This article presents an automatic load-balancing approach for coupled ESM runs and evaluates it based on EC-EARTH. The methodology can be easily understood.

I think the following points should be addressed in the next revision.

1. Although only EC-EARTH is used in the evaluation, the approach proposed should be somehow common to other ESMs. So related works (such as the following list) should be referenced and compared, to show why the approach in this article is novel, more advanced or more effective.

1) D. Kim, J. W. Larson, and K. Chiu, "Automatic performance prediction for load-balancing coupled models," in Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on. IEEE, 2013, pp. 410–417.
2) CESMTuner: An Auto-Tuning Framework for the Community Earth System Model
3) An automatic performance model-based scheduling tool for coupled climate system models
4) Machine-Learning-Based Load Balancing for Community Ice Code Component in CESM
5) https://esmci.github.io/cime/versions/cesm2.2/html/misc_tools/load-balancing-tool.html

2. When obtaining the SYPD (i.e. execution time) for each component model under different parallel settings, how long of the simulation should be, and should the initialization cost be neglected?

3. There should be some conditions for using the new approach especially for prediction script. For example, the model run should be stable enough, which means multiple runs of the same model setting get adjacent runtime. If a HPC runs many applications at the same time, runs of the same model setting may not be stable enough.

4. Does NEMO or IFS have different processor layouts under the same number of PEs. For example, many models have been parallelized based on the decomposition on both X and Y directions. Given 32 PEs, the processor layout can be 1x32, 2x16, 4x8, … Moreover, some models may use hybrid MPI and OpenMP. Should the resource configuration take consideration of such kind of processor layout?

5. What is the key idea of the load-balance workflow? Is there any risk of failure in convergence? If there is no risk, why?

6. The word "optimal" has been used many times in the context. How to prove the "optimal" configuration is absolutely optimal? "near optimal" may be better.

7. Some words in Figure 3 should be improved. For example, "ESM with poor load-balance" should be "ESM run with poor load-balance".

8. Couplers like OASIS-MCT support flexible coupling lags which can transform concurrent run among component models into sequential run where one component model will wait in the whole run of another component model. Can the approach in this article be effective for such kind of ESM run?