Cover Letter for

"Efficient Anytime Computation and Execution of Decoupled Robustness Envelopes for Temporal Plans"

The attached reviews from IJCAI 2019 were mainly positive. The main concern with the paper raised by all reviewers was that while it was claimed that the approach was able to handle non-temporal uncertainty, this was not fully demonstrated through experimental evaluation. This is a critical point, as one novelty of our approach with respect to existing algorithms is the handling of non-temporal uncertainty.

To address this point, and others, we have significantly expanded the experimental evaluation section, and clarified various points throughout the paper, as detailed below.

Experimental evaluation:

- We consider continuous parameters that can represent durations of actions, but also temporal distances between action instances, consumption rates, and any numeric assignments. Both the algorithm and the executor fully supports these features as long as the quantities modeled by the parameters are observable at run-time. The new version of the paper presents experiments demonstrating our flexible execution approach on non-temporal parameters.
- The new evaluation section includes three parts: (A) an evaluation of the scaling of DRE, (B) a comparison of DRE against replanning to illustrate the online performance of DRE, (C) a comparison of DRE against replanning with non-temporal parameters.

Clarifications:

- We note in the paper that some existing PDDL planners are unable to generate flexible STNs either because of an implementation limitation or because the technique does not allow it (e.g. SAT-based planners). Our approach is able to generate DREs from these planners as well as general PDDL planners, and work in concert with existing algorithms for the execution of STNs.
- The paper clarifies that the results are from executions in which plan and DRE generation are both performed online - see evaluation B and C, in which the times presented include everything and there is no offline phase. It is now explicit that IRR scales well enough for use in online execution.
- We highlight that IRR is in fact an optimization procedure to maximize the size of the DRE, but many equally-optimal DREs are possible. This is why we do support a weighting of the parameters. The parameters used in the benchmark problems are reported in the paper.

METAREVIEW

The paper has been subject to a careful debate among reviewers. All of them acknowledge the comments in the authors rebuttal. Nevertheless, some sense of weakness still remains in the experimental part that prevents the current paper to achieve an IJCAI-standard for publication We collectively encourage the author(s) to continue their work on the topic because we acknowledge its relevance.

REVIEW 1

The author(s) build on top of a recently published paper (Cashmore et al AAAI 2019) and it is good to have such a level of reactivity on the results of our community.

Indeed while the AAAI paper is pretty good in terms of results here I maintain some doubts.

The issue is to address robustness at execution time. The topic is well chosen, the coverage of the literature is OK. The technical ability of the author(s) is significant (in term of writing). This reviewer remains with the sense that still something is missing in term of the novelty of results with respect to the conference target (IJCAI).

In particular I am not surprised that an approximation of a computation is better of the complete calculus (sounds rather obvious).

Furthermore, in my view the paper sounds a bit of "incremental work" with the experiments that are completely internal to this work and the previously mentioned AAAI paper.

The weaker part at the moment are the results in Table 1 -- cannot say that what they are comparing with really demonstrate that the proposal is such a novel achievement. I am aware that the authors are in the right direction with their work but I really think that something is missing for a crisp-IJCAI-style-result.

REVIEW 2

The paper aims at leveraging robust envelopes (REs) (presented by Cashmore et al 2019) to deploy more robust temporal plan-based controllers in terms of a reduced number of re-planning occurrences. Given the computational costs for synthesising RE, authors propose to consider approximation of REs, called Decoupled REs (DREs). A planning and execution process in ROSPlan is proposed to leverage DREs and enforcing more robust execution with an algorithm (IRR) for generating them. The IRR algorithm is compared against the algorithm for processing the complete RE. An assessment of the DREEX executor is compared with a set of "simple" executors set with different tolerance rates on temporal action durations.

The paper addresses a key problem in planning and execution, i.e., the synthesis of robust executors and leverages an interesting result coming from the possibility of considering RE to robustify temporal plans (considering PDDL 2.1).

Nevertheless, I have some major concerns with the paper.

The paper claims two main contributions: a novel and scalable algorithm for computing DREs as sound approximations of REs; to demonstrate the practical usage of DREs in a robotic executor, extending the classical flow from planning to execution to re-planning.

- As for the first result, the proposed algorithm seems novel and correct. But I have some concerns about its efficiency and scalability.

First, it seems to me a straightforward result that computing DREs is more efficient than computing REs.

Being an approximation of REs, clearly, DREs should be easier to be calculated.

Second, IRR is clearly able to synthesize far more DREs then the original algorithm but, looking at the charts in Fig. 4 - based on a logarithmic scale, its scalability seems to follow an exponential trend that may prevent its practical deployment.

And this opens a question: how much can I gain from using DREs and reducing replanning rather than having more "naive" executors with more replanning? At the end, the real objective is to have a good trade-off between the two costs rather than just minimizing replanning steps.

- As for the second result, comparing only executors seems to me not fully appropriate. Indeed, a more comprehensive evaluation should include all the involved "costs".

Namely, a comprehensive comparison should include and compare the costs for plan synthesis, for generating DREs and for execution evaluating the differences between synthesis costs and execution costs.

In fact, I guess that setting up an executor with DREs is not a "cheap" task and that costs should be included in the assessment of the whole process.

For instance, BLEX(60) seems to show similar (in one case even better) performance with respect to DREEX, and its "setting cost" is clearly neglectable with respect to DREEX. So in that case, it would be clear that computing DREs is (globally) not efficient at all.

As a final remark on comparison, rather than comparing DREEX with the naive BLEX executors, it would be more interesting to consider DREEX vs different flexible control approaches (e.g., with EUROPA or PLATINUM, etc) comparing also different robust control strategies (with STN or flexible temporal timelines) and considering all the costs for generating plan-based robust controllers.

Finally, in my opinion algorithm 2 can be replaced by "classical" and well established dispatching algorithms (see e.g., Tsamardinos et al, Hunsberger et al, etc).

In general, the paper seems to me not suitable for being accepted as affected by several major issues.

REVIEW 3

This paper proposes an algorithm to compute the Robustness Envelopes (RE) for temporal plans. A RE is a multidimensional convex polytope associated to a determined temporal problem (and to its solution), where each point contained in the polytope represents a parameter value selection that guarantees termination of the execution of the plan despite the possible occurrence of exogenous events. This work presents an algorithm to compute the REs in an appoximate fashion, i.e., where the value of every dimension can be freely (and safely) selected, independently from the values selected in any other dimension. The newly produced REs are hence called Decoupled Robustness Envelopes (RDE).

The paper is rather easy to read; it is well structured, and the english language is clear (with only a few typos, see below).

The paper is significantly based on results in the past literature; it actually explores an idea that has been introduced in a previous publication, duly cited in the paper, but not finalized. In this paper, the authors: (i) propose the algorithm (called Incremental Rectangular Robustufucation - IRR) that realizes such idea (i.e., computes the RDEs), (ii) describe the Planning & Execution framework within which the IRR algorithm is used, (iii) compare the IRR algorithm against the existing complete version of the RE-computing algorithm (called CCMMZ), and (iv) present the results of such comparison.

The main novel contribution of the paper is the IRR algorithm, whose pseudocode is thoroughly described. The algorithm in itself is rather simple, as it is basically a generate-and-test procedure: for each parameter representing the flexibility of the plan, the algorithm iteratively attempts to widen the dimension associated to such parameter of a given "delta", until the widening is permitted (i.e., the approximated polytope does not exceed the real polytope's size). At the end of the process, an approximation of the "complete" RE is obtained, and given that such approximation is rectangular, any value independently selected for each dimension is guaranteed to constitute the coordinates of a point inside the polytope (i.e., it represents a feasible combination of values for all the parameters).

One interesting aspect that is missing in the analysis, and also in view of the forthcoming experimental analysis, is the computation of the algorithm's complexity.

About the experimental section: how many parameters are involved in the DRE approximation process for the selected benchamrk? That is, how many dimensions is IRR called to work on? Are *all* the activity durations parametrized, or only a part of them? What is the size of the planning solution instances?

Looking at the graphs (Figures 3 and 4), while IRR is certainly faster than the complete version of the DE computation procedure (CCMMZ), it can be observed that both procedures have a hard time against one of the domains used for the empirical tests, which raises some doubts on the IRR's true scalability properties, and applicability potential even on medium-scale real-world problems (i.e., problems and domains that go beyond the academic versions).

What is the reason why a "much simplified" Robot Delivery domain had to be used?

Moreover, to the best of the reviewer's understanding, depending on: (i) the number of constraints and number of parameters involved in all the constraints (that is, depending on the shape of the exact RE), (ii) the "location" of the initial point the widening starts from, (iii) the first parameter selected for expansion, (iv) the first direction selected for expansion of the selected parameter, and (v) given that a DRE is constrained to be hyper-rectangular, the reviewer's guess is that the IRR algorithm might in some cases return DREs whose volume is much smaller than the volume of the corresponding exact RE.

In other words, this could result in decoupled robustness envelopes of much less quality w.r.t. the exact RE counterparts, despite they are computed much more efficiently.
The authors correctly state that "many DREs are possible for a given problem and plan" (Section 2); as a consequence, what we are ideally interested in is the greatest DRE possibly contained in the exact RE, which would turn the problem into an optimization problem, thus greatly increasing its complexity and probably making it not feasible for online utilization.

Indeed, the execution's empirical tests reported in the paper do not seem to highlight this problem; yet, might this become an issue on differently structured domains?

Any comment on this?

Relatively to the Flexible Execution part of the experimental section: it would have been very interesting to see the execution behavior's difference between exact REs and DREs, against the same execution disturbances. The reviewer understands that probably the CCMMZ lends itself poorly for dynamic execution due to its high latency; yet, it would have been interesting to test the quality difference between the obtained REs and DREs, in terms of retained temporal flexibility.

A last question: did the authors think of adopting a dichotomic search approach in the IRR's approximation process of the DRE, instead than an incremental augmentation approach? Would not it converge faster to the selected \beta approximation threshold?

Overall,this seems to be a decent paper, but whose overall novel technical contribution w.r.t. the research that inspires it is not really compelling. The proposed method is rather simple, even though the reviewer does not intend to equate simplicity with scarce significance. The results obtained from the selected benchmarks are promising, but the reviewer's perplexities expressed above still stand. A deeper, more informative, experimentation campaign would have been appreciated.

POST-REBUTTAL COMMENTS
I have carefully read the authors' rebuttals as well as the other reviewers' comments. My overall judgement about this paper has not substantially changed. The tackled problem is of great interest, but im my opinion, more experimental work is required in order to compellingly demonstrate the true advantages of the proposed approach w.r.t. possible existing alternatives, in order to make the analysis more convincing and the results crisper.

 # REVIEW 4

The paper addresses the problem of minimizing the number of required replans during the execution of a temporal plan, handling temporal uncertainty by means of "robustness envelopes" (RE). REs are a method, recently proposed in the literature, to represent the possible scenarios that can be dealt with a given temporal plan without the need of replanning. However, REs are expensive to build and to execute. This paper presents a subclass of REs, called "Decoupled REs" (DRE), that under-approximate REs for a given plan but at the same time are easier to compute and to execute. The paper defines DREs and provides an algorithm for computing them which has the nice feature of being incremental, in such a way that stopping it at any time provides a correct DRE. The paper experimentally evaluates the approach in two aspects: how efficient is the DRE computation, and which effect does their use have on the amount of replanning needed during the execution.

The paper is clearly written and well-motivated. A theoretically interesting but impractical approach (REs) is approximated in such a way to perform well in practice. The incrementality of the algorithm increase its usefulness. The experimental setting appears well designed.

Authors claim it can be adapted to problems where parameters represent things other than actions durations. This point is only remarked by the authors, but I think it would have been worth exploring it better. How to deal with temporal uncertainty is a very well studied field (many relevant works have been cited) but handling other kind of uncertainty (e.g. in consumption rates as the authors write) in a uniform way is, as far as I know, a less walked path. It would be interesting to see something more about this point in the final version of the paper, or in future work. Can the authors elaborate more on this in the rebuttal?

Minor typos:

Page 2, "patametrized" and "liked" instead of "linked"