Din, Crystal Chang; **Owe, Olaf**

A sound and complete reasoning system for asynchronous communication with shared futures. (English) Zbl 1371.68190

J. Log. Algebr. Methods Program. 83, No. 5-6, 360-383 (2014).

Summary: Distributed and concurrent object-oriented systems are difficult to analyze due to the complexity of their concurrency, communication, and synchronization mechanisms. We consider the setting of concurrent objects communicating by asynchronous method calls. The future mechanism extends the traditional method call communication model by facilitating sharing of references to futures. By assigning method call result values to futures, third party objects may pick up these values. This may reduce the time spent waiting for replies in a distributed environment. However, futures add a level of complexity to program analysis, as the program semantics becomes more involved.

This paper presents a Hoare style reasoning system for distributed objects based on a general concurrency and communication model focusing on asynchronous method calls and futures. The model facilitates invariant specifications over the locally visible communication history of each object. Compositional reasoning is supported, and each object may be specified and verified independently of its environment. The presented reasoning system is proven sound and (relatively) complete with respect to the given operational semantics.

MSC:

68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)
03B70 Logic in computer science
68N30 Mathematical aspects of software engineering (specification, verification, metrics, requirements, etc.)
68Q55 Semantics in the theory of computing

Keywords:

distributed systems; compositional reasoning; Hoare logic; concurrent objects; operational semantics; communication history

Software:

Java Jr; Multilisp; Spec#

Full Text: DOI

References:

[1] International Telecommunication Union, Open distributed processing - reference model, parts 1-4, (Jul. 1995), ISO/IEC Geneva, Tech. Rep.

[2] Ahern, A.; Yoshida, N., Formalising Java RMI with explicit code mobility, Theor. Comput. Sci., 389, 3, 341-410, (2007) · Zbl 1132.68020

[3] O.-J. Dahl, O. Owe, Formal methods and the RM-ODP, Tech. Rep. Research Report 261, Dept. of Informatics, Univ. of Oslo, Full version of a paper presented at NWPT'98: Nordic Workshop on Programming Theory, Turku, 1998.

[4] Baker, H. G.; Hewitt, C., The incremental garbage collection of processes, (Proceedings of the 1977 Symposium on Artificial Intelligence and Programming Languages, (1977), ACM New York, NY, USA), 55-59

[5] Halstead, R. H., Multilisp: a language for concurrent symbolic computation, ACM Trans. Program. Lang. Syst., 7, 4, 501-538, (1985) · Zbl 0581.68037

[6] Liskov, B. H.; Shrira, L., Promises: linguistic support for efficient asynchronous procedure calls in distributed systems, (Wise, D. S., Proc. SIGPLAN Conference on Programming Language Design and Implementation (PLDI'88), (1988), ACM Press), 260-267

[7] Yonezawa, A.; Briot, J.-P.; Shibayama, E., Object-oriented concurrent programming in ABCL/1, Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA'86), SIGPLAN Not., 21, 11, 258-268, (1986)

[8] Johnsen, E. B.; Owe, O., An asynchronous communication model for distributed concurrent objects, Softw. Syst. Model., 6, 1, 35-58, (2007)

[9] Din, C. C.; Dovland, J.; Owe, O., Compositional reasoning about shared futures, (Eleftherakis, G.; Hinchey, M.; Holcombe, M., Proc. International Conference on Software Engineering and Formal Methods (SEFM'12), LNCS, vol. 7504, (2012), Springer-Verlag), 94-108 · Zbl 1315.68192

[10] Broy, M.; Stølen, K., Specification and development of interactive systems, Monographs in Computer Science, (2001), Springer-Verlag · Zbl 0981.68115

[11] Hoare, C. A.R., Communicating sequential processes, International Series in Computer Science, (1985), Prentice Hall · Zbl 0637.68007

[12] Dahl, O.-J., Object-oriented specifications, (Research Directions in Object-Oriented Programming, (1987), MIT Press Cambridge, MA, USA), 561-576

[13] Dahl, O.-J., Verifiable programming, International Series in Computer Science, (1992), Prentice Hall New York, NY · Zbl 0790.68005

[14] Ábrahám, E.; Grabe, I.; Grüner, A.; Steffen, M., Behavioral interface description of an object-oriented language with futures and promises, J. Log. Algebr. Program., 78, 7, 491-518, (2009) · Zbl 1187.68130

[15] Jeffrey, A. S.A.; Rathke, J., Java jr.: fully abstract trace semantics for a core Java language, (Proc. European Symposium on Programming, LNCS, vol. 3444, (2005), Springer-Verlag), 423-438 · Zbl 1108.68349

[16] Alpern, B.; Schneider, F. B., Defining liveness, Inf. Process. Lett., 21, 4, 181-185, (1985) · Zbl 0575.68030

[17] Din, C. C.; Dovland, J.; Johnsen, E. B.; Owe, O., Observable behavior of distributed systems: component reasoning for concurrent objects, J. Log. Algebr. Program., 81, 3, 227-256, (2012) · Zbl 1247.68184

[18] Full ABS modeling framework (mar 2011), deliverable 1.2 of project FP7-231620, (HATS), available at

[19] Dovland, J.; Johnsen, E. B.; Owe, O.; Steffen, M., Lazy behavioral subtyping, J. Log. Algebr. Program., 79, 7, 578-607, (2010) · Zbl 1204.68072

[20] Din, C. C.; Owe, O., Compositional and sound reasoning about active objects with shared futures, (Feb. 2014), Dept. of Informatics, University of Oslo, FAC J., submitted for publication, available at

[21] Apt, K. R., Ten years of Hoare's logic: a survey — part I, ACM Trans. Program. Lang. Syst., 3, 4, 431-483, (1981) · Zbl 0471.68006

[22] Barnett, M.; Leino, K. R.M.; Schulte, W., The $\operatorname{Spec}\^\sharp$ programming system: an overview, (Proceedings of the 2004 International Conference on Construction and Analysis of Safe, Secure, and Interoperable Smart Devices, CASSIS'04, (2005), Springer-Verlag Berlin, Heidelberg), 49-69

[23] Owe, O., Axiomatic treatment of processes with shared variables revisited, Form. Asp. Comput., 4, 4, 323-340, (1992) · Zbl 0748.68040

[24] Soundararajan, N., A proof technique for parallel programs, Theor. Comput. Sci., 31, 1-2, 13-29, (1984) · Zbl 0543.68010

[25] Leino, K. M.; Müller, P., A verification methodology for model fields, (Sestoft, P., Programming Languages and Systems, Lecture Notes in Computer Science, vol. 3924, (2006), Springer Berlin, Heidelberg), 115-130 · Zbl 1178.68348

[26] Agha, G.; Frolund, S.; Kim, W.; Panwar, R.; Patterson, A.; Sturman, D., Abstraction and modularity mechanisms for concurrent computing, IEEE Parallel Distrib. Technol., 1, 2, 3-14, (1993)

[27] Morandi, B.; Bauer, S. S.; Meyer, B., SCOOP - a contract-based concurrent object-oriented programming model, (Müller, P., Advanced Lectures on Software Engineering, LASER Summer School 2007/2008, LNCS, vol. 6029, (2008), Springer), 41-90

[28] Falkner, K. E.K.; Coddington, P. D.; Oudshoorn, M. J., Implementing asynchronous remote method invocation in Java, (6th Australian Conference on Parallel and Real-Time Systems, (1999), Springer-Verlag), 22-34 · Zbl 0961.68511

[29] Ahrendt, W.; Dylla, M., A system for compositional verification of asynchronous objects, Sci. Comput. Program., 77, 12, 1289-1309, (2012) · Zbl 1264.68050

[30] Dahl, O.-J., Can program proving be made practical?, (Amirchahy, M.; Néel, D., Les Fondements de la Programmation, (1977), Institut de Recherche d'Informatique et d'Automatique Toulouse, France), 57-114

[31] Dovland, J.; Johnsen, E. B.; Owe, O., Verification of concurrent objects with asynchronous method calls, (Proceedings of the IEEE International Conference on Software Science, Technology \& Engineering (SwSTE'05), (2005), IEEE Computer Society Press), 141-150

[32] Dovland, J.; Johnsen, E. B.; Owe, O., Observable behavior of dynamic systems: component reasoning for concurrent objects, (Goldin, D.; Arbab, F., Proc. Workshop on the Foundations of Interactive Computation (FInCo'07), Electr. Notes Theor. Comput. Sci., vol. 203, (2008), Elsevier), 19-34 · Zbl 1277.68056

[33] Soundararajan, N., Axiomatic semantics of communicating sequential processes, ACM Trans. Program. Lang. Syst., 6, 4, 647-662, (1984) · Zbl 0542.68013

[34] de Boer, F. S.; Clarke, D.; Johnsen, E. B., A complete guide to the future, (de Nicola, R., Proc. 16th European Symposium on Programming (ESOP'07), LNCS, vol. 4421, (2007), Springer-Verlag), 316-330

[35] de Boer, F. S., A Hoare logic for dynamic networks of asynchronously communicating deterministic processes, Theor. Comput. Sci., 274, 3-41, (2002) · Zbl 0992.68026

[36] Nordio, M.; Calcagno, C.; Müller, P.; Meyer, B., Soundness and completeness of a program logic for eiffel, (2009), ETH Zurich, Tech. Rep. 617

[37] Din, C. C.; Owe, O.; Bubel, R., Runtime assertion checking and theorem proving for concurrent and distributed sys-

tems, (Proceedings of the 2nd Intl. Conf. on Model-Driven Engineering and Software Development, Modelsward'14, (2014), SCITEPRESS), 480-487