Why Confluence is More Powerful than Ample Sets in Probabilistic and Non-Probabilistic Branching Time*

Henri Hansen
Department of Software Systems
Tampere University of Technology
PO Box 553, FI-33101 Tampere, Finland
henri.hansen@tut.fi

Mark Timmer
Formal Methods and Tools
Faculty of EEMCS
University of Twente, The Netherlands
timmer@cs.utwente.nl

Confluence reduction and partial order reduction by means of ample sets are two different techniques for state space reduction in both traditional and probabilistic model checking. This presentation provides an extensive comparison between these two methods, answering the long-standing question of how they relate. We show that, while both preserve branching time properties, confluence reduction is strictly more powerful than partial order reduction: every reduction that can be obtained with partial order reduction can also be obtained with confluence reduction, but the converse is not true.

A core problem in the comparison is that confluence reduction was defined in an action-based setting, whereas partial order reduction was defined in a state-based setting. We therefore redefine confluence reduction in the state-based setting of Markov decision processes, and discuss a nontrivial proof of its correctness. Additionally, we pinpoint precisely in what way confluence reduction is more general, and provide a restricted variant of confluence and relaxed variant of partial order reduction that exactly coincide. The results we present also hold for non-probabilistic models, as they can just as well be applied in a context where all transitions are non-probabilistic.

To show the practical applicability of our results, we adapt a state space generation technique based on representative states, already known in combination with confluence reduction, so that it can also be applied with partial order reduction.

1 Introduction

Probabilistic model checking has proved to be an effective way for improving the quality of communication protocols and encryption techniques, but also for studying biological systems or measuring the performance of networks. The omnipresent state space explosion poses a serious threat to the efficiency of model checking and similar methods; therefore, several reduction techniques have been introduced to deal with large systems.

Recently, two powerful reduction techniques from non-probabilistic model checking were generalised to the probabilistic setting: *partial order reduction* [8, 11, 16] and *confluence reduction* [4, 3]. Both use a notion of independence between transitions of a system, either explicitly or implicitly, and try to reduce the state space by eliminating redundant paths through the system (and therefore often also states). In the non-probabilistic setting, partial order reduction techniques have been defined for a large range of property classes, most notably variants that preserve $\mathsf{LTL}_{\setminus X}$ and $\mathsf{CTL}^*_{\setminus X}$ [17, 12]. Most work on confluence reduction has been designed such that the reduced system is branching bisimilar to the original system; thus, these techniques preserve virtually all branching properties (in particular, $\mathsf{CTL}^*_{\setminus X}$). There is not as much work on weaker variants of confluence, though in [10] a variant is explored that

*This research has been partially funded by NWO under grants 612.063.817 (SYRUP) and Dn 63-257 (ROCKS), and by the Finnish Foundation for Technology Promotion.

Submitted to:
QAPL 2012

makes no distinction between visible and invisible actions and does not require acyclicity. This variant preserves deadlocks much in the same way as weaker versions of ample and stubborn sets [17].

Partial order reduction, in the form of *ample sets*, was the first of these methods to be applied in the probabilistic setting. In [2] and [6], the concept was lifted from labelled transition systems to Markov decision processes (MDPs), providing reductions that preserve quantitative $\mathsf{LTL}_{\setminus X}$. These techniques were refined in [1] to also preserve probabilistic $\mathsf{CTL}^{*}_{\setminus X}$, a branching logic. Later, a revision of partial order reduction for distributed schedulers was introduced and implemented in PRISM [7]. Of the other partial order reduction techniques, the so-called weak stubborn set method was also defined for a class of safety properties of MDPs under fairness constraints in [9].

Recently, confluence reduction was lifted to the probabilistic realm as well. In [15, 14] a probabilistic variant was introduced that, just like the ample set reduction of [1], preserves branching properties. It was defined as a reduction technique for action-based probabilistic automata [13], but as we will show in this presentation, it can also be used in the context of MDPs.

Ample sets and confluent transitions are defined and detected quite differently: ample sets are defined by first giving an independence relation for the action labels, whereas confluence is a property of a set of (invisible) transitions in the final state space. Even so, the underlying ideas are similar on the intuitive level. Therefore, an obvious question is: to what extent do they indeed coincide? This presentation addresses that question by comparing the notion of probabilistic ample sets from [1] to the notion of strongly probabilistically confluent sets from [15].

2 Results

We will first redefine confluence for MDPs.[1] This task is nontrivial, because confluence is originally defined in a purely action-based formalism. Then, we show that confluence reduction is strictly more powerful than ample set reduction, by proving that every nontrivial ample set can be mimicked by a confluent set, while also providing examples where confluent transitions do not qualify as ample sets. In such cases, confluence reduction is able to reduce more than ample set reduction. Additionally, we pinpoint precisely in what way confluence is more general than ample sets, and restrict the definition of confluence as well as relax the definition of ample sets, to make them coincide. Interestingly, this relaxed definition of ample sets can be seen as a probabilistic generalisation of stubborn sets.

While revealing exactly where the extra reduction with confluence comes from, the results we present support the idea that confluence reduction is a well-suited alternative to the thus far more often used partial order reduction method. In particular, this is a major consideration in contexts where detection of confluence using heuristics that make use of these more relaxed conditions is possible, or where the conditions of confluence are just easier to check than their partial order reduction counterparts. This seems to be the case for statistical model checking and when working with process-algebraic modelling languages, respectively. Alternatively, the relaxed definition of ample sets might be used in settings where the notion of partial order reduction is more natural. In addition to providing these practical opportunities, our precise comparison of confluence and partial order reduction fills a significant gap in the theoretical understanding of the two notions.

The theory is presented in such a way, that the results hold for non-probabilistic automata as well, as they form a special case of the theory where all probability distributions are deterministic. Hence, as a

[1] This abstract is based on a paper with the same name, currently in submission at a journal, that can be downloaded from `http://wwwhome.cs.utwente.nl/~timmer/research.php`. Due to the space limitations of this abstract we refer to the paper for all the technical details. During the talk, the technicalities will be made intuitively clear by means of many pictures.

side effect we also answered the long-standing question of how the non-probabilistic variants of partial order reduction and confluence reduction relate.

Our findings imply that results and techniques applicable to confluence can be used in conjunction with ample sets. As an example of such a technique, we show how a state space generation technique based on *representative states*, already known in the context of confluence reduction [4], can also be applied with partial order reduction. This is a very general technique for replacing a class of states by a single representative, and a quite similar method has also been used in conjunction with the so-called essential state abstraction in [5]. The technique makes explicit checking of the cycle condition of ample sets redundant, in addition to further reducing the number of states and transitions. The latter is important, especially if the MDP is to be subjected to further analysis.

3 Conclusions and discussion

We redefined probabilistic confluence reduction to an MDP-based setting, enabling a comparison to probabilistic partial order reduction based on ample sets. We proved that every nontrivial ample set can be mimicked by a confluent set, and that in some cases reductions are possible using confluence but not using ample sets. Therefore, at least in theory confluence reduction is able to reduce more than the ample set method. We also showed the exact way in which confluence has to be strengthened and ample sets have to be relaxed for the two notions to coincide. These results hold for the non-probabilistic variants of the two reduction techniques as well. Our observation that probabilistic ample set reduction can be mimicked by probabilistic confluence reduction has additional implications, among which the above-mentioned application of a representation map for partial order reduction.

As both ample sets and confluence are detected symbolically on the language level, the quality of the heuristics applied there will decide which notion works best in practice. The results of this presentation already strengthen our theoretical understanding of the two methods, and this is independent of the heuristics that are applied. Also, no matter how such heuristics might be improved, the results of this presentation will remain valid. Even though a case study on probabilistic confluence reduction in [15] seemed to outperform similar reductions based on ample sets, future work could focus more on the relative merits of the two notions in practice and potentially on the improvement of the syntactical heuristics, if some "best of both worlds" approach is found.

A natural question is, whether there are similar results that could be proven for weaker semantics, like reductions that preserve $\mathsf{LTL}_{\setminus X}$. For most part, the answer is obvious: confluence reduction preserves branching time properties, so it also preserves $\mathsf{LTL}_{\setminus X}$. However, since confluence is designed to preserve branching properties, it has the inherent restriction that confluent transitions must lead to bisimilar states. This means that we must be able to take single confluent transitions, for if we couldn't, we would lose some state that is not bisimilar to the current state. Ample set, and similar methods, do not need such a restriction when dealing with weaker semantics.

One class of open and interesting questions remains, however. When aiming to make confluence reduction and partial order reduction coincide, we worked mostly by *restricting* confluence. It is sensible to ask, if we could have proven the theorem by relaxing the ample set conditions more and restricting the confluence conditions less, while maintaining a practical method that can make use of the extra reduction. How would the less restrictive conditions of confluence be used in conjunction with ample sets or other partial order reduction methods? Could similar conditions be used when partial order reduction preserves weaker properties, like $\mathsf{LTL}_{\setminus X}$? Future work might focus on answering these questions.

References

[1] C. Baier, P. R. D'Argenio & M. Größer (2006): *Partial Order Reduction for Probabilistic Branching Time*. In: *Proceedings of the Third Workshop on Quantitative Aspects of Programming Languages (QAPL)*, *ENTCS* 153(2), Elsevier, pp. 97–116.

[2] C. Baier, M. Größer & F. Ciesinski (2004): *Partial Order Reduction for Probabilistic Systems*. In: *Proceedings of the 1st International Conference on Quantitative Evaluation of Systems (QEST)*, IEEE Computer Society, pp. 230–239.

[3] S. C. C. Blom (2001): *Partial τ-confluence for efficient state space generation*. Technical Report SEN-R0123, CWI.

[4] S. C. C. Blom & J. C. van de Pol (2002): *State Space Reduction by Proving Confluence*. In: *Proceedings of the 14th International Conference on Computer Aided Verification (CAV)*, *Lecture Notes in Computer Science* 2404, Springer, pp. 596–609.

[5] P. R. D'Argenio, B. Jeannet, H. E. Jensen & K. G. Larsen (2002): *Reduction and Refinement Strategies for Probabilistic Analysis*. In: *Proceedings of the 2nd Joint International Workshop on Process Algebra and Probabilistic Methods, Performance Modeling and Verification (PAPM-PROBMIV)*, *Lecture Notes in Computer Science* 2399, Springer, pp. 57–76.

[6] P. R. D'Argenio & P. Niebert (2004): *Partial Order Reduction on Concurrent Probabilistic Programs*. In: *Proceedings of the 1st International Conference on Quantitative Evaluation of Systems (QEST)*, IEEE Computer Society, pp. 240–249.

[7] S. Giro, P. R. D'Argenio & L. María Ferrer Fioriti (2009): *Partial Order Reduction for Probabilistic Systems: A Revision for Distributed Schedulers*. In: *Proceedings of the 20th International Conference on Concurrency Theory (CONCUR)*, *Lecture Notes in Computer Science* 5710, Springer, pp. 338–353.

[8] P. Godefroid (1996): *Partial-order Methods for the Verification of Concurrent Systems: an Approach to the State-explosion Problem*. *Lecture Notes in Computer Science* 1032, Springer.

[9] H. Hansen, M. Kwiatkowska & H. Qu (2011): *Partial order reduction for model checking Markov decision processes under unconditional fairness*. In: *Proceedings of the 8th International Conference on Quantitative Evaluation of SysTems (QEST)*, IEEE Computer Society, pp. 203–212.

[10] F. Lang & R. Mateescu (2009): *Partial Order Reductions Using Compositional Confluence Detection*. In: *Proceedings of the 2nd World Congress on Formal Methods (FM)*, *Lecture Notes in Computer Science* 5850, Springer, pp. 157–172.

[11] D. Peled (1993): *All from One, One for All: on Model Checking Using Representatives*. In: *Proceedings of the 5th International Conference on Computer Aided Verification (CAV)*, *Lecture Notes in Computer Science* 697, Springer, pp. 409–423.

[12] D. Peled (1998): *Ten Years of Partial Order Reduction*. In: *Proceedings of the 10th International Conference on Computer Aided Verification (CAV)*, *Lecture Notes in Computer Science* 1427, Springer, pp. 17–28.

[13] R. Segala (1995): *Modeling and Verification of Randomized Distributed Real-Time Systems*. Ph.D. thesis, Massachusetts Institute of Technology.

[14] M. Timmer, M. I. A. Stoelinga & J. C. van de Pol (2010): *Confluence Reduction for Probabilistic Systems (extended version)*. Technical Report 1011.2314, ArXiv e-prints.

[15] M. Timmer, M. I. A. Stoelinga & J. C. van de Pol (2011): *Confluence reduction for Probabilistic Systems*. In: *Proceedings of the 17th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, *Lecture Notes in Computer Science* 6605, Springer, pp. 311–325.

[16] A. Valmari (1989): *Stubborn sets for reduced state space generation*. In: *Proceedings of the 10th International Conference on Application and Theory of Petri Nets*, *Lecture Notes in Computer Science* 483, Springer, pp. 491–515.

[17] Antti Valmari (1996): *Stubborn Set Methods for Process Algebras*. In: *Proceedings of the DIMACS workshop on Partial order methods in verification (POMIV)*, AMS Press, pp. 213–231.