Partial Order Reduction in Presence of Rendez-vous Communications with Unless Constructs and Weak Fairness

Dragan Bošnački *

Dept. of Computing Science, Eindhoven University of Technology
PO Box 513, 5600 MB Eindhoven, The Netherlands
fax: +31 40 246 3992, e-mail: dragan@win.tue.nl

Abstract. If synchronizing (rendez-vous) communications are used in the Promela models, the unless construct and the weak fairness algorithm are not compatible with the partial order reduction algorithm used in Spin’s verifier. After identifying the wrong partial order reduction pattern that causes the incompatibility, we give solutions for these two problems. To this end we propose corrections in the identification of the safe statements for partial order reduction and as an alternative, we discuss corrections of the partial order reduction algorithm.

1 Introduction

The issue of fairness is inherent and important one in the study of concurrency and nondeterminism, in particular in the area of the verification of concurrent systems. Since fairness is used as generic notion there is a broad taxonomy of fairness concepts. In this paper we confine our attention to the notion of weak fairness on the level of processes which is implemented in the Spin verifier. This means that we require that for every execution sequence of the concurrent program which is a composition of several processes, if some process becomes continuously enabled at some point of time (i.e. can always execute some of its statements), then at least one statement from that process will eventually be executed. This kind of fairness is most often associated with mutual exclusion algorithms, busy waiting, simple queue-implementations of scheduling, resource allocation. Weak fairness will guarantee the correctness of statements like eventually entering the critical region for every process which is continuously trying to do that (in the mutual exclusions) or eventually leaving the waiting queue for each process that has entered it (in the scheduling) [7].

Partial order reduction is one of the main techniques that are used to alleviate the problem of state space explosion in the verification of concurrent systems [16][13][11][14] and it is indeed one of Spin’s main strengths. The idea is, instead of

* On leave from the Institute of Informatics, Faculty of Natural Sciences and Mathematics, University “Sts. Cyril and Methodius”, Skopje, Macedonia. Supported by EC ESPRIT LTR Project No. 23498 (VIRES).
exploring all the execution sequences of a given program, to group them in equivalence classes which are interleaving of independent program statements. Then only representatives for each equivalence class are considered. In practice this is realized such that from each state only a subset of the executable statements are taken.

Combining the algorithms for model-checking under weak fairness with partial order reduction is a prerequisite for the verification of many interesting properties to be feasible in practice. However, recently it was discovered that the two algorithms are not compatible when rendez-vous communications occur in the Promela models. As a result, in the present implementation of Spin the combination of weak fairness with partial order reduction when rendez-vous are used in the models is not allowed.

Another problem with Spin’s partial order reduction in presence of rendez-vous occurs when the unless construct is used in the Promela models. The combination of this three Spin’s features is also currently forbidden.

Interestingly, it turns out that both incompatibilities are caused exactly by the same pattern of wrong partial order reduction. After pointing out this incorrect reduction pattern we propose solutions for the problems with fairness and unless. For both cases we discuss two kinds of solutions, classified according to the two different phases of the verification in which they are implemented. The first kind corrects the identifications of so called safe statements for the partial order reduction algorithm. The marking of the statements as safe is done during the compilation of the Promela model, so we call this solutions static. The second kind are the dynamic solutions which are applied during the exploration of the state space and are in fact corrections of the partial order reduction algorithm.

In the next section we give the necessary preliminaries for the rest of the paper. Section 3 is devoted to partial order reduction and the concrete algorithm that is used in Spin. In Section 4 we discuss the problem with the unless construct and give some solutions to overcome it. Section 5 deals with the Spin’s weak fairness algorithm. After location of the problem and the comparison with the unless case, we again propose both static and dynamic solutions. The last section is a standard summary with some considerations about the future work.

2 Preliminaries

In this section following [11] and [5] we give semantics of the Promela programs (models) and their verification in terms of finite labeled transition systems.

We represent the programs as collections of processes. The semantics of the process $P_i$ can be represented as a labeled transition system (LTS). An LTS is a quadruple $(S_i, s_{0i}, \tau_i, L_i)$, where $S_i$ is a finite set of states, $s_{0i}$ is a distinguished initial state, $L_i$ is a set of program statements (labels), and $\tau_i : S_i \times L_i \rightarrow 2^{S_i}$ is a nondeterministic transition function. The transition function induces a set $T_i \subseteq S_i \times S_i$ of transitions. Every transition in $T_i$ is the result of an execution of a statement from the process, i.e., $(s_i, s'_i) \in T_i$ iff there exists a statement $a$