Research - Distributed Systems Correct by Construction

Fernando Luís Dotti

The correct construction of Distributed Systems is not a trivial task due to concurrency, time and partial failures. Along with fault tolerance, this is an additional approach in building reliable systems [29]. We have worked on methods and techniques to build correct concurrent distributed systems. In a very succinct way, our work in this sense has undergone the introduction of a formal language for reactive message passing systems and, based on models in this language and suitable translations, we enabled the use of different existing analysis tools. Our article [10] describes this approach.

We adopted Graph Grammars (GGs) [30] and proposed a formal specification method, which we call object-based graphical grammars (OBGGs) [11]. Although formal, GGs are considered intuitive and easy to learn. Graphs are a very natural way to explain complex situations intuitively. Graphical rules can complement each other to capture the dynamic aspects of systems. The resulting notion of Graph Grammar [31, 30, 32] generalizes Chomsky grammars of strings for graphs. Due to their declarative nature, GGs are suitable for specification of competing systems, and has already been used for such cite HB3.

Object-Based Graphing Grammars (OBGGs) [11] follows the object based paradigm, familiar to most users. The language itself is a restricted form of Graph Grammars and captures the main abstractions to represent reactive distributed systems: communication occurs through the passage of messages; state changes are local and competing; there is no limitation on delays for processing as well as for message delivery, characterizing the asynchronous computing model [33].

Functional analysis of OBGG models is supported by model verification [12, 14, 10] using a transformation to PROMELA. One approach for the partial analysis of OBGG models is presented in [15]. Quantitative analysis of OBGG models is possible using various means. If delay probability distributions are assigned to the messages, OBGG models can be translated into discrete event simulation models. Both a kernel and a library for defining simulation entities from OBGG were proposed [16, 17]. Markovian models can be generated from OBGGs through a transformation [18] for stochastic automata networks [34]. If we restrict our models and allow temporal assumptions about message delays, a transformation from OBGGs to Timed Automata [35] allows the analysis of deadlines for receiving and processing messages. In addition to these methods of analysis, there is also the possibility of generating code for execution through a transformation to the Java programming language [17]. The modeling of high performance applications in OBGGs is possible through a mapping to cluster environments [19] (C ++ code using MPI (Message Passing Interface) [36]).

In addition, [20] and [21] introduce the representation of classic failure models for distributed systems in OBGG models, allowing reasoning on a distributed system in the presence of such faults. According to classical ideas in the literature of fault-tolerant distributed systems [37], the representation of fault behaviors may take place through a model transformation step. The formal specification of distributed and fault-tolerant systems has also been the subject of contributions using other methods and languages, as in [25, 26, 27, 28].

Using the methods and tools mentioned above, a framework has been defined to assist in the development of concurrent and distributed systems [13]. A tool to aid in the modeling and reasoning of OBGG systems was developed [22]. Several models were defined and analyzed using OBGGs: mobile code applications [11], a fault detector [20], active networks [23], distributed election in a ring [24], dinner philosophers [12], readers and writers [14], among others.

As you can see, our approach is heavily based on OBGG transformations for several environments and target languages. In [10] we proved the correctness of the transformation from OBGG to PROMELA, central to the above work and which served as the basis for several other mappings mentioned. This proof consisted of demonstrating the semantic compatibility of the generated PROMELA model, described by the transformation, in relation to the original OBGG model.

References

[1] E. Alchieri, F. Dotti, O. M. Mendizabal, and F. Pedone, "Reconfiguring parallel state machine replication," in *SRDS*, 2017.

[2] E. Alchieri, F. Dotti, and F. Pedone, "Early Scheduling in Parallel State Machine Replication," *ArXiv e-prints: 1805.05152*, 2018.

[3] O. M. Mendizabal, R. T. S. Moura, F. L. Dotti, and F. Pedone, "Efficient and deterministic scheduling for parallel state machine replication," in *IPDPS*, 2017.

[4] O. Mendizabal, P. J. Marandi, F. Dotti, and F. Pedone, "Recovery in parallel state-machine replication," in *OPODIS*, 2014.

[5] O. M. Mendizabal, F. L. Dotti, and F. Pedone, "High performance recovery for parallel state machine replication," in *ICDCS*, 2017.

[6] P. Coelho, T. C. Jr., A. Bessani, F. Dotti, and F. Pedone, "Byzantine fault-tolerant atomic multicast," in *48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2018, Luxemburg, June 26-29, 2018*, pp. 39–50, 2018.

[7] O. M. Mendizabal, F. L. Dotti, and F. Pedone, "Analysis of checkpointing overhead in parallel state machine replication," in *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, (New York, NY, USA), pp. 534–537, ACM, 2016.

[8] O. Mendizabal and F. L. Dotti, "Model checking the deferred update replication protocol," in *SBRC – Simpósio Brasileiro de Redes de Computadores*, pp. 995–1008, 2013.

[9] A. Corradini, L. Ribeiro, F. L. Dotti, and O. M. Mendizabal, "A formal model for the deferred update replication technique," in *Trustworthy Global Computing - 8th International Symposium, TGC 2013, Buenos Aires, Argentina, August 30-31, 2013, Revised Selected Papers*, pp. 235–253, 2013.

[10] L. Ribeiro, O. M. dos Santos, F. L. Dotti, and L. Foss, "Correct transformation: From object-based graph grammars to promela," *Sci. Comput. Program.*, vol. 77, no. 3, pp. 214–246, 2012.

[11] F. L. Dotti and L. Ribeiro, "Specification of mobile code systems using graph grammars," in *4th International Conference on Formal Methods for Open Object-Based Distributed Systems*, vol. 177 of *IFIP Conference Proceedings*, (USA), pp. 45–63, Kluwer, 2000.

[12] F. L. Dotti, L. Foss, L. Ribeiro, and O. M. Santos, "Especificação e verificação formal de sistemas distribuídos," in *17º Simpósio Brasileiro de Engenharia de Software*, (Brasil), pp. 225–240, 2003. (In portuguese).

[13] L. Ribeiro, F. L. Dotti, and R. Bardohl, "A formal framework for the development of concurrent object-based systems," in *Formal Methods in Software and Systems Modeling*, vol. 3393 of *Lecture Notes in Computer Science (LNCS)*, pp. 385–401, Springer, 2005.

[14] O. M. Santos, F. L. Dotti, and L. Ribeiro, "Verifying object-based graph grammars," *Eletronic Notes in Theoretical Computer Science*, vol. 109, pp. 125–136, 2004.

[15] F. Dotti, L. Ribeiro, O. dos Santos, and F. Pasini, "Verifying object-based graph grammars: An assume-guarantee approach," *Software and Systems Modeling*, vol. 5, pp. 289–311, September 2006.

[16] B. Copstein, M. C. Móra, and L. Ribeiro, "An environment for formal modeling and simulation of control systems," in *33rd Annual Simulation Symposium*, (USA), pp. 74–82, IEEE Computer Society, 2000.

[17] F. L. Dotti, L. M. Duarte, B. Copstein, and L. Ribeiro, "Simulation of mobile applications," in *Communication Networks and Distributed Systems Modeling and Simulation Conference*, (USA), pp. 261–267, SCS, 2002.

[18] O. M. Mendizabal, F. L. Dotti, and L. Ribeiro, "Stochastic object-based graph grammars," *Electronic Notes in Theoretical Computer Science*, vol. 184, pp. 151–170, 2007.

[19] F. Pasini and F. L. Dotti, "Code generation for parallel applications modelled with object-based graph grammars," *Electronic Notes on Theoretical Computer Science*, vol. 184, pp. 113–131, 2007.

[20] F. L. Dotti, L. Ribeiro, and O. M. Santos, "Specification and analysis of fault behaviours using graph grammars," in *Applications of Graph Transformations with Industrial Relevance (AGTIVE 2003)*, vol. 3062 of *Lecture Notes in Computer Science (LNCS)*, pp. 120–133, Springer, 2004.

[21] F. L. Dotti, O. M. Mendizabal, and O. M. dos Santos, "Verifying fault-tolerant distributed systems using object-based graph grammars," in *Dependable Computing, Second Latin-American Symposium (LADC 2005)*, vol. 3747 of *Lecture Notes in Computer Science (LNCS)*, pp. 80–100, Springer, 2005.

[22] F. L. Dotti, L. M. Duarte, L. Foss, L. Ribeiro, D. Russi, and O. M. Santos, "An environment for the development of concurrent object-based applications," *Eletronic Notes in Theoretical Computer Science*, vol. 127-1, pp. 3–13, 2005.

[23] L. Duarte and F. Dotti, "Development of an active network architecture using mobile agents - a case study," Tech. Rep. TR-043, FACIN - PPGCC - PUCRS, 2004.

[24] F. L. Dotti, L. Foss, L. Ribeiro, and O. M. Santos, "Verification of object-based distributed systems," in *6th International Conference on Formal Methods for Open Object-based Distributed Systems*, vol. 2884 of *Lecture Notes in Computer Science (LNCS)*, pp. 261–275, Springer, 2003.

[25] F. L. Dotti, A. Iliasov, L. Ribeiro, and A. B. Romanovsky, "Modal systems: Specification, refinement and realisation," in *Formal Methods and Software Engineering, 11th International Conference on Formal Engineering Methods, ICFEM 2009, Rio de Janeiro, Brazil, December 9-12, 2009. Proceedings*, pp. 601–619, 2009.

[26] A. Iliasov, A. Romanovsky, and F. L. Dotti, "Structuring specifications with modes," in *2009 Fourth Latin-American Symposium on Dependable Computing*, pp. 81–88, Sept 2009.

[27] F. L. Dotti and L. Ribeiro, "Modeling communication semantics for distributed systems in event-b.," in *Workshop de Testes e Tolerância a Falhas*, pp. 101–114, 2012.

[28] L. Oleksinski, C. Correa, F. L. Dotti, and A. Sales, "A ctl model checker for stochastic automata networks," in *Quantitative Evaluation of Systems* (K. Joshi, M. Siegle, M. Stoelinga, and P. R. D'Argenio, eds.), (Berlin, Heidelberg), pp. 286–289, Springer Berlin Heidelberg, 2013.

[29] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, pp. 11–33, Jan 2004.

[30] G. Rozenberg, ed., *Handbook of graph grammars and computing by graph transformations, volume 1: foundations*. World Scientific Publishing Co., 1997.

[31] H. Ehrig, "Introduction to the Algebraic Theory of Graph Grammars," in *1st Graph Grammar Workshop*, vol. 73 of *Lecture Notes in Computer Science (LNCS)*, pp. 1–69, Springer, 1979.

[32] H. Ehrig, G. Engels, H.-J. Kreowski, and G. Rozenberg, eds., *Handbook of graph grammars and computing by graph transformation: vol. 2: applications, languages, and tools.* World Scientific Publishing Co., 1999.

[33] R. Guerraoui, M. Hurfin, A. Mostefaoui, R. Oliveira, M. Raynal, A. Schiper, and S. S. S. Krakowiak, "Consensus in Asynchronous Distributed Systems: A Concise Guided Tour," in *Advances in Distributed Systems, Advanced Distributed Computing: From Algorithms to Systems*, Lecture Notes in Computer Science (LNCS), pp. 33–47, Springer, 1999. Sacha Krakowiak, Santosh K. Shrivastava (Eds.).

[34] B. Plateau and K. Atif, "Stochastic automata network of modeling parallel systems," *IEEE Transactions on Software Engineering*, vol. 17, no. 10, pp. 1093–1108, 1991.

[35] L. R. Leonardo Michelon, Simone André da Costa, "Formal specification and verification of real-time systems using graph grammars," *Journal of the Brazilian Computer Society*, vol. 13, no. 4, pp. 51–68, 2007.

[36] M. Snir, S. W. Otto, D. W. Walker, J. Dongarra, and S. Huss-Lederman, *MPI: The Complete Reference.* Cambridge, MA, USA: MIT Press, 1995.

[37] F. C. Gärtner, "Transformational approaches to the specification and verification of fault-tolerant systems: formal background and classification," *Journal of Universal Computer Science*, vol. 5, no. 10, pp. 668–692, 1999.