The manuscript is concerned with how the public cloud can be used in lieu of traditional on-premises HPC resources for CFD simulations. A key focus is in ensuring that the resulting workflow is reproducible; something which is accomplished through the use of container technology. The promise of containers is that they have the potential to make the entire build environment readily reproducible -- right down to specific library versions. This is an area which is likely to be of interest to readers of CiSE.

Whilst the manuscript is well written the narrative is somewhat muddled. It starts with an overview of reproducible computational research, then pivots to how containers can be employed within the context of cloud computing, then jumping to benchmarks of one particular cloud providers' hardware, and then -- finally -- pivots a final time to a cost analysis of the public cloud for simulations. These are all interesting topics. Indeed, many a conference proceeding has been dedicated to each of these areas. However, by combining them into a single 10 page document this present manuscript fails to do justice to any of these areas. Further, the high degree of specificity will likely have a negative impact the relevance of the manuscript going forwards; a consequence of it being a review of the current state of affairs of a rapidly evolving field. It would therefore be better if it focused on just one or two areas such as "benchmarking cloud computing platforms" or "reproducibility through containers."

Specific technical remarks:

The manuscript only appears to be concerned with reproducibility in the 'now' and pays no consideration to 'link rot' and if one will be able to build and run the proposed containers in 5, 10, or even 20 years. (C.f. Old F77 codes which are expected to remain viable long after the heat death of the universe.) This is one of the most serious issues affecting reproducible research today. Further, the manuscript also glosses over several of the issues regarding deploying containerised applications within an HPC context. One of the most important is with regards to MPI libraries and the fact that many HPC/Cloud environments depend on specific MPI library. For example, Cray machines require Cray MPI, IBM POWER 9 machines require Spectrum MPI, and so on. This results in a 'leaky abstraction' for the container must now concern itself with the idiosyncrasies of the hardware which it is running on.

As highlighted by the authors' in both the manuscript and the Dockerfiles it is necessary (on the Azure platform at least) to use the Intel MPI library. This is highly problematic.

1. It explicitly ties the container to x86 based clusters which support the Intel MPI library (even when the underlying solver software and libraries are portable).
2. The library itself has to be downloaded off a random URL on the Intel website (whose long term stability and availability is suspect).
3. Downloading the library requires acceptance of a proprietary EULA whose terms could easily put an unsuspecting researcher who does not also possess a JD into serious legal trouble with the Intel Corporation -- especially when it comes to moving the container between systems which may be employed for commercial gain. (Aside: whilst in all of the Dockerfiles an rm -rf is performed at the end to remove the Intel components, due to the layered nature of Docker images the compilers are *100%* still present in the resulting images and thus may result in even more legal issues for an unsuspecting researcher.)

4. It is also sub-optimal in the context of CUDA applications for the Intel MPI library is not (to the reviewers' knowledge) CUDA aware and thus parallel CUDA applications are likely to run substantially slower than on say OpenMPI or MVAPICH2 w/GDR.

The result is a highly fragile system which is coupled to the wonts of one particular cloud platform.

Similarly, with the CUDA containers there are also compatibility issues. The most obvious is the container specifying a specific GPU architecture version which may not map onto the hardware of the host. This is another 'leaky abstraction' issue.

Table 1 contains performance data for the Azure cloud nodes and the Colonial One nodes. However, it makes no mention of the memory configuration of the nodes (# of channels and speed rating of the DDR memory). Different configurations here can result in substantial differences with regards to peak memory bandwidth. This is vital as all of the benchmarks fall into the category of 'memory bandwidth bound problems'. Along these lines it is also important to note if ECC was enabled or disabled for the NVIDIA GPUs as this can have a substantial impact on performance.

With regards to the benchmarks in Fig 2. could the authors explain why the lower latency for the Colonial One cluster at small message sizes does not translate into greater peak bandwidth at these sizes?

Fig 3 is somewhat cluttered and amounts to three plots in one. The first appears to be strong scaling on CPUs whilst the second two correspond to weak scaling on GPUs (although it is not elucidated as such). This breakdown seems somewhat arbitrary.

On line 45 of page 6 a remark is made around "One of the requirements for reproducible computational results is to make code available under a public license (allowing reuse and modification by others)." However this does not appear to follow; a piece of work whose code and data is made available for "reproducibility purposes only" would still appear to meet the requirements for facilitating reproducible computational results laid out in the introduction even though the license does not permit reuse or modification.

With regards to the CFD results there does not appear to be any kind of mesh/convergence study nor are any attempts made to quantify the uncertainties on the resulting C_L and C_D coefficients. As such it is difficult to know what to make of the results beyond "they look nice." Although the reviewer appreciates the focus of this paper is not the CFD results per-se it is nevertheless important when presenting numerical results to make a case that the method is resolving all relevant dynamics and that any time-averaged statistics are indeed converged.

Although the cost analysis is interesting without any real basis of comparison it is difficult to glean much from it.