Formal models of software-defined networks
Feng, H.

Citation
Feng, H. (2024, December 3). *Formal models of software-defined networks*. Retrieved from https://hdl.handle.net/1887/4170508

Note: To cite this publication please use the final published version (if applicable).

Bibliography

[1] Elvira Albert, Miguel Gomez-Zamalloa, Miguel Isabel, Albert Rubio, Matteo Sammartino, and Alexandra Silva. Actor-based model checking for software-defined networks. *Journal of Logical and Algebraic Methods in Programming*, 118:100617, 2021.

[2] Elvira Albert, Miguel Gómez-Zamalloa, Albert Rubio, Matteo Sammartino, and Alexandra Silva. SDN-actors: Modeling and verification of SDN programs. In *International Symposium on Formal Methods*, pages 550–567. Springer, 2018.

[3] Rajeev Alur and David L Dill. A theory of timed automata. *Theoretical computer science*, 126(2):183–235, 1994.

[4] Carolyn Jane Anderson, Nate Foster, Arjun Guha, Jean-Baptiste Jeannin, Dexter Kozen, Cole Schlesinger, and David Walker. NetKAT: Semantic foundations for networks. *Acm sigplan notices*, 49(1):113–126, 2014.

[5] Farhad Arbab. Reo: a channel-based coordination model for component composition. *Mathematical Structures in Computer Science*, 14(3):329–366, 2004.

[6] Farhad Arbab, Christian Koehler, Ziyan Maraikar, Young-Joo Moon, and José Proença. Modeling, testing and executing reo connectors with the eclipse coordination tools. *Tool demo session at FACS*, 8, 2008.

[7] Farhad Arbab and JJMM Rutten. A coinductive calculus of component connectors. In *Recent Trends in Algebraic Development Techniques: 16th International Workshop, WADT*, volume 2755, pages 34–55. Springer, 2002.

[8] Youssef Arbach, David S. Karcher, Kirstin Peters, and Uwe Nestmann. Dynamic causality in event structures. *Logical Methods in Computer Science*, 14(1), 2018.

[9] Ebrahim Ardeshir-Larijani, Alireza Farhadi, and Farhad Arbab. Simulation of hybrid reo connectors. In *2020 CSI/CPSSI International Symposium on Real-Time and Embedded Systems and Technologies (RTEST)*, pages 1–10. IEEE, 2020.

[10] Abdelhadi Azzouni, Nguyen Thi Mai Trang, Raouf Boutaba, and Guy Pujolle. Limitations of openflow topology discovery protocol. In *2017 16th annual mediterranean Ad hoc networking workshop (Med-Hoc-Net)*, pages 1–3. IEEE, 2017.

[11] Jos CM Baeten. A brief history of process algebra. *Theoretical Computer Science*, 335(2-3):131–146, 2005.

[12] Christel Baier, Tobias Blechmann, Joachim Klein, Sascha Klüppelholz, and Wolfgang Leister. Design and verification of systems with exogenous coordination using vereofy. In *Leveraging Applications of Formal Methods, Verification, and Validation: 4th International Symposium on Leveraging Applications, ISoLA 2010, Part II 4*, pages 97–111. Springer, 2010.

[13] Christel Baier, Marjan Sirjani, Farhad Arbab, and Jan Rutten. Modeling component connectors in Reo by constraint automata. *Science of computer programming*, 61(2):75–113, 2006.

[14] Thomas Ball, Nikolaj Bjørner, Aaron Gember, Shachar Itzhaky, Aleksandr Karbyshev, Mooly Sagiv, Michael Schapira, and Asaf Valadarsky. Vericon: Towards verifying controller programs in software-defined networks. *SIGPLAN Not.*, 49(6):282–293, 2014.

[15] Gerd Behrmann, Alexandre David, and Kim G Larsen. A tutorial on UPPAAL. *Formal methods for the design of real-time systems*, pages 200–236, 2004.

[16] Johan Bengtsson, Kim Guldstrand Larsen, Fredrik Larsson, Paul Pettersson, and Wang Yi. UPPAAL - a tool suite for automatic verification of real-time systems. In *Hybrid Systems III: Verification and Control, Proceedings of the DIMACS/SYCON Workshop on Verification and Control of Hybrid Systems*, volume 1066 of *LNCS*, pages 232–243. Springer, 1995.

[17] Kevin Benton, L Jean Camp, and Chris Small. Openflow vulnerability assessment. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 151–152. ACM, 2013.

[18] Benedikt Bollig, Joost-Pieter Katoen, Carsten Kern, Martin Leucker, Daniel Neider, and David R Piegdon. libalf: The automata learning framework. In *International Conference on Computer Aided Verification*, pages 360–364. Springer, 2010.

[19] Marcello Bonsangue, Dave Clarke, and Alexandra Silva. A model of context-dependent component connectors. *Science of Computer Programming*, 77(6):685–706, 2012.

[20] Neil Briscoe. Understanding the osi 7-layer model. *PC Network Advisor*, 120(2):13–15, 2000.

[21] Stephen Brookes. Full abstraction for a shared-variable parallel language. *Information and Computation*, 127(2):145–163, 1996.

[22] Olav Bunte, Jan Friso Groote, Jeroen J. A. Keiren, Maurice Laveaux, Thomas Neele, Erik P. de Vink, Wieger Wesselink, Anton Wijs, and Tim A. C. Willemse. The mCRL2 toolset for analysing concurrent systems - improvements in expressivity and usability. In *Tools and Algorithms for the Construction and Analysis of Systems - 25th International Conference, TACAS*, volume 11428 of *LNCS*, pages 21–39. Springer, 2019.

[23] Georgiana Caltais, Sophie Linnea Guetlein, and Stefan Leue. Causality for General LTL-definable Properties. In Bernd Finkbeiner and Samantha Kleinberg, editors, *Proceedings 3rd Workshop on formal reasoning about Causation, Responsibility, and Explanations in Science and Technology*, volume 286 of *EPTCS*, pages 1–15, 2018.

[24] Georgiana Caltais, Hossein Hojjat, Mohammad Reza Mousavi, and Hünkar Can Tunç. DyNetKAT: An Algebra of Dynamic Networks. In *International Conference on Foundations of Software Science and Computation Structures*, pages 184–204. Springer, 2022.

[25] Georgiana Caltais, Mohammad Reza Mousavi, and Hargurbir Singh. Causal reasoning for safety in Hennessy Milner logic. *Fundamenta Informaticae*, 173(2-3):217–251, 2020.

[26] Georgiana Caltais and Can Olmezoglu. Counterfactual causality in networks. *arXiv preprint arXiv:2211.00758*, 2022.

[27] Dexian Chang, Ning Zhu, and Yingjie Yang. Security analysis of sdn access control protocol based on proverif. In *Proceedings of IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, pages 1155–1159. IEEE, 2021.

[28] Guillermina Cledou, José Proença, Bernhard H. C. Sputh, and Eric Verhulst. Hubs for VirtuosoNext: Online verification of real-time coordinators. *Science of Computer Programming*, 203:102566, 2021.

[29] Rúben Cruz and José Proença. Reolive: Analysing connectors in your browser. In *Software Technologies: Applications and Foundations: STAF 2018 Collocated Workshops, Toulouse, France, June 25-29, 2018, Revised Selected Papers*, pages 336–350. Springer, 2018.

[30] Ahmed El-Hassany, Jeremie Miserez, Pavol Bielik, Laurent Vanbever, and Martin Vechev. SDNRacer: Concurrency Analysis for Software-Defined Networks. In *Proceedings of the 37th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '16, page 402–415. ACM, 2016.

[31] R Enns, M Bjorklund, J Schoenwaelder, and A Bierman. Rfc 6241: Network configuration protocol (netconf), 2011.

[32] Hui Feng. Link to the code and results. https://github.com/githuifeng/Reo2Promela.git.

[33] Hui Feng, Farhad Arbab, and Marcello Bonsangue. A Reo model of Software Defined Networks. In *Formal Methods and Software Engineering: 21st International Conference on Formal Engineering Methods, ICFEM 2019, Shenzhen, China, November 5–9, 2019, Proceedings 21*, pages 69–85. Springer, 2019.

[34] Hui Feng, Marcello Bonsangue, and Benjamin Lion. From symbolic constraint automata to Promela. *Journal of Logical and Algebraic Methods in Programming*, 128:100794, 2022.

[35] Bernd Finkbeiner, Manuel Gieseking, Jesko Hecking-Harbusch, and Ernst-Rüdiger Olderog. Model checking data flows in concurrent network updates (full version). *ArXiv*, abs/1907.11061, 2019.

[36] Nate Foster, Dexter Kozen, Matthew Milano, Alexandra Silva, and Laure Thompson. A coalgebraic decision procedure for NetKAT. In *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 343–355. ACM, 2015.

[37] Nate Foster, Dexter Kozen, Matthew Milano, Alexandra Silva, and Laure Thompson. A coalgebraic decision procedure for NetKAT. In *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2015, Mumbai, India, January 15-17, 2015*, pages 343–355. ACM, 2015.

[38] Open Networking Foundation. Openflow overview.

[39] Open Networking Foundation. Openflow specification.

[40] Rob Gerth, Doron Peled, Moshe Y Vardi, and Pierre Wolper. Simple on-the-fly automatic verification of linear temporal logic. In *Protocol Specification, Testing and Verification XV: Proceedings of the Fifteenth IFIP WG6. 1 International Symposium on Protocol Specification, Testing and Verification, Warsaw, Poland, June 1995*, pages 3–18. Springer, 1996.

[41] Fatemeh Ghassemi, Samira Tasharofi, and Marjan Sirjani. Automated mapping of Reo circuits to constraint automata. *Electronic Notes in Theoretical Computer Science*, 159:99–115, 2006.

[42] Paul Göransson and Chuck Black. *Software Defined Networks: A Comprehensive Approach*. Morgan Kaufmann, 2016.

[43] A. Groce, S. Chaki, D. Kroening, and O. Strichman. Error explanation with distance metrics. *International Journal on Software Tools for Technology Transfer (STTT)*, 8(3), 2006.

[44] A. Groce and W. Visser. What went wrong: Explaining counterexamples. In *Workshop on Software Model Checking (SPIN)*, Lecture Notes in Computer Science 2648, pages 121–135. Springer, 2003.

[45] Evangelos Haleplidis, Jamal Hadi Salim, Joel M Halpern, Susan Hares, Kostas Pentikousis, Kentaro Ogawa, Weiming Wang, Spyros Denazis, and Odysseas Koufopavlou. Network programmability with forces. *IEEE Communications Surveys & Tutorials*, 17(3):1423–1440, 2015.

[46] Joseph Y. Halpern. A Modification of the Halpern-Pearl Definition of Causality. In Qiang Yang and Michael J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 3022–3033. AAAI Press, 2015.

[47] Joseph Y. Halpern and Judea Pearl. Causes and explanations: A structural-model approach: Part 1: Causes. In Jack S. Breese and Daphne Koller, editors, *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 194–202. Morgan Kaufmann, 2001.

[48] Gerard Holzmann. *Spin Model Checker, the: Primer and Reference Manual*. Addison-Wesley Professional, first edition, 2003.

[49] Gerard J. Holzmann. The model checker spin. *IEEE Transactions on software engineering*, 23(5):279–295, 1997.

[50] Gerard J Holzmann. An analysis of bitstate hashing. *Formal methods in system design*, 13(3):289–307, 1998.

[51] Fei Hu. *Network Innovation through Openflow and SDN*. Crc Press, 2014.

[52] Fei Hu, Qi Hao, and Ke Bao. A survey on software-defined network and openflow: From concept to implementation. *IEEE Communications Surveys & Tutorials*, 16(4):2181–2206, 2014.

[53] Ali Hussein, Imad H Elhajj, Ali Chehab, and Ayman Kayssi. Sdn verification plane for consistency establishment. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 519–524. IEEE, 2016.

[54] Mohammad Izadi, Marcello Bonsangue, and Dave Clarke. Büchi automata for modeling component connectors. *Software & Systems Modeling*, 10:183–200, 2011.

[55] Mohammad Izadi and Marcello M Bonsangue. Recasting constraint automata into Büchi automata. In *Theoretical Aspects of Computing-ICTAC 2008: 5th International Colloquium, Istanbul, Turkey, September 1-3, 2008. Proceedings 5*, pages 156–170. Springer, 2008.

[56] Daniel Jackson. Alloy: a language and tool for exploring software designs. *Communications of the ACM*, 62(9):66–76, 2019.

[57] Sung-Shik T. Q. Jongmans, Tobias Kappé, and Farhad Arbab. Constraint automata with memory cells and their composition. *Science of Computer Programming*, 146:50–86, 2017.

[58] Sung-Shik TQ Jongmans and Farhad Arbab. Overview of thirty semantic formalisms for reo. *Scientific Annals of Computer Science*, 22(1), 2012.

[59] Sung-Shik TQ Jongmans and Farhad Arbab. Modularizing and specifying protocols among threads. *arXiv preprint arXiv:1302.6333*, 2013.

[60] Miyoung Kang, Eun-Young Kang, Dae-Yon Hwang, Beom-Jin Kim, Ki-Hyuk Nam, Myung-Ki Shin, and Jin-Young Choi. Formal modeling and verification of SDN-openflow. In *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*, pages 481–482. IEEE, 2013.

[61] Young-Mi Kim, Miyoung Kang, and Jin-Young Choi. Formal specification and verification of firewall using TLA+. In *Proceedings of the International Conference on Security and Management (SAM)*, pages 247–251. IEEE, 2017.

[62] Keith Kirkpatrick. Software-defined networking. *Communications of the ACM*, 56(9):16–19, 2013.

[63] Stephen Cole Kleene. Representationof events in nerve nets and finite automata. *CE Shannon and J. McCarthy*, 1951.

[64] Rowan Klöti, Vasileios Kotronis, and Paul Smith. Openflow: A security analysis. In *2013 21st IEEE International Conference on Network Protocols (ICNP)*, pages 1–6. IEEE, 2013.

[65] Natallia Kokash, Christian Krause, and Erik P de Vink. Data-aware design and verification of service compositions with Reo and mCRL2. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 2406–2413. ACM, 2010.

[66] Natallia Kokash, Christian Krause, and Erik P. de Vink. Reo + mCRL2: A framework for model-checking dataflow in service compositions. *Formal Aspects of Computing*, 24(2):187–216, 2012.

[67] Dexter Kozen. Kleene algebra with tests. *ACM Transactions on Programming Languages and Systems*, 19(3):427–443, 1997.

[68] Dexter Kozen. Automata on guarded strings and applications. *Matemática Contemporânea*, 24:117–139, 2003.

[69] Dexter Kozen. NetKAT – a formal system for the verification of networks. In Jacques Garrigue, editor, *Programming Languages and Systems*, pages 1–18. Springer, 2014.

[70] Marta Kwiatkowska, Gethin Norman, and David Parker. Prism 4.0: Verification of probabilistic real-time systems. In *Computer Aided Verification: 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings 23*, pages 585–591. Springer, 2011.

[71] Peter Ladkin and Karsten Loer. Analysing Aviation Accidents Using WB-Analysis – an Application of Multimodal Reasoning. In *AAAI Spring Symposium*. AAAI, 1998.

[72] Leslie Lamport. *Specifying Systems, The TLA+ Language and Tools for Hardware and Software Engineers*. Addison-Wesley, 2002.

[73] Petr Lapukhov, Ariff Premji, and Jon Mitchell. Use of bgp for routing in large-scale data centers. Technical report, Internet Engineering Task Force (IETF), 2016.

[74] Florian Leitner-Fischer and Stefan Leue. Causality checking for complex system models. In Roberto Giacobazzi, Josh Berdine, and Isabella Mastroeni, editors, *Verification, Model Checking, and Abstract Interpretation, 14th International Conference, VMCAI*, volume 7737 of *Lecture Notes in Computer Science*, pages 248–267. Springer, 2013.

[75] D. Lewis. Causation. *Journal of Philosophy*, 70:556–567, 1973.

[76] D. Lewis. *Counterfactuals*. Blackwell Publishers, 1973.

[77] Benjamin Lion, Samir Chouali, and Farhad Arbab. Compiling protocols to Promela and verifying their LTL properties. In *Proceedings of MODELS 2018 Workshops*, volume 2245 of *CEUR Workshop Proceedings*, pages 31–39. CEUR-WS.org, 2018.

[78] Nancy Lynch. I/o automata: A model for discrete event systems. In *Annual Conference on Information Sciences and Systems*, pages 29–38. Princeton University, Princeton, N.J, 1998.

[79] Mani Prashanth Varma Manthena, Niels LM van Adrichem, Casper van den Broek, and Fernando Kuipers. An sdn-based architecture for network-as-a-service. In *Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft)*, pages 1–5. IEEE, 2015.

[80] John C Martin. *Introduction to Languages and the Theory of Computation*, volume 4. McGraw-Hill NY, 1991.

[81] Jedidiah McClurg, Hossein Hojjat, Nate Foster, and Pavol Černỳ. Event-driven network programming. *ACM SIGPLAN Notices*, 51(6):369–385, 2016.

[82] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: Enabling innovation in campus networks. *Computer Communication Review*, 38(2):69–74, 2008.

[83] Dmitrij Melkov and Sarunas Paulikas. Security benefits and drawbacks of software-defined networking. In *2021 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)*, pages 1–4. IEEE, 2021.

[84] Anders Møller. dk.brics.automaton – Finite-State Automata and Regular Expressions for Java, 2021. `http://www.brics.dk/automaton/`.

[85] Seyedeh Elham Mousavi Bafrooi. Specification and implementation of workflow control patterns in reo. Master's thesis, University of Waterloo, 2006.

[86] Nadya El Moussaid, Ahmed Toumanari, and Maryam El Azhari. Security analysis as software-defined security for SDN environment. In *Proceedings of 2017 Fourth International Conference on Software Defined Systems (SDS)*, pages 87–92. IEEE, 2017.

[87] Gilbert N. Nde and Rahamatullah Khondoker. Sdn testing and debugging tools: A survey. In *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*, pages 631–635. IEEE, 2016.

[88] Mogens Nielsen, Gordon D. Plotkin, and Glynn Winskel. Petri nets, event structures and domains, part I. *Theoretical Computer Science*, 13:85–108, 1981.

[89] José Proença, Dave Clarke, Erik De Vink, and Farhad Arbab. Dreams: A Framework for Distributed Synchronous Coordination. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 1510–1515. ACM, 2012.

[90] Mark Reitblatt, Marco Canini, Arjun Guha, and Nate Foster. Fattire: Declarative Fault Tolerance for Software-Defined Networks. In *Proceedings of the 2nd ACM SIGCOMM workshop on Hot topics in Software Defined Networking*, pages 109–114. ACM, 2013.

[91] Yakov Rekhter, Tony Li, and Susan Hares. Rfc 4271: A border gateway protocol 4 (bgp-4), 2006.

[92] Manos Renieres and Steven P Reiss. Fault localization with nearest neighbor queries. In *18th IEEE International Conference on Automated Software Engineering, 2003. Proceedings.*, pages 30–39. IEEE, 2003.

[93] Justus Rischke and Hani Salah. Software-defined networks. In *Computing in Communication Networks*, pages 107–118. Elsevier, 2020.

[94] Natali Ruchansky and Davide Proserpio. A (not) nice way to verify the openflow switch specification: formal modelling of the openflow switch using alloy. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, page 527–528. ACM, 2013.

[95] Cole Schlesinger, Michael Greenberg, and David Walker. Concurrent netcore: From policies to pipelines. In *Proceedings of the 19th ACM SIGPLAN international conference on Functional programming*, pages 11–24. ACM, 2014.

[96] Hanan Shabana and Mikhail V. Volkov. Optimal synchronization of partial deterministic finite automata. *CoRR*, abs/2002.01045, 2020.

[97] Rob Sherwood, Michael Chan, Adam Covington, Glen Gibb, Mario Flajslik, Nikhil Handigol, Te-Yuan Huang, Peyman Kazemian, Masayoshi Kobayashi, Jad Naous, et al. Carving research slices out of your production networks with openflow. *ACM SIGCOMM Computer Communication Review*, 40(1):129–130, 2010.

[98] Nitin Shukla, Mayank Pandey, and Shashank Srivastava. Formal modeling and verification of software-defined networks: A survey. *Transactions on Emerging Telecommunications Technologies*, 29(5):2082, 2019.

[99] Alexandra Silva. A specification language for reo connectors. In Farhad Arbab and Marjan Sirjani, editors, *Fundamentals of Software Engineering - 4th IPM International Conference, FSEN 2011, Tehran, Iran, April 20-22, 2011, Revised Selected Papers*, volume 7141 of *Lecture Notes in Computer Science*, pages 368–376. Springer, 2011.

[100] Robert Soulé, Shrutarshi Basu, Parisa Jalili Marandi, Fernando Pedone, Robert Kleinberg, Emin Gun Sirer, and Nate Foster. Merlin: A language for provisioning network resources. In *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, pages 213–226. ACM, 2014.

[101] Alireza. Souri, Monire Norouzi, Parvaneh Asghari, Amir Masoud Rahmani, and Gazaleh Emadi. A systematic literature review on formal verification of software-defined networks. *Transactions on Emerging Telecommunications Technologies*, 31(2):3788, 2020.

[102] William Stallings. *Handbook of computer-communications standards; Vol. 1: the open systems interconnection (OSI) model and OSI-related standards*. Macmillan Publishing Co., Inc., 1987.

[103] Mevlut Serkan Tok and Mehmet Demirci. Security analysis of SDN controller-based DHCP services and attack mitigation with DHCPguard. *Computers & Security*, 109:102394, 2021.

[104] Moshe Y Vardi and Pierre Wolper. Reasoning about infinite computations. *Information and computation*, 115(1):1–37, 1994.

[105] Jean Philippe Vasseur and Jean Louis Le Roux. Path computation element (pce) communication protocol (pcep). Technical report, Network Working Group, 2009.

[106] Jana Wagemaker, Nate Foster, Tobias Kappé, Dexter Kozen, Jurriaan Rot, and Alexandra Silva. Concurrent NetKAT. In *European Symposium on Programming*, pages 575–602. Springer, Cham, 2022.

[107] Raniyah Wazirali, Rami Ahmad, and Suheib Alhiyari. Sdn-openflow topology discovery: an overview of performance issues. *Applied Sciences*, 11(15):6999, 2021.

[108] Shuangqing Xiang, Marcello Bonsangue, and Huibiao Zhu. Pdnet: A programming language for software-defined networks with vlan. In *International Conference on Formal Engineering Methods*, pages 203–218. Springer, 2019.

[109] Shuangqing Xiang, Huibiao Zhu, Xi Wu, Lili Xiao, Marcello M. Bonsangue, Wanling Xie, and Lei Zhang. Modeling and verifying the topology discovery mechanism

of openflow controllers in software-defined networks using process algebra. *Science of Computer Programming*, 187:102343, 2020.

[110] Shuangqing Xiang, Huibiao Zhu, Lili Xiao, and Wanling Xie. Modeling and Verifying TopoGuard in OpenFlow-Based Software Defined Networks. In *2018 International Symposium on Theoretical Aspects of Software Engineering (TASE)*, pages 84–91. IEEE Computer Society, 2018.

[111] Lily Yang, Todd A. Anderson, Ram Gopal, and Ram Dantu. Forwarding and Control Element Separation (ForCES) Framework. RFC 3746, 2004.

[112] Yuan Yu, Panagiotis Manolios, and Leslie Lamport. Model checking TLA+ specifications. In *Advanced Research Working Conference on Correct Hardware Design and Verification Methods*, pages 54–66. Springer, 1999.

[113] Vladimir A. Zakharov, R. L. Smelyansky, and Eugene V. Chemeritsky. A formal model and verification problems for software defined networks. *Automatic Control and Computer Sciences*, 48(7):398–406, 2014.

[114] Andreas Zeller. *Why Programs Fail: A Guide to Systematic Debugging*. Morgan Kaufmann, 2009.

[115] Yongyue Zhang, Xiangyang Gong, Yannan Hu, Wendong Wang, and Xirong Que. SDNMP: Enabling SDN management using traditional NMS. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 357–362. IEEE, 2015.