A Brief Research Statement

Manish Gaur
director.cas@aktu.ac.in

Abstract

This research statement briefly describes the area, objectives and the results obtained of the the research work carried out during recent research studies of the author. It also summarises the possible future directions of this research. This statement is supported by a brief bibliography to make the line of research more clear.

Contents

1 Introduction

This research statement primarily focuses on both, the research work done with a emphasis on the results obtained and an outline of the possible future work. It basically sums up the research work done by the author during his post doctoral, doctoral and master studies. Additionally, it gives a brief insight about the author's current research as well. This statement is organised in five sections. Section 2 is a short description of the research done while doing Master's thesis at IIT Delhi, India. Section 3 briefly describes the research work done while pursuing PhD degree at the University of Sussex, UK. Section 4 is about research in collaboration with the School of Computing Science at University of Glasgow, UK. Section 5 is about the ongoing research and a tentative future plan. Section 5 is the conclusion. The statement is supported by a brief bibliography which is pointer to the related works described in Sections 2,3,4 and 5. This also indicates the precise area of research of the author.

2 Previous research work : Deciding Efficiency Prebisimulations

Research in process algebra has focused on the use of behavioural relations such as equivalences and refinement orderings as a basis for establishing system correctness. In the process algebraic framework, specifications and implementations both are defined in the same language; the intuition is that a specification describes the desired high level behaviour, while the implementation details the proposed means for achieving this behaviour. One then uses an appropriate equivalence or preorder to establish that an implementation behaves as defined in the specification. In the case of equivalence based reasoning, conformance means "has the same behaviour as"; in this case an implementation is correct if its behaviour is indistinguishable from that of its specification. Refinement (or preorder) relations, on the other hand, typically embody a notion of comparison: an implementation conforms to (or refines) a specification if the behaviour of the former is "at least as good as" that stipulated by the specification. The benefits of such process algebraic approaches include the following:

- Users as well as testing and verification tools work within a single formalism for specification and implementation.
- The algebra provides explicit support for *compositional* specification and implementation, allowing the specification (implementation) of a system to be built up from the specification (implementation) of its components.
- Specifications include information about what is disallowed as well as what is allowed.

Consequently, a number of different process algebras have been studied [54, 41], and a variety of different equivalences and refinement relations capturing different aspects of behaviour have been developed. Refinements could come in several flavours. One of the earliest refinement relations [41] in the process algebra framework related refinement to the level of indeterminary in the specification i.e. a more determinate process was considered a refinement of a less determinate one in terms of behaviour. Other notions of refinement such as action refinement yield various other preorders. Theories of efficiency preorders for concurrent systems have been described in various papers [7, 5, 6]. Our primary aim in this thesis was to study and implement refinement relations called efficiency preorders within a process algebra framework.

One such method is efficiency prebisimulation for processes [7, 5, 6]. It is based on the simple idea of, essentially, counting the number of internal moves by a process. We discover that it can be incorporated within the general frame work of bisimulation, to obtain a mathematically tractable preorder, which in common with the standard notions of bisimulation equivalence, is sensitive to the branching structure of processes.

The theory of efficiency prebisimulation has been developed in papers [5, 7, 6]. The largest precongruences contained in these preorders have also been characterised and axiomatized for finite processes. However, in the context of verification methodology it is more fruitful to regard these preorders as particular refinement relations. Due to the state explosion problem in verifying concurrent systems, it makes more sense to adopt a top-down methodology in both the development and verification of concurrent systems.

The theme of this work is to describe a method for deciding efficiency prebisimulation for processes. There exist very good algorithms [61, 45]and tools to construct strong bisimulation on labeled transition systems. Many equivalence checking problems on labelled transition systems may be reduced to the problem of constructing strong bisimulation. In any labeled transition system of finite state processes if n is the total number of states and m be the number of transitions then Paige and Tarjan [61] gave an $O(m\ \log(n))$ solution to the generalised partitioning problem on some relation E on states of FSPs. Kanellakis and Smolka studied the problem of equivalence checking of CCS expression and gave $O(m\ \log(n) + n)$ and $O(n^2 m\ \log(n) + mn^\alpha)$ time algorithms [45] for strong bisimulation and weak bisimulation respectively. Here the smallest such α known is 2.376. We proposed a method running in $O(mn^3 + n^2 m \log(n) + mn^2)$ time complexity for deciding efficiency prebisimulation[34, 33].

The related work in this line of research can be referenced at [18, 6, 5, 19, 45, 61, 54, 41, 7, 20, 30].

3 Recent research work: A Routing Calculus

We model a distributed network with routers [75, 48, 43, 74] acting as an active component in determining the quality of service [16, 12, 79, 44, 63, 11] of the network. This is done by selecting the path of connected and communicating processes out of many possible paths of communication. Our model may be considered as an extension of the asynchronous distributed pi-calculus (ADpi) [38]. We believe that such models help in prototyping the routing algorithms in context of large networks and reasoning about them while abstracting away the excessive details. Being general, the model may also be applied to demonstrate the role of routers in determining the quality of services of the network. We provide two models of distributed networks where computations are described explicitly in the presence of routers. However these models differ in method of updating the routing tables about the newly created computing agents.

We show that both the routing languages [32], after abstracting away the details of routers and paths, are reduction equivalent upto structural equivalence to a specification which is similar to ADpi [38]. For one of the models we compare two instances of a distributed network, called configurations [32] to show that one is at least as efficient as the other on the basis of three key properties. We also define a bisimulation [6, 38, 69, 55, 66, 67] based preorder between configurations on the basis of different costs of reduction over a labeled transition system [38, 69, 55, 66]. We justify our choice of bisimulation based preorder by proving that it coincides with an observational preorder primarily defined in terms of three key properties. Further, we prove that this observational preorder can also be recovered from the bisimulation based preorder.

In addition to the above we describe a new variation[35] on the pi-calculus [55] in which the processes incur a cost in doing a communication along channels. Processes operate relative to a cost function defined over the set of channel names to the set of integers. We justify this calculi by proving that a notion of observation based preorder between processes with respect to their cost functions can be recovered from a bisimulation based preorder and vice-versa. This theory is useful in formalising the routing calculus.

This related work on this line of research can be referenced at [58, 6, 2, 13, 8, 78, 60, 37, 71, 27, 28, 38, 69, 55, 75, 48, 43, 1, 53, 14, 49, 68, 47, 35, 31, 40, 36, 64, 73, 74, 9, 10, 16, 21, 12, 79, 44, 65, 63, 50, 3, 57, 62, 72, 77, 11, 26, 22, 70, 56, 76, 42, 32].

4 Post doctoral research

In our recent research we described three calculi. One, named as π_{cost} [32, 35], is basically an extension of the asynchronous pi-calculus [38] to incorporate the cost of a pi-calculus computation in some cost framework. This frame work is based on the type setting of typed asynchronous pi-calculus [40, 69, 27, 28, 38]. Further we described two routing calculi named as, DR_{π}^{ω} and DR_{π} [32], with the intention of modeling a distributed network to demonstrate the cost of communication between the communicating processes. The cost of communication is the number of hops (router) a value propagating message crosses before delivering it to the destination in a network of routers. In fact the value propagating messages used in these models closely resemble the IP packet in TCP/IP [75, 48, 73, 9, 10, 21, 12] model of networks. However we only use the names (addresses) of source and destination nodes and data in messages unlike real IP packets where lots of other information is contained in it. The crucial role of routers in determining the quality of communication services in a distributed network is demonstrated in both the calculi. Based upon these works, this research can be taken forward in following two directions:

1. to extend the basic description of π_{cost} to incorporate a more general frame work of cost settings which accounts for the cost of computation too. Some work in this direction has already being done in [35]. This at one hand explores the new possibilities of pi-calculus extensions with respect to applications in a distributed network settings and on the other it forms a theoretical basis for developing new versions of routing calculi with a underlying network infrastructure which is closer to the actual distributed networks.
2. to develop new variants of recently developed calculi DR_{π}^{ω} and DR_{π} to take
   (a) mobility of nodes into account. The work may be extended on the lines of asynchronous distributed pi-calculus [38].
   (b) failures of nodes and links into account on the lines of [27, 28] is planned.
   (c) amortised cost of communication into account to have a more realistic notion of cost of computation than in DR_{π}^{ω} and DR_{π}. This extension may be done along the lines of [47].
3. stochastic extension of process algebras is developed to add quantification to process algebra models [4, 39, 23, 24, 52, 17, 25]. Probability can be used to model many sources of uncertainty in network. It would be interesting to consider development of a stochastic extension of the routing calculi.

During my tenure at School of Computing, University of Glasgow, UK (October 2012-January 2013) we have already obtained results for DR_{π}^{ω} in amortised bisumulation based proof methods. My current research work is progressing along 3 which is explained briefly in the next section.

5 Current Research and Future Plans

The modern distributed systems have not only functional requirements (i.e. absence of deadlock, livelock etc.) but also have non-functional requirements (i.e. security, reliability, performance, Quality of Service(QoS)). These requirements are probabilistic in nature because of uncertainty [59]. The methods for checking their correctness and analyze their performance is at very primitive stage. In the last few decades, formal verification techniques such as process algebras [55, 38, 35] offer a powerful and rigorous approach for establishing the correctness of computer systems. Routing calculi [31, 32, 51] is a language, an elaboration of asynchronous distributed π-calculus [38]. The classical process algebras were concerned with the functional aspects of the concurrent systems. Stochastic extensions of process algebras is developed to add quantification to process algebra models [29, 46]. Probability can be used to model many sources of uncertainty in network [59]. We have developed a stochastic extension of the routing calculi , *ttR*. The new concept is that the router has the probabilistic choices for routing the message along communication link between the routers. We define a formal model which is described as three level syntactic categories; each describes the processes, nodes where the processes reside and routers where the nodes are located. Processes at nodes can communicate via routers. Router determines the paths between communicating processes. The paths are selected using probability distribution thus adding quantification to it. We believe that such models help in prototyping the probabilistic routing algorithms/protocols in the large distributed networks and reasoning about them while abstracting away the excessive details. We justify our model by showing its reduction equivalence with a specification, "A Simple Probabilistic Broadcast Language(PBL)" [15, 4]. Next, we describe three touchstone properties [32,38] on the reduction semantics to compare the two instances of same distributed networks where messages follow different routes based upon different probability functions. Further, we justify our choice of touchstone property by recovering it through a bisimulation based preorder [32, 38] over the label transition system of the calculi. Upon proper documentation and publications we plan to take this research in combining the features described in 2(a) and 2(b).

There is a considerable literature available on this line of proposed research. Some recent ongoing works by Robin Milner on http://www.cl.cam.ac.uk/~rm135/uam-theme.html and Matthew Hennessy on http://www.cs.tcd.ie/Matthew.Hennessy/onlinepubs.html (A calculus for costed computation) are helpful in determining the direction and relevance of this research.

6 Conclusion

In this report I started with a brief description of my previous research work and later also covered the research carried out during my doctoral and subsequent studies. The main idea was to illustrate the line of research as well the work done so far without going into the fine technical details. Further I described, in very short, a possible future research directions too. This clearly is along the lines of my previous work.

References

[1] Gustavo Alonso, Fabio Casati, Harumi A. Kuno, and Vijay Machiraju. *Web Services - Concepts, Architectures and Applications*. Data-Centric Systems and Applications. Springer, 2004.

[2] Roberto M. Amadio and Sanjiva Prasad. Modelling ip mobility. In *CONCUR*, pages 301–316, 1998.

[3] Ross J. Anderson, Francesco Bergadano, Bruno Crispo, Jong-Hyeon Lee, Charalampos Manifavas, and Roger M. Needham. A new family of authentication protocols. *Operating Systems Review*, 32(4):9–20, 1998.

[4] Matthew Hennessy Andrea Cerone. Modelling probabilistic wireless networks. In Holger Giese and Grigore Rosu, editors, *Formal Techniques for Distributed Systems*, volume 7273 of *Lecture Notes in Computer Science*, pages 135–151. Springer Berlin Heidelberg, 2012.

[5] S. Arun-Kumar and Matthew Hennessy. An efficiency preorder for processes. In *TACS*, pages 152–175, 1991.

[6] S. Arun-Kumar and Matthew Hennessy. An efficiency preorder for processes. *Acta Inf.*, 29(8):737–760, 1992.

[7] S. Arun-Kumar and V Natarajan. Conformance: A precongruence close to bisimilarity. In *Springer Workshop in Computer Science Series*, 1995.

[8] Franco Barbanera, Michele Bugliesi, Mariangiola Dezani-Ciancaglini, and Vladimiro Sassone. A calculus of bounded capacities. In *ASIAN*, pages 205–223, 2003.

[9] Lawrence S. Brakmo, Sean W. O'Malley, and Larry L. Peterson. Tcp vegas: New techniques for congestion detection and avoidance. In *SIGCOMM*, pages 24–35, 1994.

[10] Lawrence S. Brakmo and Larry L. Peterson. Tcp vegas: End to end congestion avoidance on a global internet. *IEEE Journal on Selected Areas in Communications*, 13(8):1465–1480, 1995.

[11] Andrew T. Campbell, Geoff Coulson, and David Hutchison. Flow management in a quality of service architectures. In *HPN*, pages 201–218, 1994.

[12] Andrew T. Campbell and S. Keshav. Quality of service in distributed systems. *Computer Communications*, 21(4):291–293, 1998.

[13] Luca Cardelli and Andrew D. Gordon. Mobile ambients. *Theor. Comput. Sci.*, 240(1):177–213, 2000.

[14] Giuseppe Castagna, Nils Gesbert, and Luca Padovani. A theory of contracts for web services. In *POPL*, pages 261–272, 2008.

[15] Andrea Cerone and Matthew. A simple probabilistic broadcast language, 2012.

[16] Tee Hiang Cheng, A. Campbell, and Klara Nahrstedt. Building qos into distributed systems. *Computer Communications*, 22(5):493–494, 1999.

[17] Giovanni Chiola, Claude Dutheillet, Giuliana Franceschinis, and Serge Haddad. Stochastic well-formed colored nets and symmetric modeling applications, 1993.

[18] Rance Cleaveland and Matthew Hennessy. Testing equivalence as a bisimulation equivalence. *Formal Asp. Comput.*, 5(1):1–20, 1993.

[19] Rance Cleaveland, Joachim Parrow, and Bernhard Steffen. The concurrency workbench: A semantics-based tool for the verification of concurrent systems. *ACM Trans. Program. Lang. Syst.*, 15(1):36–72, 1993.

[20] Rance Cleaveland and Oleg Sokolsky. Equivalence and preorder checking for finite state systems. In *Handbook of process algebra*. Elsevier, 2001.

[21] Douglas Comer, editor. *Internetworking with TCP/IP - Principles, Protocols, and Architectures, Fourth Edition*. Prentice-Hall, 2000.

[22] Alan J. Demers, Srinivasan Keshav, and Scott Shenker. Analysis and simulation of a fair queueing algorithm. In *SIGCOMM*, pages 1–12, 1989.

[23] Yuxin Deng, Rob van Glabbeek, Matthew Hennessy, and Carroll Morgan. Characterising testing preorders for finite probabilistic processes. *Logical Methods in Computer Science*, 4(4:4):1–33, 2008.

[24] Yuxin Deng, Rob van Glabbeek, Matthew Hennessy, and Carroll Morgan. Testing finitary probabilistic processes (extended abstract). In *Proceedings of the 20th International Conference on Concurrency Theory*, volume 5710 of *Lecture Notes in Computer Science*, pages 274–288. Springer, 2009.

[25] S. Donatelli, M. Ribaudo, and J. Hillston. A comparison of performance evaluation process algebra and generalized stochastic petri nets. In *In Proc. 6th International Workshop on Petri Nets and Performance Models*, pages 158–168. IEEE Computer Society Press, 1995.

[26] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Trans. Netw.*, 1(4):397–413, 1993.

[27] Adrian Francalanza and Matthew Hennessy. A theory of system behaviour in the presence of node and link failures. In *CONCUR*, pages 368–382, 2005.

[28] Adrian Francalanza and Matthew Hennessy. A theory of system behaviour in the presence of node and link failure. *Inf. Comput.*, 206(6):711–759, 2008.

[29] M. Gaur and R. Kant. A survey on process algebraic stochastic modelling of large distributed systems for its performance analysis. In *3rd International Conference on Eco-friendly Computing and Communication Systems (ICECCS), 2014*, pages 206–211, Dec 2014.

[30] Manish Gaur. Deciding efficiency prebisimulation. Master's thesis, Department of Computer Science and Engineering, Indian Institute of Technology Delhi, New Delhi, December 2001.

[31] Manish Gaur. A routing calculus for distributed computing. In Elena Troubitsyna, editor, *Proceedings of Doctoral Symposium held in conjunction with Formal Methods 2008*, volume 48, pages 23–32. Turku Centre for Computer Science General Publication, May 2008.

[32] Manish Gaur. *A Routing Calculus: Towards formalising the cost of computation in a distributed computer network*. PhD thesis, Informatics, University of Sussex, U.K., December 2008.

[33] Manish Gaur and S. Arun-Kumar. On efficiency preorder. In Sirjani Marjan Arbab Farhad, editor, *Pre Proceedings of 5th IPM International Conference on Fundamentals of Software Engineering(FSEN 2013)*, pages 84–95, Tehran, Iran, April 2013. Institute of Research in Fundamental Sciences, IPM.

[34] Manish Gaur and S. Arun-Kumar. On efficiency preorder(revised). In Sirjani Marjan Arbab Farhad, editor, *Post Proceedings of 5th IPM International Conference on Fundamentals of Software Engineering(FSEN 2013)*, pages 83–94. Lecture Notes in Computer Science (LNCS), Vol 8161, 2013.

[35] Manish Gaur and Matthew Hennessy. Counting the cost in the picalculus (extended abstract). *Electronic Notes in Theoretical Computer Science (ENTCS)*, 229:117–129, 2009.

[36] Simon J. Gay and Malcolm Hole. Subtyping for session types in the pi-calculus. *Acta Inf.*, 42(2-3):191–225, 2005.

[37] Timothy G. Griffin and João L. Sobrinho. Metarouting. In *SIGCOMM*, pages 1–12, 2005.

[38] Matthew Hennessy. *A distributed Pi-Calculus*. Cambridge University Press, 2007.

[39] Matthew Hennessy. Exploring probabilistic bisimulations, part i. *Formal Asp. Comput.*, 24(4-6):749–768, 2012.

[40] Matthew Hennessy and Julian Rathke. Typed behavioural equivalences for processes in the presence of subtyping. *Mathematical Structures in Computer Science*, 14(5):651–684, 2004.

[41] C. A. R. Hoare. Communicating sequential processes. *Commun. ACM*, 21(8), 1978.

[42] Kohei Honda and Mario Tokoro. On asynchronous communication semantics. In P Wegner M. Tokoro and O Nierstrasz, editors, *Proceedings of the ECOOP'91 Workshop on Object-Based Concurrent Computing*, volume 612 of LNCS 612. Springer-Verlag, 1992.

[43] Christian Huitema. *Routing in the Internet (2nd ed.)*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2000.

[44] Van Jacobson. Congestion avoidance and control. In *SIGCOMM*, pages 314–329, 1988.

[45] Paris C. Kanellakis and Scott A. Smolka. Ccs expressions, finite state processes, and three problems of equivalence. *Inf. Comput.*, 86(1):43–68, 1990.

[46] Rama Kant and Manish Gaur. Article: A stochastic extension of the routing calculi. *IJCA Proceedings on International Conference on Distributed Computing and Internet Technology*, ICDCIT 2015(1):13–17, January 2015. Full text available.

[47] Astrid Kiehn and S. Arun-Kumar. Amortised bisimulations. In *FORTE*, pages 320–334, 2005.

[48] James F. Kurose and Keith W. Ross. *Computer Networking: A top down approach featuring the Internet*. Addison Wesley, 2001.

[49] Cosimo Laneve and Luca Padovani. The *ust* preorder revisited. In *CONCUR*, pages 212–225, 2007.

[50] C. L. Liu and James W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *J. ACM*, 20(1):46–61, 1973.

[51] Ian Mackie Manish Gaur and Simon Gay. A routing calculus with flooding updates. *(accepted)11th International Conference on Distributed Computing and Internet Technology (ICDCIT-2015)Proceedings to appear in Lecture Notes in Computer Science.*, 2015.

[52] M. Ajmone Marsan. Stochastic petri nets: An elementary introduction. In *In Advances in Petri Nets*, pages 1–29. Springer, 1989.

[53] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[54] Robin Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer, 1980.

[55] Robin Milner. *Communicating and mobile systems: The π-Calculus*. Cambridge University Press, 1999.

[56] Nagle. On packet switches with infinite storage. *IEEE Transaction on Communications*, 35:435–438, 1987.

[57] Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Commun. ACM*, 21(12):993–999, 1978.

[58] Rocco De Nicola, Daniele Gorla, and Rosario Pugliese. Basic observables for a calculus for global computing. *Inf. Comput.*, 205(10):1491–1525, 2007.

[59] Gethin Norman and David Parker. Quantitative verification: Formal guarantees for timeliness, reliability and performance. Technical report, The London Mathematical Society and the Smith Institute, 2014.

[60] Fredrik Orava and Joachim Parrow. An algebraic verification of a mobile network. *Formal Asp. Comput.*, 4(6):497–543, 1992.

[61] Robert Paige and Robert Endre Tarjan. Three partition refinement algorithms. *SIAM J. Comput.*, 16(6):973–989, 1987.

[62] Craig Partridge, James Hughes, and Jonathan Stone. Performance of checksums and crcs over real data. In *SIGCOMM*, pages 68–76, 1995.

[63] Vamsi Paruchuri, Arjan Durresi, and Raj Jain. Optimized flooding protocol for ad hoc networks. *CoRR*, cs.NI/0311013, 2003.

[64] Benjamin C. Pierce and Davide Sangiorgi. Typing and subtyping for mobile processes. *Mathematical Structures in Computer Science*, 6(5):409–453, 1996.

[65] K. K. Ramakrishnan and Raj Jain. A binary feedback scheme for congestion avoidance in computer networks. *ACM Trans. Comput. Syst.*, 8(2):158–181, 1990.

[66] Davide Sangiorgi. *Introduction to Bisimulation and Coinduction*. Cambridge University Press, 2012.

[67] Davide Sangiorgi and Jan Rutten (eds). *Advanced Topics in Bisimulation and Coinduction*. Cambridge University Press, 2012.

[68] Davide Sangiorgi and Robin Milner. The problem of “weak bisimulation up to”. In *CONCUR*, pages 32–46, 1992.

[69] Davide Sangiorgi and David Walker. *The π-Calculus: A theory of Mobile Processes*. Cambridge University Press, 2001.

[70] Huzur Saran, Srinivasan Keshav, and Charles R. Kalmanek. A scheduling discipline and admission control policy for xunet 2. *Multimedia Syst.*, 2(3):118–128, 1994.

[71] Peter Sewell, Pawel T. Wojciechowski, and Benjamin C. Pierce. Location-independent communication for mobile agents: A two-level architecture. In *ICCL Workshop: Internet Programming Languages*, pages 1–31, 1998.

[72] Nachum Shacham and Paul E. McKenney. Packet recovery in high-speed networks using coding and buffer management. In *INFOCOM*, pages 124–131, 1990.

[73] W. Richard Stevens. *TCP/IP Illustrated : The Protocols*, volume 1. Addison-Wesley, 1994.

[74] Cisco systems Inc. Handbook of internetworking: Routing basics. Web site, 1992-2008. http://www.cisco.com/en/US/docs/internetworking/technology/handbook/ Routing-Basics.html.

[75] Andrew S. Tanenbaum. *Computer Networks*. Pearson Education, Inc., Upper Saddle River, New Jersey, fourth edition, 2003.

[76] Jonathan S. Turner. New directions in communications. *IEEE Communications Magazine*, 520 GSCs 3/06., 1986.

[77] Robbert van Renesse, Hans van Staveren, and Andrew S. Tanenbaum. Performance of the world's fastest distributed operating system. *Operating Systems Review*, 22(4):25–34, 1988.

[78] Oskar Wibling, Joachim Parrow, and Arnold Neville Pears. Automatized verification of ad hoc routing protocols. In *FORTE*, pages 343–358, 2004.

[79] John Wroclawski, Christophe Diot, Christian Huitema, and Edward W. Knightly. Qos research in a complicated world. In *INFOCOM*, 2002.