

Amjad Yousef Majid  
Embedded and Networked Systems Group  
Delft University of Technology  
Mekelweg 4  
2628 CD Delft, The Netherlands  
E-mail: a.y.majid@tudelft.nl  
Phone: +31 61 695 5224

June 24, 2019

**TOSN-2019-0002:** *Coala: Dynamic Task-Based Intermittent Execution for Energy-Harvesting Devices*

Dear Editor,

We would like to thank for handling our submission and providing valuable review results. We are also grateful to the reviewers for their substantial effort spent for evaluating our work.

With this letter we address all comments of the reviewers and point to the modified sections of our revised paper.

With best regards,

in the name of all authors—Amjad Yousef Majid

## Comments by Referee 1

- [C: 1] *In section 2.3, authors mention "these models do not rely on capturing an expensive checkpoint, they are usually faster than the checkpoint-based solution [12, 42]". I don't fully agree with this. It depends on the underlying checkpointing mechanism. If it copies complete stack every time during the checkpoint then yes it's expensive. But checkpointing mechanism like incremental checkpointing [2] (also see the paper by Faycal Ait Aouda), where you only update modified variables, is much better in terms of energy consumption (even better than the task-based solution in my opinion). Also, [12] and [42] are not checkpoint-based solutions.*

As indicated by the reviewer, [12, 42] are not checkpoint systems. They are task-based systems and their results show that task-based systems are faster than checkpoint-based ones. Therefore, we restated their claim and provided the supporting references, which are [12, 42].

Coala addresses deficiencies in the *static* task-based systems and provides more efficient alternative. Therefore, we compared its performance against the state-of-the-art task-based runtime.

- [C: 2] *In section 7.1, authors mention "produce a high number of complete runs". Please mention the exact number iterations. Also please mention the capacitor size for the WIPS 5.1.*

The number of complete runs, which ranges between 4 and 125, and WISP capacitor size are added to Section 7.1.

- [C: 3] *In figure 10, it is not very clear how authors normalized the values for Coala in case of fft for 30cm and 50cm. Also in the text authors state that "This is marked with  $\infty$  signs in Fig. 9", its figure 10.*

We normalized these two special cases on 1 second. This information is added to Section 8.2; additionally, the figure reference is corrected.

- [C: 4] *However, I was hoping to see more evaluations to understand the effect of capacitor size. In the introduction, authors state that "Coala is the only system that eliminates restructuring and re-compilation of applications considering device's energy buffer". However, in the evaluation, I didn't find any graph related to the performance of Coala w.r.t to different capacitor sizes.*

An additional experiment regarding the capacitor size has been conducted and the results are added to Section 8.2.

- [C: 5] *I would have also liked to see a comparison between Coala and Region Formation [2]. Both try to reduce checkpointing overhead.*

As it is indicated in response [R:1], Coala addresses deficiencies in the *static* task-based systems and provides a more efficient alternative. Therefore, we compared its performance against the performance of the state-of-the-art task-based runtime.

- [C: 6] *Also, I would suggest incorporating error bars in all the graphs as your "results were averaged across the duration of the experiment"*

We chose to normalize the results to focus on the speedup (or slowing down) and that is why we omitted the error-bars.

- [C: 7] *In the abstract (and also in the introduction), authors claim that "Coala reduces run time by up to 54%", however, I didn't see this number again anywhere in the evaluation.*

This number is highlighted in Section 8.2.

- [C: 8] *muntioned typos*

these typos are corrected

## Additional questions

- [Q: 1] *Does this paper cite and use appropriate references?: No*

*If not, what important references are missing?: Ratchet and Clank were the first to propose using a timer for breaking up a long task so that it can make forward progress during short power cycles. The*

*authors propose the idea as if they were the first to use it.* In section 4.2, we explicitly mentioned that the timer-based approach is similar to [72]. [72] references the Ratchet paper. Table 1 mentioned Ratchet and Clank by names and they are referenced in our original paper.

### **Additional remarks**

- [R: 1] We have added a new figure to the paper,i.e. Fig. 5. The figure's sole purpose is to aid explaining the concept of *task coalescing*. We hope that this modification further helps in understanding the core contribution of our paper.