

CURRICULUM VITAE

Bahman Arasteh, Ph.D. Associated Professor in Software Engineering



Address:

Faculty Member, Associated Professor
Department of Software Engineering
Faculty of Engineering and Natural Science
Istinye University, Istanbul, Türkiye

Contacts:

Email: bahman.arasteh@istinye.edu.tr
b_arasteh2001@yahoo.com

Tel: **+90-5013432390**

Postal Code: 34010

ORCID: [0000-0001-5202-6315](https://orcid.org/0000-0001-5202-6315)

RESEARCH INTERESTS

- Search based Software Engineering
- Software Testing
- Optimization Algorithms
- Machine Learning
- Software Dependability and Fault Tolerance
- Software Reverse Engineering

LANGUAGES

Turkish, English, Persian

EDUCATION

- Ph.D.**
2009 – 2014 **Islamic Azad University, Science and Research Branch, Tehran, Iran**
Ph.D. in Computer Engineering, Software Engineering
Dissertation: *Developing Inherently Resilient Software against Soft-error using Algorithm Level Features*
Supervisor: Seyed Gasem Miremadi (Sharif University of Technology, Tehran, Iran)
- M.S.**
2004 – 2007 **Islamic Azad University, Arak Branch, Arak, Iran**
Master of Science in Computer Engineering, Software Engineering
Master's Thesis: *Optimization of Grid Scheduling Algorithm*
Supervisor: Ali Movaghar (Sharif University of Technology, Tehran, Iran)

ACADEMIC EXPERIENCE

- 2021-present** **Department of Software Engineering, Faculty of Engineering and Natural Science, Istinye University, Istanbul Turkiye**
Associated Professor in Computer Engineering Department
- 2020- 2021** **Islamic Azad University, Tabriz Branch**, Tabriz, Azerbaijan Province, Iran
Associated Professor in Computer Engineering Department
- 2014 - 2020** **Islamic Azad University, Tabriz Branch**, Tabriz, Azerbaijan Province, Iran
Assistant Professor in Computer Engineering Department
- 2004 -2014** **Islamic Azad University, Tabriz Branch**, Tabriz, Azerbaijan Province, Iran
Instructor in Computer Engineering Department

ADMINISTRATIVE EXPERIENCE

- 2023** - Program Committee Member, 6th international conference on computational Intelligence in Data Science (ICCIDS 2023), India.
- 2021** - Program Committee Member, 6th National Conference on Distributed Computing and Big Data Processing (DCBDP 2021), Tabriz, Iran, Feb 2021.
- 2018** - Program Committee Member, 4th National Conference on Distributed Computing and Big Data Processing (DCBDP 2018), Tabriz, Iran, May 2018.
- 2014-2016** - Chair of Computer Engineering Department, Islamic Azad University, Tabriz Branch, Tabriz, Azerbaijan Province, Iran
- 2014-2015** - Head of Student Programing-Contest Lab, Islamic Azad University, Tabriz Branch, Tabriz, Azerbaijan Province, Iran
- 2011** - Program Committee Member, 2th National Student Programing Contest, Tabriz, Iran, April 2011.
- 2010** - Program Committee Member, 4th National SAMA Research and Skill Festival, Tabriz, Iran, Jul 2010.

COURSES TAUGHT

• *Graduate:*

- Software Testing
- Search Based Software Engineering
- Optimization Algorithms and its Application in Computer Engineering
- Software Dependability
- Software Architecture
- Artificial Intelligence

• *Undergraduate:*

- Database System Design and Implementation
- Visual C# Programing
- Object Oriented Programing and Design

- Web Programing
- Software Verification and Testing
- Data Structures and Algorithms
- Theory of Finite Automata
- Formal Method in Software Engineering
- Principles of Compiler Design and Implementation

PEER-REVIED PUBLICATIONS

Published Journal Papers:

1. Majidian Z., TaghipourEivazi S., **Arasteh B.**, Shahram Babai, An intrusion detection method to detect denial of service attacks using error-correcting output codes and adaptive neuro-fuzzy inference, Computers and Electrical Engineering, Vol. 106, **2023**, 108600, ISSN 0045-7906, <https://doi.org/10.1016/j.compeleceng.2023.108600>.
2. Gharehchopogh, F.S., Ucan, A., Ibrikci, T., **Arasteh B.**, Slime Mould Algorithm: A Comprehensive Survey of Its Variants and Applications. Arch Computat Methods Eng (**2023**). <https://doi.org/10.1007/s11831-023-09883-3>
3. **Arasteh B.**, Ghanbarzadeh R., Soleimanian Gharehchopogh F., Hosseinalipour A., Generating the Structural Graph-based Model from a Program Source-code using Chaotic Forrest Optimization Algorithm, Wiley Journal of Expert Systems, (**2023**), <https://doi.org/10.1111/exsy.13228>.
4. **Arasteh B.**, Seyyedabbasi A., Rasheed J., Abu-Mahfouz A.M., Program Source-Code Re-Modularization by Using Discretized and Modified Sand Cat Swarm Optimization Algorithm, MDPI symmetry, Vol.15, No. 2, (**2023**), <https://doi.org/10.3390/sym15020401>.
5. **Arasteh B.**, Sefati SS., Halunga S., Fratu O., Allahviranloo T., A Hybrid Heuristic Algorithm Using Artificial Agents for Data Replication Problem in Distributed Systems, Symmetry, (**2023**); Vol. 15, No. 2:487. <https://doi.org/10.3390/sym15020487>
6. Bouyer A., Ahmadi H., **Arasteh B.**, Aghae Z., Ghanbarzadeh R., FIP: A fast overlapping community-based influence maximization algorithm using probability coefficient of global diffusion in social networks, Expert Systems with Applications, Volume 213, Part A, (**2022**), 118869, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2022.118869>.
7. **Arasteh, B.** Clustered design-model generation from a program source code using chaos-based metaheuristic algorithms. Neural Computing & Applications (**2022**). <https://doi.org/10.1007/s00521-022-07781-6>
8. **Arasteh B.**, Abdi M., Bouyer A., Program source code comprehension by module clustering using combination of discretized gray wolf and genetic algorithms, Advances in Engineering Software, Volume 173, (**2022**), 103252, ISSN 0965-9978, <https://doi.org/10.1016/j.advengsoft.2022.103252>.
9. Soleimanian F., Abollahzadeh B., **Arasteh B.**, “An Improved Farmland Fertility Algorithm with Hyper-heuristic Approach for Solving Travelling Salesman Problem: An Improved FFA with Hyper-heuristic Approach for Solving TSP”, Journal of Computer Modeling in Engineering and Sciences, Accepted, September, (**2022**), <https://doi.org/10.32604/cmescs.2023.024172>
10. **Arasteh, B.**, Karimi, M.B. & Sadegi, R. Düzen: generating the structural model from the software source code using shuffled frog leaping algorithm. Neural Computing & Applications (**2022**). <https://doi.org/10.1007/s00521-022-07716-1>
11. **Arasteh, B.**, Imanzadeh, P., Arasteh, K. et al. A Source-code Aware Method for Software Mutation Testing Using Artificial Bee Colony Algorithm. J Electron Test 38, 289–302 (**2022**). <https://doi.org/10.1007/s10836-022-06008-9>
12. **Arasteh, B.**, Hosseini, S.M.J. Traxtor: An Automatic Software Test Suit Generation Method Inspired by Imperialist Competitive Optimization Algorithms. J Electron Test 38, 205–215 (**2022**).

<https://doi.org/10.1007/s10836-022-05999-9>

13. **Arasteh, B** and Solhi, R, "Programming-level and redundancy-free method for enhancing software reliability against transient errors in hardware", International Journal of Reliability, Quality and Safety Engineering, Vol. 29, No. 01, pp. 2150038, 2022, [DOI:10.1142/S0218539321500388](https://doi.org/10.1142/S0218539321500388), 2022.
14. **Arasteh, B**, Fatolahzadeh, A, Kiani, F. Savalan: Multi objective and homogeneous method for software modules clustering. J Softw Evol (2022); Vol. 34, No. 1, [doi:10.1002/smr.2408](https://doi.org/10.1002/smr.2408), 2022.
15. **Arasteh B.**, Sadegi R. and Arasteh K. "ARAZ: A Software Modules Clustering Method Using the Combination of Particle Swarm Optimization and Genetic Algorithms", Intelligent Decision Technologies, Vol. 14, No. 4, pp. 449 – 462, 2020, [DOI: 10.3233/IDT-200070](https://doi.org/10.3233/IDT-200070)
16. Hatami, E., **Arasteh, B.** An efficient and stable method to cluster software modules using ant colony optimization algorithm. J Supercomput 76, 6786–6808 (2020). <https://doi.org/10.1007/s11227-019-03112-0>
17. **Arasteh, B.**, Khosroshahzadeh, S., "Software reliability enhancement against hardware transient errors using inherently reliable data structures", Int J Syst Assur Eng Manag, Vol. 11, 883–898 (2020). <https://doi.org/10.1007/s13198-020-01011-9>
18. Ghaemi, A., **Arasteh, B.**, "SFLA-based heuristic method to generate software structural test data", Journal of Software: Evolution and Process, Vol 32, No 1, 2047-7473, 2020 , <https://doi.org/10.1002/smr.2228>
19. **Arasteh, B.**, Sadegi, R. and Arasteh, K., "Bölen: software module clustering method using the combination of shuffled frog leaping and genetic algorithm", Data Technologies and Applications, Vol. ahead-of-print No. ahead-of-print. 2020, <https://doi.org/10.1108/DTA-08-2019-0138>
20. Shomali, N. and **Arasteh, B.**, "Mutation reduction in software mutation testing using firefly optimization algorithm", Data Technologies and Applications, Vol. 54 No. 4, 461480 ,(2020). <https://doi.org/10.1108/DTA-08-2019-0140>
21. **Arasteh, B.**, "ReDup: A software-based method for detecting soft-error using data analysis", Computers & Electrical Engineering, Vol. 78, 2019, PP 89-107, <https://doi.org/10.1016/j.compeleceng.2019.07.006>.
22. **Arasteh, B.**, Najafi, J., "Programming guidelines for improving software resiliency against soft-errors without performance overhead", Computing, Vol. 100, PP. 971–1003, (2018). <https://doi.org/10.1007/s00607-018-0592-y>
23. **Arasteh, B.**, "A Program-Aware Fault-Injection Method for Dependability Evaluation Against Soft-Error Using Genetic Algorithm", Journal of Circuits, Systems and Computers, Vol. 27, No. 09, 1850144 (2018), <https://doi.org/10.1142/S021812661850144X>
24. Karimi A. Z. and **Arasteh B.**, "An Efficient Method to Generate Test Data for Software Structural Testing Using Artificial Bee Colony Optimization Algorithm", International Journal of Software Engineering and Knowledge Engineering, Vol. 27, No. 06, pp. 951-966 (2017). <https://doi.org/10.1142/S0218194017500358>
25. Keshtgar S. A., **Arasteh B.**, "Enhancing Software Reliability against Soft-Error using Minimum Redundancy on Critical Data", Journal of Computer Network and Information Security, Vol. 5, PP. 21-30, 2017.
26. **Arasteh B.**, "Software Fault-Prediction using Combination of Neural Network and Naive Bayes Algorithm", Journal of Networking Technology, Vol. 9. No. 3, 2018, [DOI: 10.6025/jnt/2018/9/3/94-101](https://doi.org/10.6025/jnt/2018/9/3/94-101)
27. **Arasteh B.**, "Improving the Resiliency of Software Against Soft-Errors Without External Redundancy and Performance Overhead, Journal of Circuits, Systems and Computers, Vol. 26, No. 07, 1750124, (2017). <https://doi.org/10.1142/S0218126617501249>
28. **Arasteh B.**, Bouyer A., Pirahesh S., "An efficient vulnerability-driven method for hardening a program against soft-error using genetic algorithm", Computers & Electrical Engineering, Vol. 48, (2015), PP. 25.43, <https://doi.org/10.1016/j.compeleceng.2015.09.020>.
29. Taghavi Afshord S., Pottosin Y., **Arasteh B.**, "An input variable partitioning algorithm for functional

- decomposition of a system of Boolean functions based on the tabular method", *Discrete Applied Mathematics*, Vol. 185, (2015), PP.s 208-219, <https://doi.org/10.1016/j.dam.2014.12.013>.
30. **Arasteh, B.**, Miremadi, S.G. & Rahmani, A.M. "Developing Inherently Resilient Software Against Soft-Errors Based on Algorithm Level Inherent Features", *Journal of Electron Test*, Vol. 30, PP. 193–212, (2014). <https://doi.org/10.1007/s10836-014-5438-8>.
 31. Bouyer, A., **Arasteh, B.**, "An Adaptable Job Submission System Based on Moderate Price-Adjusting Policy in Market-Based Grids", *Wireless Pers Commun*, Vol. 73, PP. 1573–1590, 2013. <https://doi.org/10.1007/s11277-013-1267-9>.
 32. **Arasteh B.**, Pirahesh S., Zakeri A., "Highly Available and Dependable E-learning Services Using Grid System", *Procedia - Social and Behavioral Sciences*, Vol. 143, 2014, PP. 471-476. DOI: [10.6025/jnt/2018/9/3/94-101](https://doi.org/10.6025/jnt/2018/9/3/94-101)
 33. Chodari Khosrowshahi A., **Arasteh B.**, Taghavi Afshord S., "A New Strategy for Optimizing Energy and Delay in MCSMAC Protocol", *Indian Journal of Science and Technology*, Vol 7, No.11, 2014,

Published Conference Papers:

1. **B. Arasteh**, M. R. Sattari and R. Shokri Kalan, "Fuzuli: Automatic Test Data Generation for Software Structural Testing using Grey Wolf Optimization Algorithm and Genetic Algorithm," 2022 IEEE 20th International Conf on Dependable, Autonomic and Secure Computing, Italy, **2022**, pp. 1-6, doi: 10.1109/DASC/PiCom/CBDCOM/Cy55231.2022.9927968.
2. **Arasteh B.**, Fallah A., Arasteh K., et al., "Cross-site Scripting Attack Detection using Combination of Multi-Layer Perceptron and Naive Bayes Algorithms", *International Conference on Distributed Computing and Big Data Processing*, Tabriz, Iran, **2022**.
3. **Arasteh B.**, Arasteh K., Ahmadzadeh S., Forughifar M., "A Two-Phase Method to Predict the Software Faults using Self-organizing Map and Support-Vector Machine Algorithms", *5th Conference on Distributed Computing and Big Data Processing*, Tabriz, Iran, October 2019.
4. Gharehchopogh F. S., Rezaii R. and **Arasteh B.**, "A new approach by using Tabu search and genetic algorithms in Software Cost estimation," 2015 9th IEEE International Conference on Application of Information and Communication Technologies (AICT), Rostov on Don, Russia, 2015, pp. 113-117, doi: 10.1109/ICAICT.2015.7338528.
5. Nangir A. **Arasteh B.**, "Detection of XSS Attacks using Self Organization Map Algorithm", *5th Conference on Distributed Computing and Big Data Processing*, Tabriz, Iran, October 2019, (In Persian).
6. **Arasteh B.**, Rahmani A. M., Mansoor A. and Miremadi S. G., "Using Genetic Algorithm to Identify Soft-Error Derating Blocks of an Application Program", *2012 15th IEEE Euromicro Conference on Digital System Design*, Izmir, 2012, pp. 359-367, doi: 10.1109/DSD.2012.136.
7. **Arasteh B.** and M. J. Hosseini, "A Dependable and Efficient Scheduling Model for Critical Applications on Grid Systems", *2011 Sixth International Symposium on Parallel Computing in Electrical Engineering*, England, England, 2011, pp. 79-86, doi: 10.1109/PARELEC.2011.24.
8. Zadahmad M., **Arasteh B.**, YousefzadehFard P., "Dynamically choosing the appropriate input control strategy using novel Aspect-Oriented Software Development approach", *9th International Conference on Electronics, Computer and Computation*, Ankara, Turkey, 2012.
9. Sima k., **Arasteh B.**, "Investigation of Mutant Reduction in Software Testing", *4th Conference on Distributed Computing and Big Data Processing*, Tabriz, Iran, October 2018, (In Persian).
10. **Arasteh B.**, Rahmani, Saeed AM., Taghavi S., "A Hybrid Fault Tolerance Model for Reliable Scheduling of Critical Real-Time Applications on Grid Systems", *2010 3rd International Symposium on Parallel Architectures, Algorithms and Programming*; 12/2010, DOI:10.1109/PAAP.2010.71.
11. **Arasteh B.**, Bouyer A., Pirahesh S., "A New Method to Develop Reliable and Efficient Software with a limited Cost", *The 13th annual research conference advancement on business, science and Technology (ARC 2014)*, Istanbul, Turkey, 2014.

12. Pirahesh S., **Arasteh B.**, “A Fault-tolerant and Cost-based Job Scheduling Method for Grid Systems”, Conference on Distributed Computing and Big Data Processing, Tabriz, Iran, 2015.

Published Book and Book Chapters:

1. Hadidi N., Akbari M., **Arasteh B.**, “English for It Students”, Islamic Azad University of Tabriz, ISBN:964-978-10-5673-7, 2020.
2. **Arasteh B.**, Hosseini M.J.(2011), “A Dependable and Efficient Scheduling Model and Fault Tolerance Service for Critical Applications on Grid Systems”, In: Park J.J., Yang L.T., Lee C. (eds) Future Information Technology. Communications in Computer and Information Science, vol 184. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-22333-4_13.
3. ZadahmadJafarlou M., **Arasteh B.**, YousefzadehFard P. (2012), “OO Divide and Conquer Pattern Suitable for Parallel, Grid and Cloud Computing”, In: J. (Jong Hyuk) Park J., Chao HC., S. Obaidat M., Kim J. (eds) Computer Science and Convergence. Lecture Notes in Electrical Engineering, Vol. 114. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-2792-2_46.
4. **Arasteh B.**, ZadahmadJafarlou M., Hosseini M.J. (2012),” A Dynamic and Reliable Failure Detection and Failure Recovery Services in the Grid Systems”, In: J. (Jong Hyuk) Park J., Chao HC., S. Obaidat M., Kim J. (eds) Computer Science and Convergence. Lecture Notes in Electrical Engineering, Vol. 114. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-2792-2_47

ACADEMIC SERVICE

Editor:

- **Coordinating Editor in Springer Journal of Electronic Test (ISI JCR)**

Reviewer:

- Elsevier Journal of Advances in Software Engineering (ISI JCR)
- Elsevier Journal of Information Science (ISI JCR)
- Springer Journal of Electronic Test (ISI JCR)
- Wiley Journal of Software: Evolution and Process (ISI JCR)
- Springer Journal of Fuzzy Systems
- Elsevier Journal of Expert systems with Applications
- IET Software (ISI JCR)
- Elsevier Journal of Computers and Electrical Engineering (ISI JCR)
- Springer Journal of Super Computing (ISI JCR)
- Journal of Software Engineering and Knowledge Engineering (ISI JCR)
- Springer Journal of System Assurance Engineering and Management (ISI WOS)
- Elsevier Journal of Microprocessors and Microsystem (ISI JCR)
- Tabriz Journal of Electrical Engineering (ISC)
- Journal of Soft Computing and Information Technology (ISC)

Advisor:

- Advisor in National Project in Türkiye (TUBITAK 1001)

SOME OF THE CURRENT AND FORMER SUPERVISED PHD STUDENTS

	Student Name	Thesis Title
1	Syed Mohamad, Javad Hosseini, 2018	Reducing the Cost of Software Mutation Test through Static Analysis of Source Code and Sensitive Path Detection
2	Zeynab Asghari, 2020	Classification of Program Instructions based on Error Propagating Features for Identifying Equivalent Mutants in Software Mutation Testing
3	Ali Chodari	Community Detection in Social Networks using Complex Network

SOME OF THE CURRENT AND FORMER SUPERVISED MASTER STUDENTS

	Student Name	Thesis Title
1	Behzad Hosseinlu, 2015	Software Fault Prediction using Executive Path Analysis and Composition of Artificial Neural Network and Support Vector Machine
2	Ahmad Dashti, 2015	Software Structural Test-Data Generation using combination of ANT Colony and Learning Automata
3	Akram Beheshti, 2015	A Method for Software Cost Estimation using Neuro-fuzzy and PSO
4	Parisa Imanzadeh, 2016	Reducing the Cost of Software Mutation Testing using Artificial Bee Colony
5	Keyvan Ebrahimi, 2016	A Test data Generation Method for Software Structural Testing Using Improved Genetic Algorithm with Bee Colony Algorithm
6	Mina Emamalipour, 2016	Quality of Service Aware Service Selection by using the Ant-lion Optimizer Algorithm
7	Yusef Movahedian 2017	Reducing the Cost of Mutation test by Decreasing the Number of Mutants through Identifying the Sensitive Part of Program
8	Amir Ghaemi, 2017	Software Structural Test case Generation Using Shuffled Frog-Leaping Algorithm
9	Hossein Jalabi, 2017	Software Fault Localization using Program Dependence Graph and Neural Network
10	Said Rahimi, 2017	Improving the Security of Database Systems against SQL Injection Attacks Using the Hash function and Web Services
11	Arman Abdollahpur, 2018	Energy aware Virtual Machine Allocation in Cloud computing using Particle Swarm Optimization and Simulated annealing
12	Maryam Zeyni, 2018	Test Cases Prioritization in Software Regression Test using Chaotic Theory
13	Vahid Hosseinzadeh, 2018	Software Test Oracle Design Using Deep Learning
14	Neda Mostafavi Khah, 2018	Using Chaotic based Two-phase Particle Swarm Optimization Algorithm to Increase the QoS of Cloud Composite Services
15	Elmira Hatami, 2019	Clustering Software Modules using Ant Colony Algorithm
16	Keyvan Arasteh, 2019	Reducing the Cost of Software Structural Testing using Program Slicing

17	Ahmad Fattolahzaeh, 2019	software Module Clustering as a Multi Objective Optimization Problem using Homogeneous Clustering Approach
18	Elmira Hatami, 2020	Clustering Software Modules using Ant Colony Algorithm
19	Behzad Gadimi, 2020	Reducing the Cost of Software Mutation Testing by Clustering the Overlapped Statement-level Mutants
20	Ayub Khani, 2020	SQL Injection Attack Detection using Learning Vector Quantization (LVQ) Algorithm
21	Mohamad Kohi, 2021	Detecting Security-vulnerability of web based software against XSS attacks using particle swarm optimization Algorithm
22	Behzad Amir Fallahi, 2021	Cross-site Scripting (XSS) Attack Detection using Combination of Multi-Layer Perceptron and Naive Bayes Algorithm
23	Milad Khadir, 2021	Software Test Data Generation using Black Widow Optimization Algorithm

Skills and Expertise

- C# and Dot Net Core
- Software Test
- Software API Testing with Postman Tool
- Software Security Testing with Selenium, OWASP ZAP and Acunetix
- Databases Design and Implementation
- Desktop and Web App. Development (Back-end and Front End)
- Implementation of Artificial intelligence algorithms in different programming languages and applications
- C++

Some of Software Development Projects (Tools)

- Development of Automatic Test-Generation Tool for Structural Testing of C# program units (Startup Project)
- Development of Automatic Mutation Test Tool for Mutation Testing of PHP program Source-code (Startup Project)
- Development of Automatic Reverse Engineering Tool for Extracting Structural Models from the Source code of Software
- Development of Software Modules Clustering Tool (Startup Project)

Some of Software Development Projects (Applications)

- Development of Accounting App. for the Agency of Insurance Company
- Development of Accounting and Management App. for Dental Clinic Treatment Files
- Accounting App. for Chain Stores