

Dear editor,

we have addressed the concerns of the reviewers in this revised version of the paper. We have also adapted our paper to the stylistic guidelines and linguistic rules of the journal. Next, we provide answers (in blue text) to the comments/remarks pointed out by the reviewers (in black text).

Please, notice that we took into account **all** the comments received by the reviewers. In particular the ones related to operational semantics of your language.

Reviewer 1

The authors addressed the concerns raised in my review for the previous version of this draft (Reviewer 1).

Question 1:

The only remark is that I don't agree with the view the authors have of 'mandatory features': in other approaches to modeling SPL, mandatory features have to be always present in any generated product, no matter what dynamics are executed. Instead, the answer given by the authors in the response letter implies that mandatory features are not necessarily present in a product. This is not clear in the text.

Given that this approach is not standard, it should be highlighted and discussed in the text.

For this reason, I suggest accepting this version paper.

Thanks for this comment. In order to clarify this aspect, the next paragraph have been included in the new version of the paper (see Section 3.1, third paragraph).

“In this research article, like in the previous definition of SPLA [8, 17], we define and express formally that even if a feature is represented with a mandatory relationship in the feature model, it might not be computed in the final set or trace of valid products. This is because of the cross tree constraints presented in the formal definition of SPLA, more in specific, the [excl2] and [excl3] rules. When these rules are computed, the affected features will be marked for hiding. This is carried out by rules [hid1] and [hid2] from Figure 6. So forth, the features disappear in the valid products traces after computing the feature model.”

Reviewer 2

Question 1:

Thanks for including the semantics of t_1 into your set of examples (Figure 3). I wonder why the "probabilities" for B and C in the second step are not $1/4$: for instance take a look at $[\text{tick} \wedge (B;\text{tick} \wedge C;\text{tick})]$, then $(B;\text{tick} \wedge C;\text{tick}) \rightarrow 1/2 (\text{tick} \wedge C;\text{tick})$ by rule [con1]. Furthermore, $[\text{tick}]$ yields $(\text{tick} \rightarrow 1 \text{ nil})$ and combined with the above transition in [con5] we obtain $[\text{tick} \wedge (B;\text{tick} \wedge C;\text{tick})] \rightarrow 1/4 (\text{tick} \wedge C;\text{tick})$ as $(1 \cdot 1/2)/2 = 1/4$. This issue I have already indicated in the last review and the authors did not provide any satisfactory answer (why the probabilities would not sum up to 1?).

We are very sorry. The example of Figure 3 was incorrectly drawn: although our semantics was indeed correct, we made a mistake in drawing the example (we assumed a simplified representation of states, as we explain below).

Indeed, as the reviewer says, according to our semantics, we have:

$\text{tick} \wedge (B;\text{tick} \wedge C;\text{tick}) \rightarrow B, 1/4 \rightarrow \text{tick} \wedge C;\text{tick}$ by rules [CON5] + [CON1]

However, we have also:

$\text{tick} \wedge (B;\text{tick} \wedge C;\text{tick}) \rightarrow B, 1/4 \rightarrow \text{tick} \wedge (\text{tick} \wedge C;\text{tick})$ by rules [CON2] + [CON1]

So, in total, we have a $\rightarrow B, 1/2 \rightarrow$ transition towards the two states above (in the example of Figure 3 we made the mistake of not distinguishing them, again we are sorry for that).

A completely symmetrical argument holds for $\rightarrow C, 1/2 \rightarrow$

Unfortunately, a similar mistake was made for subsequent states of Figure 3 (although probabilities were collectively correct, the states were not correctly distinguished). In the new version of the paper we present the precise semantics of the t_1 example (e.g. correctly distinguishing the states above): being it very wide, the presentation of the example semantics has been split into 2 figures, current Figure 3 and Figure 4. Thanks a lot for pointing out our mistake.

For me it is still not clear what kind of "associativity" is considered. In the standard setting, associativity would state that t_1 and t_2 have the same operational semantics or at least a bisimilar one (which is not the case as choosing A in t_1 has probability $1/2$, while in t_2 it has probability $1/4$). From the first statement in the answer of the authors I probably could deduce that associativity might only be considered with respect to probabilities of sets of features. But in this case I miss a justification why the proposed semantics is chosen to be operational as the operational behaviors are not used at all and no aspect of time plays a role in the considerations. In this spirit, I also miss an explanation why the authors are

departing from standard definitions in the field of probabilistic process algebras. Please indicate which SOS rules correspond to existing definitions and when/for which reason one introduces non-standard rules (such as the conjunctive rules).

The reviewer is right: associativity holds with respect to probabilities of sets of features (which correspond to sets of trace labels in the labelled transition system originated by the operational semantics). Formally, commutativity and associativity of \wedge is a consequence of the denotational semantic definition of the \wedge operator and Theorem 2, which relates equivalence in terms of (probability of) sets of features and denotational semantics.

Concerning Structural Operational Semantics, we just use it to define the labelled transition system (LTS) originated from a process algebraic term. As in the process algebra literature, this is not necessarily related to the kind of equivalence that is considered on the obtained LTS: notions of equivalence, besides bisimulation, considered in the literature (both in a probabilistic or non-probabilistic setting) are e.g. trace equivalence, may/must/fair/probabilistic testing preorder/equivalence, simulation (half-bisimulation) preorder/equivalence... In this paper we do not consider bisimulation, but a kind of “unordered” trace equivalence and we show it to correspond to a denotationally defined semantics. As in similar semantics in the literature, the “operational” nature of the semantics is used as a way to “compute” the possible sets and their associated probability. Our notion of equivalence (which in the previous version of the paper was, from a formal viewpoint, implicitly expressed through the relationship with the denotational semantics) has now been made explicit by introducing an equivalence relation (see the last line of Definition 3).

The fact, observed by the reviewer, that SOS rules differ from existing probabilistic process algebra in the literature reflect the originality of our approach: the reason is the different kind of equivalence that we consider.

Question 2:

The statement in the answer of 1) that probabilities at the operators are not "critical" and "we could have chosen any other distribution factor" made me even more wonder about the meaning of the probabilities. I would like to strengthen the need of an explanation for 2), as I do not see where probabilities come from and what is their meaning. As indicated, the natural way would be to present a probabilistic feature model (e.g., via FODA/feature diagrams) that then is translated to SPLA^P terms. Note that this approach would not cover the still missing explanation of the meaning of probabilities in the operational semantics.

1) When, in our previous answer, we indicated 'that the probabilities are not critical and that we could have chosen any other distribution factor' we referred to the conjunction operator. This operator does not have probabilities. It does not need them because, intuitively, any product derived from $P \wedge Q$ is the union of a product of P and a product of Q . Technically, however, the operational semantics uses a distribution factor between the two branches in order to compute product probabilities. Anyway, as long as such a factor is a number strictly between 0 and 1, its actual value is not significant. We could have chosen $1/3$, $2/3$ instead $1/2$, $1/2$ for the left and right terms of the \wedge operator: the probabilities obtained for the

products (unordered trace labels) would have been the same. If we consider the term $\text{tick} \wedge a;\text{tick}$ (and probabilities $1/3, 2/3$), we obtain

$\text{tick} \wedge a;\text{tick} \xrightarrow{a, 2/3} \text{tick} \xrightarrow{\text{tick}, 1} \text{tick}$
 $\text{tick} \wedge a;\text{tick} \xrightarrow{a, 1/3} \text{tick} \wedge \text{tick} \xrightarrow{\text{tick}, 1} \text{tick}$

So we obtain the product “a” with probability 1.

2) The probabilities in a FODA diagram would be in the choose-1 operator. Any element is chosen with a probability. It is true that our choice operator is binary, but adjusting properly the probabilities we can represent n-ary choices. We have mentioned this in the paper in Section 3.1, pages 9 and 10.

Question 3:

Thank you for explicitly devoting a section to related work. For estimating the contribution of this paper, I still miss a comparison to the approaches mentioned, especially to the ones that use operational semantics. What is the difference of your operational semantics to the ones in [25] (considering the Markov chain fragment) and [32]? The authors might furthermore explain the new sentence "state that state it is possible to describe a formal framework that translates the current feature models to probabilistic methods." as I could not grasp its meaning. Which "probabilistic methods" are meant and how in principle models can be translated to methods?

Thanks for this comment. We have re-written the paragraph that presents this comparison (see Section 2, third paragraph). Basically, the difference is that they use probabilities to model the behavior of products whereas we use probabilities to quantify how much significant is a product, or a feature, within the product line.

Question 4:

Still Figure 7 makes me wonder whether the analysis of the results were faithfully evaluated. As I already indicated in the last review, timings around 20ms are not realistic to analyse. The authors could scale their experiments, e.g., by considering feature numbers greater than, e.g., 10k as they did, e.g., in Figure 11 and 12? This could eventually mitigate the side effects of other processes etc. and could enable a trustworthy statement about the results.

Thanks for this comment. In the new version of the paper we have repeated the experiment presented in Section 6.1, which now uses a model containing 3000 features. Hence, the new executions require a higher computational time to process the models, which are shown in the statistical analysis presented in Table 1.

Small issues:

The authors say that they do not consider the operators to be n-ary anymore, but they are still highlighted to be n-ary directly after Definition 1.

The wrong references for n-ary operators have been removed from the new version of the paper.

The citation of PRISM is not appropriate, as it refers to its benchmark suite and not its implementation. Please use the reference indicated on the homepage <http://www.prismmodelchecker.org>

This citation has been fixed.

Reviewer 3

I want to thank the authors for addressing my comments. The evaluation section has been rewritten and now presents the empirical evidence of the applicability of the approach on synthetic feature models in a more explicit way.

Question 1:

However, the comparisons with previous work in Section 6.3 is not sound. As far as I can tell (since the feature models are not provided in the replication package), the set of feature models used in [8] is different from the set of feature models used in the current evaluation. The number of products for the different feature models presented in Figure 16 varies a lot, and the number of features is different between the two evaluations.

I suggest toning down a bit the claims about these comparisons. In particular, the claims made in the following paragraph: "For those specific implementations and simulations we can conclude that the denotational semantics of the probabilistic extension implementation improves dramatically the performance in comparison with the denotational semantics implementations presented in previous works [8, 17]." Based on the current data from the evaluations, the paper cannot conclude anything.

The paper can only claim that empirical evidence suggests that the performances of the denotational semantics of the probabilistic extension implementation are better than the denotational semantics implementations presented in previous works [8, 17].

The reviewer is right. It is difficult to compare the results of this paper with the older ones. In those papers we computed the whole product line while in this one we compute the probabilities of single features. We have decided to remove this comparison.

Question 2:

The paper has still no threats to validity section. Since the empirical evaluation has been done using a limited number of synthetic feature models, I believe that this section should be there and discuss the limitations of the approach and its empirical evaluation.

The new version of the paper includes a new section presenting the threats to validity (see Section 7)

Specific comments:

- p.18, l.17, of A for b -> of A for b
 - p.37, l.16, l.21, l.26, I suggest giving the 3 times values in seconds to ease the comparison
- p.38, l.3, l.18, to ask -> to answer

These small issues have been fixed.