, S. Sural, J. Vaidya, A. L. Ferrara,, G. Parlato and P. Madhusudan).

[55] **Fast Payment Schemes for Truthful Mechanisms with Verification.** *Theor. Comput.Sci. (TCS), vol. 410, n. 8-10, March, 2009, pages 886-899* (A. Ferrante, G. Parlato, F. Sorrentino, and C. Ventre).

[56] **Verification of Scope-Dependent Hierarchical State Machines.** Information and Computation, vol. 206, n. 9-10, Sept-Oct, 2008, pages 1161-1177 (S. La Torre, M. Napoli, M. Parente, and G. Parlato).

[57] **A Linear Time Algorithm for the Minimum Weighted Feedback Vertex Set on Diamonds.** *Information Processing Letters, vol. 94, n. 1, Apr. 15, 2005, pages 29-35* (F. Carrabs, R. Cerulli, M. Gentili, and G. Parlato).

[58] **Minimum Weighted Feedback Vertex Set on Diamonds.** *Electronic Notes in Discrete Mathematics, vol. 17, n. 1, Oct. 20, 2004, pages 87-91* (F. Carrabs, R. Cerulli, M. Gentili, and G. Parlato).

INVITED LECTURES

Parallel Bug-finding in Concurrent Programs Via Reduced Interleaving Instances
Guest talk @ Amazon,
9 Sept 2020, streaming on Amazon Chime

Finding Rare Concurrent Programming Bugs
Workshop on Verification of Distributed Systems (VDS)
19-21 June, Marrakesh, Morocco

Finding Rare Concurrent Programming Bugs: An Automatic, Symbolic, Randomized, and Parallelizable Approach
15th International Colloquium on Theoretical Aspects of Computing (ICTAC),
12-19 October 2018, Stellenbosch, South Africa

A Pragmatic Bug-finding Approach for Concurrent Programs
Institut de Recherche en Informatique Fondamentale (IRIF),
Paris, France, November 24, 2016

Security Analysis of Self-Administrated Role-Based Access Control through Program Verification
Information Security Group, Department of Computer Science, University College London (UCL),
London, UK, July 2, 2015

On BMC Sequentializations of Concurrent Programs
Computer Science Department, University of Oxford,
Oxford, UK, June 18, 2015

On Sequentializing Concurrent Programs
UPMARC 7th Summer School on Multicore Computing,
Uppsala University, Sweden, June 8-10, 2015
http://www.it.uu.se/research/upmarc/events/SS2015/ss2015/Start.html

Bounded Model Checking of Multi-Threaded C Programs via Lazy Sequentialization
Laboratoire d'Informatique Algorithmique: Fondements et Applications (LIAFA),
Paris, France, May 2014

The Tree-width of Decidable Problems
FSTTCS post-conference workshop on "Verification of Infinite-State Systems", Hyderabad, India,
December, 2012

Sequentializing Concurrent Programs
Computer Science Department, University of Oxford,
Oxford, UK, January 17, 2012 `http://www.cs.ox.ac.uk/seminars/699.html`

Decidable Logics Combining Heap Structures and Data

- LORIA-INRIA-Lorraine (CASSIS team)
  Nancy, France, February 2011
- Workshop on Automata and Logic for Data Manipulating Programs,
  Paris, France, December 2010
- ANR Veridyc Project,
  Laboratoire d'Informatique Algorithmique: Fondements et Applications (LIAFA),
  Paris, France, October 2010

The Tree-Width of the Auxiliary Storage

- School of Electronic Engineering and Computer Science at Queen Mary, University of London, London, UK, March 2013
- IFIP WG 2.3: Working Group on Programming Methodology, Meeting 52
  Winchester, UK. Set. 2011
- École normale supérieure de Cachan (ENS Cachan)
  Cachan, France, May 2010
- Laboratoire Bordelais de Recherche en Informatique (LaBRI),
  Bordeaux, France, May 2010
- Laboratoire d'Informatique Algorithmique: Fondements et Applications (LIAFA),
  Paris, France, April 2010

Writing Model-Checkers for Boolean Recursive Programs using a Fixed-Point Calculus
Laboratoire d'Informatique Algorithmique: Fondements et Applications (LIAFA),
Paris, France, September 2009

GRANTS

Amazon Reseach Grants (ARA): Program Analysis in the Clouds (PAC): a distributed symbolic algorithm to scale up bug-finding in concurrent programs. *Aug 2021–Present*
Total funding: 60,000 USD (Principal Investigator)

EPSRC Grant (EP/M008991/1): CONSEQUENCER: Sequentialization-based Verification of Concurrent Programs with FIFO Channels. *Mar 2015–Aug 2016*
Total funding: £ 120,000 (Principal Investigator)

EPSRC Grant (EP/P022413/1): VAC+: Verifier of Access Control. *Aug 2017–Present*
Total funding: £ 121,000 (Co-Investigator)

GCHQ Grant: Language-based security for smart contracts in distributed ledger. *Dec 2015–Mar 2016*
Total funding: £ 21,000 (Co-PI)

"Annual Adventures in Research" Grant, University of Southampton *May 2015*
Total funding: £ 7,000

"Annual Adventures in Research" Grant, University of Southampton *May 2012*
Total funding: £ 7,000

"7k scholarship project" Grant, University of Southampton *Feb 2012*
Total funding: £21,000

GRADUATE STUDENTS PURSUING PH.D.

Dr Enrico Steffinlongo *Oct 2016-Feb-2018*
Visiting PhD student from the University of Venice Ca' Foscari, Italy, Advisor Prof. Michele Bugliesi
(Now at **IBM research**, Eastleagh, UK)

Mikhail Ramalho *Feb 2015-Dec 2018*
(co-advised with Denis A. Nicole)

Dr Ermenegildo Tomasco *May 2012–Mar 2018*
Thesis title: *Separating Computation from Communication: A Design Approach for Concurrent Bug Finding.*
(Now official at **Italian Income Revenue Authority**, Rome, Italy.)

Dr Truc Lam Nguyen *Oct 2013-May 2017*
Thesis title: *A Pragmatic Verification Approach for Concurrent Programs.*
(Software Engineer at **GeoSpock Ltd**, Cambridge, UK.)

Dr Omar Inverso *Oct 2011-Oct 2015*
Thesis title: *Bounded Model Checking of Multi-threaded Programs via Sequentialization.*
(Postdoc at **Gran Sasso Science Institute**, Aquila, Italy.)

POSTDOCS

Mikhail Ramalho *May 2016-Aug 2016*

Dr Truc Lam Nguyen *Mar 2016-Aug 2016*

Dr Omar Inverso *Apr 2015-Nov 2015*

PROFESSIONAL ACTIVITIES

Refereed journals and conferences:
Formal Methods in System Design, Springer
Information and Computation, Elsevier Press
International Journal of Foundations of Computer Science (IJFCS)
Journal of Computer Security (JCS)
Journal of Systems and Software (JSS)
Journal on Software Tools for Technology Transfer (STTT)
Theoretical Computer Science (TCS)
IEEE Transactions on Software Engineering (TSE)

Int'l Conference on Automated Software Engineering (ASE)
Int'l Conference on Computer Aided Verification (CAV)
Int'l Conference on Concurrency Theory (CONCUR)
Int'l Conference on Developments in Language Theory (DLT)
Int'l Conference on Computer Aided Verification (FCT)
Int'l Conference on Formal Methods in Computer-Aided Design (FMCAD)
Int'l Workshop on Formal Methods for Industrial Critical Systems (FMICS)
Int'l Foundations of Software Science and Computation Structures (FOSSACS)
Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)
Int'l Symposium on Games, Automata, Logics and Formal Verification (GandALF)
Int'l Colloquium on Automata, Languages and Programming (ICALP)
Int'l Conference on Formal Engineering Methods (ICFEM)
Int'l Conference on Software Engineering (ICSE)

Int'l Symposium on Logic in Computer Science (LICS)
Int'l Symposium on Mathematical Foundations of Computer Science (MFCS)
Int'l Symposium on Principles of Programming Languages (POPL)
Int'l SPIN Workshop on Model Checking of Software (SPIN)
Int'l Symposium on Theoretical Aspects of Computer Science (STACS)
Int'l Competition on Software Verification (SV-COMP)
Int'l Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)
Int'l Symposium on Trustworthy Global Computing (TGC)
Int'l Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI)

Service

❑ PC Member: **OVERLAY 2021**, Padova, Italy.
3rd Workshop on Artificial Intelligence and fOrmal VERification, Logic, Automata, and sYnthesis.
❑ PC Member: **LATA 2020 & 2021**, Milan, Italy.
14th-15th Int'l Conference on Language and Automata Theory and Applications.
❑ PC Member: **GandALF 2020**, Brussels, Belgium.
11th Int'l Symposium on Games, Automata, Logics and Formal Verification.
❑ PC Member: **GandALF 2019**, Bordeaux, France.
10th Int'l Symposium on Games, Automata, Logics and Formal Verification.
❑ PC Member: **VSTTE 2018**, Oxford, UK.
10th Working Conference on Verified Software: Theories, Tools, and Experiments,
part of FLoC 2018.
❑ PC Member: **LATA 2018**, Ramat Gan, Israel.
12th Int'l Conference on Language and Automata Theory and Applications.
❑ PC Member: **GandALF 2018**, Saarbrücken, Germany.
9th Int'l Symposium on Games, Automata, Logics and Formal Verification.
❑ PC & Jury Member: **SV-COMP 2017**, Uppsala, Sweden.
6th Int'l Competition on Software Verification.
❑ ERC Member: **CAV 2016**, Toronto, Ontario, Canada.
28th Int'l Conference on Computer Aided Verification.
❑ PC & Jury Member: **SV-COMP 2016**, Eindhoven, Netherlands.
5th Int'l Competition on Software Verification.
❑ PC Member: **ICSE 2016**, Austin, TX, USA.
38th Int'l Conference on Software Engineering.
❑ PC Member: **TGC 2015**, Madrid, Spain.
10th Int'l Symposium on Trustworthy Global Computing.
❑ PC Member: **FCT 2015**, Gdansk, Poland.
20th Int'l Symposium on Fundamentals of Computation Theory.
❑ PC Member: **SPIN 2015**, Stellenbosch, South Africa.
22nd Int'l SPIN Workshop on Model Checking of Software.
❑ PC & Jury Member: **SV-COMP 2015**, London, UK.
4th Int'l Competition on Software Verification.
❑ PC Member: **GandALF 2015**, Genoa, Italy.
5th Int'l Symposium on Games, Automata, Logics and Formal Verification.
❑ PC & Jury Member: **SV-COMP 2014**, Grenoble, France.
3rd Int'l Competition on Software Verification.
❑ PC Member: **CAV 2012**, Berkeley, USA.
24th Int'l Conference on Computer Aided Verification.
❑ PC Member: **MFCS 2012**, Bratislava, Slovakia.
37th Int'l Symposium on Mathematical Foundations of Computer Science.
❑ PC Member: **GandALF 2010**, Minori, Italy.
1st Int'l Symposium on Games, Automata, Logics and Formal Verification.

TEACHING EXPERIENCE	**Semester 1, AY 2018/19, U. Southampton**: Theory of Computing (COMP2210) **July 2018**: 10-hour module on Program Verification, PhD program (IMT School for Advanced Studies Lucca, Italy) **May 2018**: 2018 Spring School on Theoretical Computer Science : Software Verification (6-hour module), Aussois, France **Semester 1, AY 2017/18, U. Southampton**: Theory of Computing (COMP2210) **Semester 2, AY 2017/18, U. Southampton**: Algorithmics (COMP1201) **Semester 1, AY 2016/17, U. Southampton**: Theory of Computing (COMP2210) **Semester 1, AY 2017/17**: Topics in Computer Science (COMP6233) **Semester 2, AY 2016/17, U. Southampton**: Algorithmics (COMP1201) **April 2016**: 20-hour module on Program Verification, PhD program (IMT School for Advanced Studies Lucca, Italy) **Semester 1, AY 2015/16, U. Southampton**: Theory of Computing (COMP2210) **Semester 1, AY 2015/16**: Topics in Computer Science (COMP6233) **June 2015**: Lecture series at UPMARC Summer School on Multicore Computing (Uppsala University, Sweden) **Semester 2, AY 2014/15**: Automated Software Verification (COMP6210) **Semester 1, AY 2014/15**: Theory of Computing (COMP2210) **Semester 2, AY 2013/14**: Formal Design of Systems (COMP6004) **Semester 1, AY 2013/14**: Theory of Computing (COMP2210) **Semester 2, AY 2012/13**: Formal Design of Systems (COMP6004) **Semester 1, AY 2012/13**: Theory of Computing (COMP2011) **Semester 2, AY 2011/12**: Formal Design of Systems (COMP6004)
RESEARCH EXPERIENCE	**Visiting Scholar, IMT School for Advanced Studies Lucca** *July-August 2018* Computer Science and System Engineering research group Hosted by Rocco De Nicola and Mirco Tribastone. Lucca, Italy **Visiting Scholar, University of Salerno** *July-August 2017* Department of Computer Science Hosted by Salvatore La Torre Fisciano, Italy **Visiting Scholar, IMT School for Advanced Studies Lucca** *Apr 2016* Computer Science and System Engineering research group Hosted by Rocco De Nicola and Mirco Tribastone. Lucca, Italy **Visiting Scholar, University of Salerno** *Apr 2016* Department of Computer Science Hosted by Salvatore La Torre Fisciano, Italy **Visiting Scholar, Stellenbosh University** *Mar 2016* Department of Computer Science Hosted by Bernd Fischer Stellenbosh, South Africa

Visiting Scholar, LIAFA, CNRS, University of Paris 7   *May 2014*
Laboratoire d'Informatique Algorithmique: Fondements et Applications
Hosted by Ahmed Bouajjani
Paris, France

Visiting Scholar, LIAFA, CNRS, University of Paris 7   *March 2012*
Laboratoire d'Informatique Algorithmique: Fondements et Applications
Hosted by Ahmed Bouajjani
Paris, France

Visiting Scholar, LIAFA, CNRS, University of Paris 7   *October 2011*
Laboratoire d'Informatique Algorithmique: Fondements et Applications
Hosted by Ahmed Bouajjani
Paris, France

Visiting Scholar, University of Illinois at Urbana-Champaign   *July 2011*
Dept. of Computer Science,
Hosted by P. Madhusudan
Urbana, USA

Visiting Scholar, University of Illinois at Urbana-Champaign   *Aug 2010*
Dept. of Computer Science,
Hosted by P. Madhusudan
Urbana, USA

LECTURES,
COLLOQUIA, AND
CONFERENCE
PRESENTATIONS

Concurrent Program Verification with Lazy Sequentialization and Interval Analysis
International Conference on Networked Systems (NETYS), Marrakech, Marocco, May 2017

Verifying Concurrent Programs by Memory Unwinding
International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS), London, UK, April 2015

On the Path-Width of Integer Linear Programming
International Symposium on Games, Automata, Logics and Formal Verification (GandALF)
Verona, Italy, September 2014

Bounded Model Checking of Multi-Threaded C Programs via Lazy Sequentialization
International Conference on Computer Aided Verification (CAV)
Vienna, Austria, July 2014

Scope-bounded Multistack Pushdown Systems: Fixed-Point, Sequentialization, and Tree-Width
Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)
Hyderabad, India, December 2012

On Sequentializing Concurrent Programs
International Static Analysis Symposium (SAS)
Venice, Italy, September 2011

Getting Rid of Store-Buffers in the Analysis of Weak Memory Models
International Conference on Computer Aided Verification (CAV)
Cliff Lodge, Snowbird, Utah, USA, July 2011

The Tree-Width of the Auxiliary Storage

Symposium on Principles of Programming Languages (POPL)
Austin, TX, USA, January 2011

The Tree-Width of the Auxiliary Storage
Formal Method seminar, University of Illinois at Urbana-Champaign
Urbana, IL, USA, March 2010

Analyzing Recursive Programs using Fixed-point Calculus
Formal Method seminar, University of Illinois at Urbana-Champaign
Urbana, IL, USA, October 2009

Reducing Context-bounded Concurrent Reachability to Sequential Reachability
Conference on Computer Aided Verification (CAV)
Grenoble, France, June 2009

An Infinite Automaton Characterization of Double Exponential Time
Conference on Computer Science Logic (CSL)
Bertinoro, Italy, September 2008

Context-Bounded Analysis of Concurrent Queue Systems
International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)
Budapest, Hungary, April 2008

On the Complexity of LTL Model-Checking of Recursive State Machines
International Colloquium on Automata, Languages and Programming (ICALP)
Wroclaw, Poland, July 2007

Hierarchical and Recursive State Machines with Context-Dependent Properties
International Colloquium on Automata, Languages and Programming (ICALP)
Eindhoven, The Netherlands, July 2003