Academic Curriculum Vitae

Yliès Falcone
Univ. Grenoble Alpes, Inria,
Laboratoire d'Informatique de Grenoble, Grenoble, France

Last updated: May 23, 2017

1 Identity and Contact Information

Title: Ph.D.
Last name: Falcone
First name: Yliès
Date of birth: December 9, 1981
Place of birth: Paris (France)

Professional Address:
Equipe CORSE - Équipe de recherche mixte Inria/LIG
Antenne Inria Giant, DRT/LETI/DACLE/
Building 50C – Office C 344
Minatec Campus, 17 rue des Martyrs
38054 Grenoble Cedex, FRANCE

E-mail: Ylies.Falcone@univ-grenoble-alpes.fr, Ylies.Falcone@inria.fr

Web: www.ylies.fr
Phone: +33 6 95 89 47 76
OrcID: 0000-0002-0114-0641

Twitter: @YliesFalcone
LinkedIn: yliesfalcone

2 Professional Experience

Current Position: Associate Professor at Univ. Grenoble Alpes and member of Inria, since September 2011.

Previous experiences

Start	End	Institution	Status
Feb 2015	Aug 2015	Formal Systems Laboratory (Urbana-Champaign - USA)	Invited Researcher
Feb 2011	March 2011	NICTA (Canberra - Australia)	Invited Researcher
Apr 2011	Apr 2011	U of Manchester (UK)	Honorary Researcher
Dec 2009	Aug 2011	Inria (Rennes - France)	Research associate
Oct 2006	Nov 2009	Verimag Laboratory (Grenoble - France)	Research Assistant
Oct 2006	Sept 2009	Grenoble University (France)	Teaching Assistant
Oct 2006	Nov 2006	Formal Systems Laboratory (Urbana-Champaign - USA)	Research Engineer
July 2005	Aug 2005	EDF R&D	Engineer (Paris - France)
July 2003	Aug 2003	Barclay's Bank PLC (Paris - France)	System Engineer

3 Education

Ph.D. in Computer Science from University of Grenoble – Verimag (Nov 2009) - French Ministry funding
Title: Study and implementation of runtime validation techniques.
Advisers: Prof. Jean-Claude Fernandez and Dr. Laurent Mounier.
(University of Grenoble does not provide distinctions to Ph.D. thesis)

Master in Computer Science from Grenoble University – Verimag (June 2006)
Specialty: Validation of critical systems.
Internship thesis: A formal framework for network security policy testing.
Adviser: Prof. Jean-Claude Fernandez and Dr. Laurent Mounier.
Distinction: Good.

4 Research Domain: Runtime Verification/Enforcement and Monitoring

Key words: monitoring, runtime verification, runtime enforcement, debugging, component-based systems, multi-threaded programs, decentralised and distributed systems.

Monitoring, runtime verification and enforcement. Monitoring comprises the set of theories, techniques, and tools for the formal verification and enforcement of specifications on a system using runtime information. Specifications are usually formalized as properties expressed in formal specification languages such variants of finite-state automata, variants of linear temporal logic, or rewriting rules. While in runtime verification one is interested in providing verdicts reflecting the satisfaction/violation of the specification, in runtime enforcement one tries to steer the system to ensure satisfaction and avoid violation of the specification. I am interested in both theoretical work and applications to several sorts of systems such as component-based, distributed, and realtime systems, as well as systems used in industry and public systems.

More on the theoretical side, I studied expressiveness of runtime techniques [54] such as runtime verification [24, 25], runtime enforcement [23, 33], and testing [21, 22]. I also contributed to the definition of expressive specification formalisms and monitors with the definition of Quantified Event Automata [3] and to the definition of certified runtime verification [8] where one can obtain certified monitors.

Decentralised monitoring. I have contributed to the definitions of frameworks for *decentralised runtime verification* where the system under scrutiny is composed of several entities and the classical central observation point required for monitoring does not exist. First, we investigated frameworks where monitors migrate across components of the system, first when monitors are encoded by LTL formulae [7, 6] or finite-state automata [18]. Then, we defined frameworks where monitors follow a choreography driven by the monitored specification [12]. Finally, we generalised previous work by defining the notion of decentralised specification and an analytical framework to design and compare algorithms for decentralised runtime verification [15].

Runtime enforcement. I wrote a tutorial/survey on *runtime enforcement* [17]. I am interested in defining frameworks for untimed [33, 51, 48, 52] and timed [43, 46, 45, 30, 50] systems. In [33], we introduced models of enforcement monitors generalising those available in the litterature. We later extended the monitors in [33] to account for uncontrollable events [48]. We recently used game theory as an alternative to synthesise the state-space of enforcement monitors ahead of time (and thus reduce runtime computation). We pioneered runtime enforcement for safety and co-safety timed properties in [47, 46] by defining enforcement monitors as delayers. We later extended enforcement monitoring to any regular timed property in [44] and endowed enforcement monitors with suppression abilities in [30]. Optimal enforcement monitors for timed properties with uncontrollable events were introduced in [50]. Our theoretical results on runtime enforcement of timed properties were implemented in the TiPeX tool [45].

Monitoring for component-based systems. I am also interested in defining runtime verification and enforcement for component-based systems (CBSs). CBSs consist of existing components composed with interactions. From a runtime verification perspective, dealing with CBSs offer the advantage of having a more precise description of the system under scrutiny. However, CBSs are endowed with more complex and subtle behaviour than traditional monolithic systems. For CBSs, I contributed to the definition of a runtime verification for sequential [28, 29] and multi-threaded [42, 41] CBSs, runtime enforcement of sequential CBSs [28, 27], and efficient instrumentation techniques [16] where we ported aspect-oriented programming to CBSs.

Applications of monitoring. I am interested also in applying runtime verification and enforcement principles to concrete application domains and case studies. I have also organised workshops on the specific topic of applications for runtime verification and enforcement [36, 37, 5]. We defined runtime verification and enforcement approaches in security by defining monitors that verify and enforce the co-called notion of opacity of secrets [31]. We designed and implemented concrete verification and enforcement monitors for Android devices [20, 13] by relying on an adaptation of aspect-oriented programming to Android [19]. We conducted case studies on using these monitors for blocking advertisement and preserving user privacy in Android apps [14]. Other applications of runtime verification and enforcement in which I was involved in include the mitigation of DMA races [53], the enforcement of document life cycles [38], and the monitoring of electronic exams [39, 40].

Contributions to the community. I am also very keen on contributing to the runtime verification community (in the large sense) by several means. I have co-organised scientific events [36, 5, 35] and special issues of journals ([37] plus one ongoing special issue). I have contributed to the diffusion of knowledge on untime verification and enforcement through tutorials [17, 26] and the delivery of lectures (in my university and abroad). I have also organised the first summer school on runtime verification [11]. Finally, I have co-founded the international competition on software tools for runtime verification, largely contributing to its design principles, rules, and scoring schemes with Ezio Bartocci. I was involved in the first three incarnations of the competition [4, 34, 49].

5 Awards and Distinctions

- *Best Paper Award* from EDOC 2016 PC for the article "Decentralized Enforcement of Artifact Lifecycles".
- *Best Paper Award* from Formal Methods Europe obtained in August 2012 for the article "Decentralised LTL Monitoring" at the conference FM 2012.
- *Best Paper Award* from IFIP obtained in November 2010 for the article "More Testable Properties" at the conference ICTSS'10. The article has been also invited for an extended version to the journal Software Tools for Technology Transfer.
- *Explorer Grant* from INRIA obtained in November 2010 in order to develop collaborative work on runtime verification.

6 International Publications and Scientific Production

6.1 Synthesis

Journal	Conferences	Proceedings	Book Chapters
17	44	2	1

The full list of my publications is at `www.ylies.fr/?page_id=47`.

6.2 Metrics (source: `Google Scholar`)

H-index: 15. **Maximal number of citations for a publication:** 75 times.
i10-index: 25. **Total number of citations:** 872.

6.3 Selected Publications

1. Runtime enforcement of regular timed properties by suppressing and delaying events. Y. Falcone, T. Jéron, H. Marchand, S. Pinisetty. In SCP: Science of Computer Programming. Volume 123, 1 July 2016, Pages 2–41. DOI: 10.1016/j.scico.2016.02.008
2. *Runtime Verification of Component-Based Systems in the BIP Framework with Formally-Proved Sound and Complete Instrumentation.* Y. Falcone, M. Jaber, Th-H. Nguyen, M. Bozga, S. Bensalem. In SOSYM: SOftware and SYstem Modeling. February 2015, Volume 14, Issue 1, pp 173-199
3. *Decentralised LTL Monitoring.* A. Bauer, Y. Falcone. In FM'2012: 18th international symposium on Formal Methods.
4. *What can you Verify and Enforce at Runtime?* Y. Falcone, J-C. Fernandez, L. Mounier. In STTT: Software Tools for Technology Transfer – Special issue on Runtime Verification. June 2012, Vol. 14, Issue 3, pp 349-382.

7 Software

I was involved in the design and development of several software demonstrators and prototypes.

- Software demonstrators:
  1. *mi-AdBlocker.* mi-AdBlocker is an ad blocker based on monitoring techniques that offered several advantages when it was released compared to the commercial existing ad blockers.
  2. *mi-Security.* mi-Security allows user of Android devices to setup security policies over applications in an easy way.

  mi-AdBlocker and mi-Security were realised based on the software prototypes WeaveDroid and RV-Droid.
- Software prototypes realised for conferences and journals:
  1. *THEMIS: A Tool for Decentralized Monitoring Algorithms.* THEMIS allows to design, simulate, and compare decentralised monitoring algorithms.
  2. *TiPEX: Timed Properties Enforcement on eXecution.* TiPEX is a tool for the runtime enforcement of timed properties: it allows to generate timed properties as timed automata and enforce them over systems. TiPEX is described in the RV 15 publication [45].

3. *DecentMon 3: Decentralised Monitoring of LTL formulas under different organizations.* DecentMon 3 is a benchmark to assess the cost of decentralized monitoring of LTL formulas under different monitor organization. DecentMon 3 is described in the RV 2014 publication [10].
4. *DecentMon 2: Efficient Decentralised Monitoring of Regular Languages.* DecentMon 2 is a benchmark to asses the cost of decentralised monitoring of regular languages in a more efficient way than DecentMon 1. DecentMon 2 is described in the FORTE 2014 publication [18].
5. *RE-BIP: Runtime Enforcement for BIP systems.* RE-BIP allows to integrate runtime enforcement monitors in a component-based systems described in the Behavior Interaction Priority framework. RE-BIP is described in the SAC 2014 and STTT journal publications [9, 27].
6. *TAKOS: Toolbox for the K-step Opacity of Systems.* TAKOS allows to analyse the opacity of secrecy properties over systems (to ensure the confidentiality of sensitive information over systems). TAKOS was realised the CDC 13 and JDEDS publications [1, 31].
7. *Weave Droid - protected by the french institute for software protection (APP).* Weave Droid allows aspect-oriented programming on Android devices.
8. *j-Cyclitur - protected by the french institute for software protection (APP).* j-Cyclitur allows compacting and compressing cyclic traces of micro-controllers. j-Cyclitur was developed in the context of project FUI IO32 and is described in the IESS 2013 publication [2].
9. *DecentMon 1: Decentralised Monitoring of LTL formulae.* DecentMon is a benchmark to assess the cost of decentralized monitoring of LTL formulas. DecentMon is described in the FM 2012 and FMSD publications [7, 6].
10. *RV-Droid: Runtime Verification and Enforcement for Android devices.* RV-Droid allows users to runtime verify and enforce properties on Android applications. RV-Droid is described in the RV 2012 publication [20].
11. Java-PT: Property Testability in Java. Java-PT allows to model regular properties and analyse their testability. Java-PT is described in the STTT journal publication [22].
12. *RV-BIP: Runtime Verification for BIP systems.* RV-BIP allows to integrate runtime verification monitors in a component-based systems described in the Behavior Interaction Priority framework. RV-BIP was developed for the SEFM 2011 and SoSYM journal publications [28, 29]
13. *j-POST: Property-Oriented Software Testing for Java.* j-POST allows property-oriented testing on software. j-POST is described in the MBT 2008 publication [32].
14. *j-VETO: Verification and Enforcement TOolbox.* j-VETO allows to monitor parametric properties on Java programs. j-VETO is described in the STTT publication [25].

8 Service to the Research Community

8.1 Scientific Leadership

- Representing France in the COST Action IC 1402 ARVI on runtime verification beyond monitoring.
- Co-chairing with Ezio Bartocci the Working Group on Core Runtime Verification in the COST Action IC 1402 ARVI on runtime verification beyond monitoring.
- Co-founded the competition on runtime verification with Borzoo Bonakdarpour and Ezio Bartocci and member of the steering committee.
- Co-organised with Christian Colombo the first international summer school on Runtime Verification sponsored by the COST Action ArVI; attracting more than 50 students to the school.
- Co-organised with Saddek Bensalem two international summer schools on Cyber-Physical Systems sponsored by PERSYVAL-lab (french excellence initiative) and EIT ICT Labs; attracting more than 70 students to both editions of the school.

8.2 co-Editing

- RV 2010: the first international conference on Runtime Verification.
- Guest co-editor of a special issue in Software Tools for Technology Transfer on Runtime Verification, in 2013.
- RV 2016: the sixteenth international conference on Runtime Verification.
- Guest co-editor of a special issue in Formal Methods in System Design on Runtime Verification, in 2017.

8.3 Chairing and Participating to Committees

8.3.1 Chairing and co-Chairing Program/Scientific Committees

SAC-SVT'17, RV'16, CRV'16, CRV'15, CSRV'14.

8.3.2 Participation to Program Committees

RV-CuBES, RV'17, TASE'17, iFM'17, PDP'17, DATE'17 – Topic E3, CARI'16, SSS'16, RV'16, Pre-Post'16, SAC-SVT'16, CRI'15, MSR'15, IPAC'15, PCODA'15, RV'15, SAC-SVT'15, SecTest'14, CSRV'14, FORTE'14, RV'14 (main track), RV'14 (tool track), RV'13, MSR'13, SecTest'13, Isola'12, RV'12, SecTest'12, SERP'11, RV'09.

8.3.3 Organizing Events

Co-organization and scientific coordination for the Cyber-Physical Systems Summer School in 2013 and 2014.

8.3.4 Participating to Organization Committees

SPIN'17, the 24th International SPIN Symposium on Model Checking of Software (Publicity Chair); IsoLA'16 (co-organization with Ezio Bartocci) for the track on "Runtime Verification and Enforcement: the Application Perspective"; CRV'16, the 3rd international Competition on Runtime Verification (co-Chair); CRV'15, the 2nd international Competition on Runtime Verification (Chair); Summer School on Cyber-Physical Systems 2014 (Chair); RV'14 (chair of the publicity committee); CSRV'14, the 1st International Competition on Software for Runtime Verification (co-Chair); ETAPS'14 (Publicity chair and assisting the general Organization chair); MAROC'13 (organization committee and PC co-chair); Summer School on Cyber-Physical Systems 2013; IsoLA'12 (co-organization with Lenore D. Zuck) for the track on "Runtime Verification: the Application Perspective"; AFADL'12 (Organization committee); RV'10 (Publicity Chair); CAV'09 (Organization committee); VVPS'09 (Web).

8.4 Reviewing

8.4.1 For Journals and Conferences

- Journals: IEEE Design and Test, ACM Transactions on Software Engineering and Methodology, Formal Aspects of Computing, ACM Transactions on Automatic and Control, Acta Informatica, Formal Methods in System Design, International Journal of Information and Computer Security, Science of Computer Programming, Software Tools for Technology Transfer, Journal of Systems and Software, ACM Transaction on Aspect-Oriented Development, The Computer Journal, ACM Transactions on Information and System Security, Elsevier Computer Review, Software Testing Verification and Reliability.
- Conferences (in addition to PC memberships): TACAS'17, NFM'16, ASE'15, OOPSLA'15, FM'15, CAV'15, DAC'15, DATE'15, CAV'14, MBT'14, SAC-SVT'14, DATE'14, SSS'13, SecTest'13, FM'12, ICST'12, DATE'12, ICFEM'11, HSCC'11, FASE'11, WoDES'10, VECoS'10, TASE'10, CAV'09, RV'08.

8.4.2 For Projects

American University of Beirut projects (2015), Icelandic Center for Research projects (2014), American University of Beirut projects (2014), ANRT Ph.D. Grant (2013), ANR (French NSF) Grants (2013).

9 Departmental Activities

- Co-responsible of the first year of the International Master in Computer Science – MoSIG (2014 – ongoing).
- Mission Valorisation for Grenoble Informatics Lab (2014 – ongoing).
- Elected member of the concil of Grenoble Informatics Lab (2016-ongoing).
- Member of the educational board of Persyval-lab (2012 – 2014).
- In charge of elaborating research collaborations between the university of Grenoble and Galatasaray University (2012,–). Several visits to the university of Galatasaray, Istanbul, Turkey.

- Representing University of Grenoble at the Galatasaray Consortiums, in Turkey and France (2011 - ongoing)
- Member of Verimag council in 2008 and 2009: Ph.D. students representative.
- Webmaster of Verimag DCS Website (in 2007 and 2008): creating and managing a Website for 25 persons.

10 Ph.D. and Scientific Supervisions

Ph.D.	Master 2 and Engineers	Master 1 & Undergraduate
3 (4 ongoing)	7 (1 ongoing)	24 (1 ongoing)

10.1 Ph.D. Students

- *Azzeddine Amiar, Ph.D. student (2011 - 2013), University of Grenoble*. In the context of IO 32 project with ST Microelectronics. Defended on November 27th, 2013. Co-supervision with Lydie du Bousquet.
  The thesis of Azzeddine deals with fault-localization in microcontrolers. By using probes on microcontrolers we retrieved their execution traces. Analyzing traces was difficult because it contained a huge amount of low-level information. However, traces are cyclic in the sense that they contain hierarchical patterns of events. Azzeddine leveraged the cyclic nature and defined algorithms to obtain more compact representations of the traces and adapted existing fault-localization algorithms to cyclic traces.
- *Srinivas Pinisetty, 2012 – 2014, University of Rennes*. In the context of the ANR Vacsim project. Defended on January 23rd, 2015. Co-supervision, with Thierry Jéron and Hervé Marchand.
  The thesis of Srinivas introduces runtime enforcement for timed properties. We introduced the notion of enforcement mechanisms to generalise the notion of enforcement monitors (usually described in the literature). Enforcement mechanisms for timed properties are delayers with suppression that is they store and suppress actions to make the execution sequence satisfy the property. Enforcement mechanisms are described at several levels of abstractions and we proved the existence of a refinement relation between the levels of abstraction.
- *Abdurrahman Pektas, 2013 - 2015, University of Grenoble and University of Galatasaray.* Defended on December 10th, 2015. Co-supervision with Tankut Acarman and Jean-Claude Fernandez.
  The thesis of Abdurrahman addresses the problem of malware detection and classification. We wanted to design a method that scales with the constantly-growing number of malware. Abdurrahman extended data mining and machine learning techniques based on the insight that new malwares are often variants (mutations) of existing (and already classified) malware.
- *Hosein Nazarpour, 2013 - 2017, Univ. Grenoble Alpes.* To be defended on June 26, 2017. Co-supervision with Saddek Bensalem. The thesis of Hosein deals with monitoring of component-based systems. We define techniques to monitor multi-threaded and distributed component-based systems. The developed techniques are efficient in the sense that they do not impact much the performance of the monitored systems. Moreover, these techniques preserve the concurrency of the monitored system; thus ensuring the soundness of the observed verdicts.
- *Antoine El-Hokayem, 2015 - 2018, Univ. Grenoble Alpes.* I am supervising Antoine (alone) as I have asked for a special authorisation.
  The thesis of Antoine deals with decentralised runtime verification. We aim at defining theories for monitoring decentralised systems by introducing the notion of decentralised specification, and study how one can obtain decentralised specifications from centralised ones.
- *Raphaël Jakse, 2016 - 2019, Univ. Grenoble Alpes.* Co-supervision with Jean-François Mehaut.
  The thesis of Raphaël aims at combining runtime verification and debugging. We aim at defining what we call interactive runtime verification where one can automate some tedious phases of debugging by using runtime monitors to locate faults and steer the execution to explore the execution and ensure better code coverage.
- *Fabian Gruber, 2016 - 2019, Univ. Grenoble Alpes.* Co-supervision with Fabrice Rastello.
  The thesis of Fabian deals with performance profiling and debugging. We aim at defining what we call interactive performance debugging where one can interactively profile the performance of application by using a debugger.

10.2 Master Students, Engineers, and U of Grenoble Excellence Internships

I have supervised or co-supervised more than 25 master, bachelor, and Grenoble excellence initiative students from University of Grenoble or abroad over the last years. Please see my website (`http://www.ylies.fr/?page_id=169`) for more details.

11 Teaching Activities

11.1 At Univ. Grenoble Alpes

As an associate professor, I perform at least 192 hours of teaching activities in front of students, every year. In France, these hours do not include surrounding teaching activities such as office hours and grading which are done by the person supervising the course.

Below is a summary of the lectures I am or was involved in.

- *Programming Languages and Compiler Design* in International Master (supervision, english lecture, and english tutorial).
- *Languages and Automata* in Bachelor year 2 (supervision, french lecture, english and french tutorials).
- *Advanced Testing Techniques* in Master year 2 (supervision, french lecture),
- *Introduction to Runtime Verification* in International Master year 2 (english lecture).
- *Object-oriented Design and Programming* in Master year 1 (french tutorial).
- *Introduction to Functional Programming* in Bachelor year 1 (supervision, english lecture and tutorial).
- *Proof techniques and logic for computer science* in International Master year 1 (lecture and tutorial).

11.2 Invited Lectures and Tutorials

In addition to the courses at Univ. Grenoble-Alpes, I have given invited lectures and tutorials:

- *Runtime Verification and Enforcement for Realtime Systems*, in ETR Summer School, Paris, France, in 2017.
- *Runtime Verification and Enforcement: an introduction*, in TAROT Summer School, Napoli, Italy, in 2017.
- *Introduction to Runtime Verification and Monitoring Android Systems* at American University of Beirut, Lebanon, in 2016.
- *Introduction to Runtime Verification* at Galatasaray University, Istanbul, Turkey, in 2016.
- *Introduction to Programming Languages and Compiler Design* at Galatasaray University, Istanbul, Turkey, in 2012-2016.
- *Programming Languages and Compiler Design* at École Nationale Polytechnique de Yaoundé (Polytechnique School Yaoundé), CETIC, Yaoundé, Cameroon, in 2015.
- *Monitoring Android Systems* at ENSA Tetouan, Morocco, in 2014.
- *Introduction to Model Checking* at Universidad Pontifica Javeriana, Bogota, Colombia, in 2012.

12 Previous and Current Collaborations

Howard Barringer (University of Manchester, UK); Ezio Bartocci (Vienna University, Austria); Saddek Bensalem (Univ. Grenoble Alpes, France); Marius Bozga (CNRS, France); Andreas Bauer (NEC, Germany); Jan Olaf Blech (RMIT University, Australia); Christian Colombo (University of Malta, Malta); Mickaël Delahaye (CEA Saclay, France); Jean-Claude Fernandez (Univ Grenoble Alpes, France); Klaus Havelund (NASA JPL, USA); Sylvain Hallé (University of Quebec, Canada); Mohamad Jaber (American University of Beirut, Lebanon); Thierry Jéron (Inria Rennes, France); Raphaël Khoury (University of Quebec, Canada); Hervé Marchand (Inria Rennes, France); Jean-François Méhaut (Univ. Grenoble Alpes, France); Laurent Mounier (Univ. Grenoble-Alpes, France); Than-Hung Nguyen (Hanoi University of Sciences and Technology, Vietnam); Abdurrahman Pektas (Tubitak, Turkey); Srinivas Pinisetty (KTH, Sweden); Giles Reger (University of Manchester, UK); Antoine Rollet (University of Bordeaux, France); Grigore Rosu (University of Illinois at Urbana-Champaign, USA); Yi Zhang (University of Illinois at Urbana-Champaign, USA); Lenore Zuck (University of Illinois at Chicago, USA).

References

[1] *Proceedings of the 52nd IEEE Conference on Decision and Control, CDC 2013, December 10-13, 2013, Firenze, Italy*. IEEE, 2013.

[2] Azzeddine Amiar, Mickaël Delahaye, Yliès Falcone, and Lydie du Bousquet. Compressing microcontroller execution traces to assist system analysis. In Gunar Schirner, Marcelo Götz, Achim Rettberg, Mauro Cesar Zanella, and Franz J. Rammig, editors, *Embedded Systems: Design, Analysis and Verification - 4th IFIP TC 10 International Embedded Systems Symposium, IESS 2013, Paderborn, Germany, June 17-19, 2013. Proceedings*, volume 403 of *IFIP Advances in Information and Communication Technology*, pages 139–150. Springer, 2013.

[3] Howard Barringer, Yliès Falcone, Klaus Havelund, Giles Reger, and David E. Rydeheard. Quantified event automata: Towards expressive and efficient runtime monitors. In Dimitra Giannakopoulou and Dominique Méry, editors, *FM 2012: Formal Methods - 18th International Symposium, Paris, France, August 27-31, 2012. Proceedings*, volume 7436 of *Lecture Notes in Computer Science*, pages 68–84. Springer, 2012.

[4] Ezio Bartocci, Borzoo Bonakdarpour, and Yliès Falcone. First international competition on software for runtime verification. In Borzoo Bonakdarpour and Scott A. Smolka, editors, *Runtime Verification - 5th International Conference, RV 2014, Toronto, ON, Canada, September 22-25, 2014. Proceedings*, volume 8734 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 2014.

[5] Ezio Bartocci and Yliès Falcone. Runtime verification and enforcement, the (industrial) application perspective (track introduction). In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation: Discussion, Dissemination, Applications - 7th International Symposium, ISoLA 2016, Imperial, Corfu, Greece, October 10-14, 2016, Proceedings, Part II*, volume 9953 of *Lecture Notes in Computer Science*, pages 333–338, 2016.

[6] Andreas Bauer and Yliès Falcone. Decentralised LTL monitoring. *Formal Methods in System Design*, 48(1-2):46–93, 2016.

[7] Andreas Klaus Bauer and Yliès Falcone. Decentralised LTL monitoring. In Dimitra Giannakopoulou and Dominique Méry, editors, *FM 2012: Formal Methods - 18th International Symposium, Paris, France, August 27-31, 2012. Proceedings*, volume 7436 of *Lecture Notes in Computer Science*, pages 85–100. Springer, 2012.

[8] Jan Olaf Blech, Yliès Falcone, and Klaus Becker. Towards certified runtime verification. In Toshiaki Aoki and Kenji Taguchi, editors, *Formal Methods and Software Engineering - 14th International Conference on Formal Engineering Methods, ICFEM 2012, Kyoto, Japan, November 12-16, 2012. Proceedings*, volume 7635 of *Lecture Notes in Computer Science*, pages 494–509. Springer, 2012.

[9] Hadil Charafeddine, Khalil El-Harake, Yliès Falcone, and Mohamad Jaber. Runtime enforcement for component-based systems. In Roger L. Wainwright, Juan Manuel Corchado, Alessio Bechini, and Jiman Hong, editors, *Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, April 13-17, 2015*, pages 1789–1796. ACM, 2015.

[10] Christian Colombo and Yliès Falcone. Organising LTL monitors over distributed systems with a global clock. In Borzoo Bonakdarpour and Scott A. Smolka, editors, *Runtime Verification - 5th International Conference, RV 2014, Toronto, ON, Canada, September 22-25, 2014. Proceedings*, volume 8734 of *Lecture Notes in Computer Science*, pages 140–155. Springer, 2014.

[11] Christian Colombo and Yliès Falcone. First international summer school on runtime verification - as part of the arvi COST action 1402. In Yliès Falcone and César Sánchez, editors, *Runtime Verification - 16th International Conference, RV 2016, Madrid, Spain, September 23-30, 2016, Proceedings*, volume 10012 of *Lecture Notes in Computer Science*, pages 17–20. Springer, 2016.

[12] Christian Colombo and Yliès Falcone. Organising LTL monitors over distributed systems with a global clock. *Formal Methods in System Design*, 49(1-2):109–158, 2016.

[13] Philip Daian, Yliès Falcone, Patrick O'Neil Meredith, Traian-Florin Serbanuta, Shin'ichi Shiriashi, Akihito Iwai, and Grigore Rosu. Rv-android: Efficient parametric android runtime verification, a brief tutorial. In Ezio Bartocci and Rupak Majumdar, editors, *Runtime Verification - 6th International Conference, RV 2015 Vienna, Austria, September 22-25, 2015. Proceedings*, volume 9333 of *Lecture Notes in Computer Science*, pages 342–357. Springer, 2015.

[14] Khalil El-Harake, Yliès Falcone, Wassim Jerad, Matthieu Langet, and Mariem Mamlouk. Blocking advertisements on android devices using monitoring techniques. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation. Specialized Techniques and Applications - 6th International Symposium, ISoLA 2014, Imperial, Corfu, Greece, October 8-11, 2014, Proceedings, Part II*, volume 8803 of *Lecture Notes in Computer Science*, pages 239–253. Springer, 2014.

[15] Antoine El-Hokayem and Yliès Falcone. Monitoring decentralized specifications. In Koushik Shen, editor, *26th International Symposium on Software Testing and Analysis - ISSTA 2017, Santa Barbara, USA, July 10-14, 2017, Proceedings*. ACM, 2017. To appear.

[16] Antoine El-Hokayem, Yliès Falcone, and Mohamad Jaber. Modularizing crosscutting concerns in component-based systems. In Rocco De Nicola and eva Kühn, editors, *Software Engineering and Formal Methods - 14th International Conference, SEFM 2016, Held as Part of STAF 2016, Vienna, Austria, July 4-8, 2016, Proceedings*, volume 9763 of *Lecture Notes in Computer Science*, pages 367–385. Springer, 2016.

[17] Yliès Falcone. You should better enforce than verify. In Howard Barringer, Yliès Falcone, Bernd Finkbeiner, Klaus Havelund, Insup Lee, Gordon J. Pace, Grigore Rosu, Oleg Sokolsky, and Nikolai Tillmann, editors, *Runtime Verification - First International Conference, RV 2010, St. Julians, Malta, November 1-4, 2010. Proceedings*, volume 6418 of *Lecture Notes in Computer Science*, pages 89–105. Springer, 2010.

[18] Yliès Falcone, Tom Cornebize, and Jean-Claude Fernandez. Efficient and generalized decentralized monitoring of regular languages. In Erika Ábrahám and Catuscia Palamidessi, editors, *Formal Techniques for Distributed Objects, Components, and Systems - 34th IFIP WG 6.1 International Conference, FORTE 2014, Held as Part of the 9th International Federated Conference on Distributed Computing Techniques, DisCoTec 2014, Berlin, Germany, June 3-5, 2014. Proceedings*, volume 8461 of *Lecture Notes in Computer Science*, pages 66–83. Springer, 2014.

[19] Yliès Falcone and Sebastian Currea. Weave droid: aspect-oriented programming on android devices: fully embedded or in the cloud. In Michael Goedicke, Tim Menzies, and Motoshi Saeki, editors, *IEEE/ACM International Conference on Automated Software Engineering, ASE'12, Essen, Germany, September 3-7, 2012*, pages 350–353. ACM, 2012.

[20] Yliès Falcone, Sebastian Currea, and Mohamad Jaber. Runtime verification and enforcement for android applications with rv-droid. In Shaz Qadeer and Serdar Tasiran, editors, *Runtime Verification, Third International Conference, RV 2012, Istanbul, Turkey, September 25-28, 2012, Revised Selected Papers*, volume 7687 of *Lecture Notes in Computer Science*, pages 88–95. Springer, 2012.

[21] Yliès Falcone, Jean-Claude Fernandez, Thierry Jéron, Hervé Marchand, and Laurent Mounier. More testable properties. In Alexandre Petrenko, Adenilso da Silva Simão, and José Carlos Maldonado, editors, *Testing Software and Systems - 22nd IFIP WG 6.1 International Conference, ICTSS 2010, Natal, Brazil, November 8-10, 2010. Proceedings*, volume 6435 of *Lecture Notes in Computer Science*, pages 30–46. Springer, 2010.

[22] Yliès Falcone, Jean-Claude Fernandez, Thierry Jéron, Hervé Marchand, and Laurent Mounier. More testable properties. *STTT*, 14(4):407–437, 2012.

[23] Yliès Falcone, Jean-Claude Fernandez, and Laurent Mounier. Enforcement monitoring wrt. the safety-progress classification of properties. In Sung Y. Shin and Sascha Ossowski, editors, *Proceedings of the 2009 ACM Symposium on Applied Computing (SAC), Honolulu, Hawaii, USA, March 9-12, 2009*, pages 593–600. ACM, 2009.

[24] Yliès Falcone, Jean-Claude Fernandez, and Laurent Mounier. Runtime verification of safety-progress properties. In Saddek Bensalem and Doron A. Peled, editors, *Runtime Verification, 9th International Workshop, RV 2009, Grenoble, France, June 26-28, 2009. Selected Papers*, volume 5779 of *Lecture Notes in Computer Science*, pages 40–59. Springer, 2009.

[25] Yliès Falcone, Jean-Claude Fernandez, and Laurent Mounier. What can you verify and enforce at runtime? *STTT*, 14(3):349–382, 2012.

[26] Yliès Falcone, Klaus Havelund, and Giles Reger. A tutorial on runtime verification. In Manfred Broy, Doron A. Peled, and Georg Kalus, editors, *Engineering Dependable Software Systems*, volume 34 of *NATO Science for Peace and Security Series, D: Information and Communication Security*, pages 141–175. IOS Press, 2013.

[27] Yliès Falcone and Mohamad Jaber. Fully automated runtime enforcement of component-based systems with formal and sound recovery. *STTT*, 19(3):341–365, 2017.

[28] Yliès Falcone, Mohamad Jaber, Thanh-Hung Nguyen, Marius Bozga, and Saddek Bensalem. Runtime verification of component-based systems. In Gilles Barthe, Alberto Pardo, and Gerardo Schneider, editors, *Software Engineering and Formal Methods - 9th International Conference, SEFM 2011, Montevideo, Uruguay, November 14-18, 2011. Proceedings*, volume 7041 of *Lecture Notes in Computer Science*, pages 204–220. Springer, 2011.

[29] Yliès Falcone, Mohamad Jaber, Thanh-Hung Nguyen, Marius Bozga, and Saddek Bensalem. Runtime verification of component-based systems in the BIP framework with formally-proved sound and complete instrumentation. *Software and System Modeling*, 14(1):173–199, 2015.

[30] Yliès Falcone, Thierry Jéron, Hervé Marchand, and Srinivas Pinisetty. Runtime enforcement of regular timed properties by suppressing and delaying events. *Systems & Control Letters*, 123:2–41, 2016.

[31] Yliès Falcone and Hervé Marchand. Enforcement and validation (at runtime) of various notions of opacity. *Discrete Event Dynamic Systems*, 25(4):531–570, 2015.

[32] Yliès Falcone, Laurent Mounier, Jean-Claude Fernandez, and Jean-Luc Richier. j-post: a java toolchain for property-oriented software testing. *Electr. Notes Theor. Comput. Sci.*, 220(1):29–41, 2008.

[33] Yliès Falcone, Laurent Mounier, Jean-Claude Fernandez, and Jean-Luc Richier. Runtime enforcement monitors: composition, synthesis, and enforcement abilities. *Formal Methods in System Design*, 38(3):223–262, 2011.

[34] Yliès Falcone, Dejan Nickovic, Giles Reger, and Daniel Thoma. Second international competition on runtime verification CRV 2015. In Ezio Bartocci and Rupak Majumdar, editors, *Runtime Verification - 6th International Conference, RV 2015 Vienna, Austria, September 22-25, 2015. Proceedings*, volume 9333 of *Lecture Notes in Computer Science*, pages 405–422. Springer, 2015.

[35] Yliès Falcone and César Sánchez, editors. *Runtime Verification - 16th International Conference, RV 2016, Madrid, Spain, September 23-30, 2016, Proceedings*, volume 10012 of *Lecture Notes in Computer Science*. Springer, 2016.

[36] Yliès Falcone and Lenore D. Zuck. Runtime verification: The application perspective. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation. Technologies for Mastering Change - 5th International Symposium, ISoLA 2012, Heraklion, Crete, Greece, October 15-18, 2012, Proceedings, Part I*, volume 7609 of *Lecture Notes in Computer Science*, pages 284–291. Springer, 2012.

[37] Yliès Falcone and Lenore D. Zuck. Runtime verification: the application perspective. *STTT*, 17(2):121–123, 2015.

[38] Sylvain Hallé, Raphaël Khoury, Antoine El-Hokayem, and Yliès Falcone. Decentralized enforcement of artifact lifecycles. In Florian Matthes, Jan Mendling, and Stefanie Rinderle-Ma, editors, *20th IEEE International Enterprise Distributed Object Computing Conference, EDOC 2016, Vienna, Austria, September 5-9, 2016*, pages 1–10. IEEE Computer Society, 2016.

[39] Ali Kassem, Yliès Falcone, and Pascal Lafourcade. Monitoring electronic exams. In Ezio Bartocci and Rupak Majumdar, editors, *Runtime Verification - 6th International Conference, RV 2015 Vienna, Austria, September 22-25, 2015. Proceedings*, volume 9333 of *Lecture Notes in Computer Science*, pages 118–135. Springer, 2015.

[40] Ali Kassem, Yliès Falcone, and Pascal Lafourcade. Formal analysis and offline monitoring of electronic exams. *Formal Methods in System Design*, 2017. To appear.

[41] Hosein Nazarpour, Yliès Falcone, Saddek Bensalem, and Marius Bozga. Concurrency-preserving and sound monitoring of multi-threaded component-based systems. *Formal Aspects of Computing*, 2017.

[42] Hosein Nazarpour, Yliès Falcone, Saddek Bensalem, Marius Bozga, and Jacques Combaz. Monitoring multi-threaded component-based systems. In Erika Ábrahám and Marieke Huisman, editors, *Integrated Formal Methods - 12th International Conference, IFM 2016, Reykjavik, Iceland, June 1-5, 2016, Proceedings*, volume 9681 of *Lecture Notes in Computer Science*, pages 141–159. Springer, 2016.

[43] Srinivas Pinisetty, Yliès Falcone, Thierry Jéron, and Hervé Marchand. Runtime enforcement of parametric timed properties with practical applications. In Jean-Jacques Lesage, Jean-Marc Faure, José E. R. Cury, and Bengt Lennartson, editors, *12th International Workshop on Discrete Event Systems, WODES 2014, Cachan, France, May 14-16, 2014.*, pages 420–427. International Federation of Automatic Control, 2014.

[44] Srinivas Pinisetty, Yliès Falcone, Thierry Jéron, and Hervé Marchand. Runtime enforcement of regular timed properties. In Yookun Cho, Sung Y. Shin, Sang-Wook Kim, Chih-Cheng Hung, and Jiman Hong, editors, *Symposium on Applied Computing, SAC 2014, Gyeongju, Republic of Korea - March 24 - 28, 2014*, pages 1279–1286. ACM, 2014.

[45] Srinivas Pinisetty, Yliès Falcone, Thierry Jéron, and Hervé Marchand. Tipex: A tool chain for timed property enforcement during execution. In Ezio Bartocci and Rupak Majumdar, editors, *Runtime Verification - 6th International Conference, RV 2015 Vienna, Austria, September 22-25, 2015. Proceedings*, volume 9333 of *Lecture Notes in Computer Science*, pages 306–320. Springer, 2015.

[46] Srinivas Pinisetty, Yliès Falcone, Thierry Jéron, Hervé Marchand, Antoine Rollet, and Omer Nguena-Timo. Runtime enforcement of timed properties revisited. *Formal Methods in System Design*, 45(3):381–422, 2014.

[47] Srinivas Pinisetty, Yliès Falcone, Thierry Jéron, Hervé Marchand, Antoine Rollet, and Omer Landry Nguena-Timo. Runtime enforcement of timed properties. In Shaz Qadeer and Serdar Tasiran, editors, *Runtime Verification, Third International Conference, RV 2012, Istanbul, Turkey, September 25-28, 2012, Revised Selected Papers*, volume 7687 of *Lecture Notes in Computer Science*, pages 229–244. Springer, 2012.

[48] Srinivas Pinisetty, Viorel Preoteasa, Stavros Tripakis, Thierry Jéron, Yliès Falcone, and Hervé Marchand. Predictive runtime enforcement. *Formal Methods in System Design*, 2016. To appear.

[49] Giles Reger, Sylvain Hallé, and Yliès Falcone. Third international competition on runtime verification - CRV 2016. In Yliès Falcone and César Sánchez, editors, *Runtime Verification - 16th International Conference, RV 2016, Madrid, Spain, September 23-30, 2016, Proceedings*, volume 10012 of *Lecture Notes in Computer Science*, pages 21–37. Springer, 2016.

[50] Matthieu Renard, Yliès Falcone, Antoine Rollet, Thierry Jéron, and Hervé Marchand. Optimal enforcement of (timed) properties with uncontrollable events. *Mathematical Structures in Computer Science*, 2017. To appear.

[51] Matthieu Renard, Yliès Falcone, Antoine Rollet, Srinivas Pinisetty, Thierry Jéron, and Hervé Marchand. Enforcement of (timed) properties with uncontrollable events. In Martin Leucker, Camilo Rueda, and Frank D. Valencia, editors, *Theoretical Aspects of Computing - ICTAC 2015 - 12th International Colloquium Cali, Colombia, October 29-31, 2015, Proceedings*, volume 9399 of *Lecture Notes in Computer Science*, pages 542–560. Springer, 2015.

[52] Matthieu Renard, Antoine Rollet, and Yliès Falcone. Runtime enforcement using büchi games. In Hakan Ergdomus and Klaus Havelund, editors, *24th International SPIN Symposium on Model Checking of Software - SPIN 2017, Santa Barbara, USA, July 13-14, 2017, Proceedings*. ACM, 2017. To appear.

[53] Selma Saidi and Yliès Falcone. Dynamic detection and mitigation of DMA races in mpsocs. In *2015 Euromicro Conference on Digital System Design, DSD 2015, Madeira, Portugal, August 26-28, 2015*, pages 267–270. IEEE Computer Society, 2015.

[54] Andrei Voronkov and Margarita V. Korovina, editors. *HOWARD-60: A Festschrift on the Occasion of Howard Barringer's 60th Birthday*. EasyChair, 2014.