Curriculum Vitae

Giovanni Ennio Quattrocchi, Ph.D.

Web: quattrocchi.faculty.polimi.it
GitHub: github.com/gioenn
E-Mail: giovanni.quattrocchi@polimi.it
Phone: (+39) 02 23993707

Giovanni obtained his Ph.D. in Computer Engineering at Politecnico di Milano (POLIMI), and he is currently working as Post-doc Researcher and Adjunct Professor at POLIMI. His research interests mostly focus on the runtime management of distributed systems deployed on Cloud and/or Edge infrastructures. Giovanni is the main responsible and lecturer of two courses (for bachelor and Ph.D. students) at POLIMI.

Employment

02/2019—curr	Adjunct Professor @ POLIMI
11/2017—curr	Post-doc Researcher @ POLIMI
10/2014—curr	Lecturer (500+ hours) of Ph.D., Ms.C., and B.Eng. courses
10/2014—11/2017	Ph.D. Candidate @ POLIMI

Education

01/2018	Ph.D. in Computer Engineering @ POLIMI *Supervisor:* Professor Luciano Baresi *Thesis Title:* Fast and Fine-grained Resource Provisioning for Modern Software Systems
04/2014	Ms.C. in Computer Engineering (110 with honors / 110) @ POLIMI *Supervisor:* Professor Sam Jesus Montalvo Guinea *Thesis Title:* Coordinamento di attuatori in cicli di controllo multi-livello in applicazioni cloud

Visiting

02/2017—05/2017	Visiting researcher @ Imperial College London *Contact:* Prof. Alessandra Russo *Research topic:* Performance optimization of big-data systems
10/2013—12/2013	Visiting student @ University of California San Diego *Contact:* Prof. Ingolf Krueger *Research topic:* Elastic resource provisioning of Cloud applications

Research interests (keywords)

- **Runtime control of distributed systems**: cloud computing, edge computing, control theory, containers, self-adaptive systems, service level agreements, microservices, big-data frameworks, federated machine learning
- **Software engineering for the Blockchain technology**: blockchain applications,

consensus mechanisms, incentives schemes, design patterns, distributed autonomous organization, smart contract verification, tokenization

- **Service continuity**: system automation, digital transformation, machine learning for software engineering, incident management, natural language processing
- **Empirical studies on software architectures**: microservices, infrastructure as code, devops, mining software repositories

Publications

- **Journal papers**[1,2]

1. L. Baresi, A. Leva, and G. Quattrocchi. "Fine-Grained Dynamic Resource Allocation for Big-Data Applications". In: *IEEE Transactions on Software Engineering (TSE)* (2019), 1–15. Presented at ICSE 2020 as Journal First.
   Core: A*, **Scimago: Q1**
2. L Baresi, D. Filgueira, M. Garriga, S. Guinea, and G. Quattrocchi. "A Unified Model for the Mobile-Edge-Cloud Continuum". In: *ACM Transactions on Internet Technology (TOIT)* 19.2 (2019), pp. 1–21.
   Core: B, **Scimago: Q1**
3. Indika Kumara, Paul Mundt, Kamil Tokmakov, Dragan Radolović, Alexander Maslennikov, Román Sosa González, Jorge Fernández Fabeiro, Giovanni Quattrocchi, Kalman Meth, Elisabetta Di Nitto, et al. "SODALITE@ RT: Orchestrating Applications on Cloud-Edge Infrastructures". In: *Journal of Grid Computing* 19.3 (2021), pp. 1–23.
   Core: B, **Scimago: Q2**
4. L. Baresi, M. Bersani, F. Marconi, G. Quattrocchi, and M. G. Rossi. "Using Formal Verification to Evaluate the Execution Time of Spark Applications". In: *Formal Aspect of Computing (FAOC)* 32 (2020), pp. 33–70.
   Core: B, **Scimago: Q3**
5. Luciano Baresi and Giovanni Ennio Quattrocchi. "PAPS: a Serverless Platform for Edge Computing Infrastructures". In: *Frontiers in Sustainable Cities* 3 (2021), p. 58.

- **Conference papers**[3]

6. L. Baresi, G. Denaro, and G. Quattrocchi. "Symbolic Execution-driven Extraction of the Parallel Execution Plans of Spark Applications". In: *Proc. of the 2019 ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*. ACM, 2019, pp. 246–256.
   Core: A*
7. L. Baresi, S. Guinea, A. Leva, and G. Quattrocchi. "A Discrete-Time Feedback Controller for Containerized Cloud Applications". In: *Proc. of the 2016 ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*. ACM, 2016, pp. 217–228.
   Core: A*

[1]Core Journal ranking: `https://www.scimagojr.com`
[2]Scimago Journal ranking: `https://www.scimagojr.com`
[3]Core Conference ranking: `http://portal.core.edu.au/conf-ranks`

8. Luciano Baresi, Giovanni Quattrocchi, and Nicholas Rasi. "Federated Machine Learning as a Self-Adaptive Problem". In: *International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*. IEEE. 2021, pp. 41–47.
   Core: A
9. L. Baresi and G. Quattrocchi. "A Simulation-based Comparison between Industrial Autoscaling Solutions and COCOS for Cloud Applications". In: *Proc. of the 2020 IEEE International Conference on Web Services (ICWS)*. IEEE, 2020, pp. 1–8.
   Core: A
10. L. Baresi and G. Quattrocchi. "COCOS: a Scalable Architecture for Containerized Heterogeneous Systems". In: *Proc. of the 2020 IEEE International Conference On Software Architecture (ICSA)*. IEEE, 2020, pp. 1–11.
    Core: A
11. L. Baresi, G. Quattrocchi, D. Tamburri, and W. Heuvel. "Automated Quality Assessment of Incident Tickets for Smart Service Continuity". In: *Proc. of the 2020 International Conference on Service-Oriented Computing (ICSOC)*. Springer, 2020), pp. 1–8.
    Core: A
12. L. Baresi, D. Filgueira, and G. Quattrocchi. "PAPS: A Framework for Decentralized Self-Management at the Edge". In: *Proc. of the 2019 International Conference on Service-Oriented Computing (ICSOC)*. Springer, 2019, pp. 508–522.
    Core: A
13. L. Baresi, S. Guinea, G. Quattrocchi, F. Seracini, M. Menarini, and I. Krüger. "Comprehensive Resource Management Solution for Web-based Systems". In: *Proc. of the 2014 IEEE International Conference on Autonomic Computing (ICAC)*. IEEE, 2014, pp. 233–239.
    Core: B
14. F. Marconi, G. Quattrocchi, L. Baresi, M. Bersani, and M. G. Rossi. "On the Time Analysis of Big-data Applications". In: *Proc. of the 2018 International Symposium on NASA Formal Methods (NFM)*. Springer, 2018, pp. 315–332.
    Core: C
15. L. Baresi, S. Guinea, G. Quattrocchi, and D. Tamburri. "A Container-based Solution for Efficient Resource Management in the Cloud". In: *Proc. of the 2016 IEEE International Conference on Smart Cloud (SmartCloud)*. IEEE, 2016, pp. 218–223.

- **Workshop and position papers**

16. Kalman Meth, Indika Kumara, and Giovanni Quattrocchi. "Intelligent re-deployment feedback loop for hybrid applications". In: *Proceedings of the 14th ACM International Conference on Systems and Storage*. 2021, pp. 1–1.
17. I. Kumara, G. Quattrocchi, D. Tamburri, and W. Heuvel. "Quality Assurance of Heterogeneous Applications: The SODALITE Approach". In: *European Conference on Service-Oriented and Cloud Computing (ESOCC)*. Springer, 2020, pp. 173–178.
18. L. Baresi, G. Denaro, and G. Quattrocchi. "Big-Data Applications as Self-Adaptive Systems of Systems". In: *Proc. of the 2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2019, pp. 155–162.

19. L. Baresi and G. Quattrocchi. "Towards Vertically Scalable Spark Applications". In: *Proc. of the 2019 International Conference on Parallel and Distributed Computing Workshops (Euro-Par)*. Springer, 2018, pp. 106–118.
20. L. Baresi, S. Guinea, and G. Quattrocchi. "Distributed Coordinated Adaptation of Cloud-based Applications". In: *Proc. of the 2015 International Conference on Software Engineering and Formal Methods Workshops (SEFM)*. Springer, 2015, pp. 215–227.

Teaching

2020—curr	Software Engineering IOL (B.Eng., POLIMI) as *Professor*
2019—curr	Information Technology for Planning, Architectural Design and Built Environment Management (Ph.D., POLIMI) as *Professor*
2019	Software Engineering Methodologies for Security (M.Sc., POLIMI) as *Assistant*
2018—curr	Algorithms and Parallel Computing (B.Eng., POLIMI) as *Assistant*
2018	Modern Cloud Computing (Ph.D., Fondazione Bruno Kessler) as *Lecturer*
2017—curr	Software Engineering (B.Eng., POLIMI) as *Assistant*
2015—curr	Design and Implementation of Mobile Applications (M.Sc., POLIMI) as *Assistant*
2015—curr	Software Engineering (B.Eng., POLIMI) as *Lab Supervisor*
2015—2016	Principles of Programming Languages (M.Sc., POLIMI) as *Assistant*

Supervised Students

- Davide Yi Xian Hu (M.Sc.)
- Luca Terracciano (M.Sc.)
- Nicholas Rasi (M.Sc.)
- Simone Ripamonti (M.Sc.)
- Davide Bertolotti (M.Sc.)
- Matteo Gazzetta (M.Sc.)
- Dimitri Stebliuk (M.Sc.)
- Rodrigo Brechard (B.Eng.)

Events Organization and Projects

2021—curr	PC Member of the MODELS ACM Student Research Competition-track
2020—curr	Organizer of the GAUSS Workshop @ IEEE ISSRE (2 editions)
2020	Session Chair of the of the GAUSS Workshop @ IEEE ISSRE
2020—curr	Organizer of the IEEE Service Hackathon
2019—curr	Participation in the EU funded SODALITE project[4]
2019	Session Chair of the Services Computing of the IEEE International Conference on Services Computing
2018—2019	PC Member of the IEEE International Conference on Services Computing
2018—2019	Participation in the Italian GAUSS project[5]
2014—2017	Participation in the Italian EEB project[6]

Editorial Presence

- Editorial Board Participation

2021—curr	Guest Editor for the IEEE Software journal
2020—curr	Guest Editor for the Journal of Software: Evolution and Process

- Reviewer for Journals
  - ACM Transactions on Autonomous and Adaptive Systems
  - ACM Transactions on Software Engineering and Methodology
  - IEEE Software
  - IEEE Transactions on Cloud Computing
  - IEEE Transactions on Network and Service Management
  - IEEE Transactions on Services Computing
  - IEEE Transactions on Software Engineering
  - Information and Software Technology Journal (Elsevier)
  - Journal of Object Technology
- Reviewer for Conferences
  - IEEE International Conference on Big Data
  - IEEE International Conference on Cloud Computing
  - IEEE International Conference on Services Computing
  - ACM International Conference on Utility and Cloud Computing

Invited Talks and Conference Presentations

2021 Speaker at the International Symposium on Software Engineering for Adaptive and Self-

[4]EU Grant agreement 825480.

[5]PRIN 2015 program - Contract 2015KWREMX.

[6]Edifici A Zero Consumo Energetico In Distretti Urbani Intelligenti - Italian Technology Cluster For Smart Communities - CTN01 00034 594053.

managing Systems (SEAMS)

2020 Speaker at the International Conference on Software Engineer (ICSE)

2020 Speaker at the International Conference on Software Architecture (ICWS)

2020 Speaker at the International Conference on Software Architecture (ICSA)

2019 Speaker at the Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)

2019 Speaker at the International Conference on Service-Oriented Computing (ICSOC)

2018 Speaker at the International Conference on Parallel and Distributed Computing (Euro-Par)

2017 Invited Speaker at the 77th Shonan Meeting on Controlled Adaptation of Self-Adaptive Systems

2016 Speaker at the Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)

2016 Speaker at the IEEE International Conference on Smart Cloud

2015 Speaker at International Conference on Software Engineering and Formal Methods (SEFM)

Professional Services

2021—curr Monitoring Trustee in the legal case between Google and the Italian Competition Authority (Autorità Garante della Concorrenza e del Mercato)[7]

2019—curr Consulting for Trame d'Italia S.r.l. on mobile application development

2018 Consulting for Lineacom S.r.l. (A2A) on Big-data and Analytics

2015—curr Lecturer and Course Organizer of professional courses for several companies including Vodafone, Fastweb, PwC, Unicredit, Samsung, Cariplo, Alcatel, and Cofidis

Main Research Line

One of the main challenges to be addressed in Cloud and Edge computing is how to provision and optimize computational and memory resources allocated to applications in presence of fluctuating workloads, unpredictable peaks of traffic, and unexpected changes on the execution environment. Without carefully provision resources, service providers struggle in satisfying functional and non-functional requirements, usually defined in SLAs (Service Level Agreements).

To avoid resource saturation and unresponsiveness, users dissatisfaction and unnecessary costs, the provisioning of resources must be elastic, that means automatically adapting to changes that could affect the quality of service perceived by users.

State of the art solutions focus on the control of software systems deployed in the cloud by changing the number of allocated virtual machines using mainly heavyweight techniques such as optimization problems or delegating to the system administrators part of the adaptation process (e.g., rule-based approaches). Moreover virtual machines are relatively slow to be provisioned (around six minutes on average) and only available in fixed configurations limiting how fast and precise the adaptation could be.

Giovanni's main research activity aims to study, analyze and evaluate novel technologies and models that enable lightweight, autonomous, fine-grained, precise, and fast elastic resource provisioning for Cloud and Edge software systems.

The technique used to plan and enact the control actions exploits two main pillars: *containers*, a lightweight virtualization technology that enables fast and fine-grained actuation, and *control*

[7] Case A529: https://en.agcm.it/en/media/press-releases/2021/5/A529

theory that provides a theoretical foundation for controlling these systems. Containers can be provisioned in seconds and re-configured in milliseconds while control theory techniques allow to compute the next state of the system in constant time.
This methodology was applied to web, big-data and interactive machine learning applications with the creation of three prototypes, all tested on well-known public cloud infrastructure (e.g., Amazon Web Service, Microsoft Azure). First, *ECoWare* (7. ESEC/FSE 2016) is a system that allows containerized web-applications to scale their resources both at the VM level and at the container level. Furthermore, applications can combine infrastructural adaptation with higher level (such as middleware or application level) adaptation actions. Experiments show that *ECoWare* is able to use less than half of the amount of resources that Amazon's Autoscaling used to control containerized applications without SLA violations.
Second, *dynaSpark* (1. TSE 2019, 6. ESEC/FSE 2019) is dedicated to the control of big-data batch applications. This type of systems manipulate huge quantities of data and users are often interested in quantifying and constraining the execution times (deadlines) for completing single runs. *dynaSpark* is a novel container-based extension to the Spark framework that exploits container technology to provide extremely fine-grained resource allocation. While Spark does not allow users to constrain deadlines, *dynaSpark* allows that and allocates required resources accordingly at runtime. Given that Spark itself does not provide the means to properly control applications, *dynaSpark* extends the core of Spark (written in Scala) by modifying existing components and adding new control-related ones. The evaluation on different well-known benchmark applications witnesses that *dynaSpark* was able to use less resources than unmodified Spark and complete the executions with a less than 2% error in terms of set deadlines.
Finally, *ROMA* (10. ICSA 2020) is an extension of TensorFlow that allows for the control of machine learning applications deployed in inference mode. These applications can be highly parallelized using both multiple CPUs and GPUs.
ROMA controls multiple TensorFlow applications deployed on the same cluster and exploits two centralized schedulers, one for CPUs and one for GPUs, that use dedicated heuristics to schedule incoming requests on a selected device. Given defined SLAs, when application performance needs to be boosted, GPUs are smartly selected as preferred execution device. On each cluster machine, control theoretical planners precisely refine the resource allocation in order to optimize their usage while fulfilling SLAs.
A comprehensive evaluation carried out using real-world machine learning applications shows that *ROMA* minimizes the amount of SLA violations while reducing resource consumption compared to TensorFlow and rule-based approaches.

Open Source Tooling

- *ECoWare* - presented at ESEC/FSE 2016
  `https://github.com/deib-polimi/ecoware`
- *dynaSpark* - published in TSE (2019), presented at ICSE 2020
  `https://github.com/deib-polimi/dynaSpark`
- *ROMA* - main paper under peer review, presented at ICSA 2020
  `https://github.com/SODALITE-EU/refactoring-ct`
- *Resource Allocation Simulator* - presented at ICWS 2020
  `https://github.com/deib-polimi/RAS`

- *PAPS* - presented at ICSOC 2019
  `https://github.com/deib-polimi/PAPS`
- *DAG-ver* - published in FAOC (2020), presented at NFM 2018
  `https://github.com/deib-polimi/DAG-ver`
- *Direct Acyclic Graphs (DAG), labeling using Dilworth's theorem* - used in *DAG-ver*
  `https://github.com/gioenn/dilworth-labeling`