Based on the interesting reviewers' comments, we believe that it is important to better clarify here (and, upon acceptance, in the final version of the paper) our two major contributions.

The **first main contribution** is CuLE, a “strong engineering feat (R2)” which achieves “impressive throughput” (R1) in the simulation of Atari games, which “can have a large impact on the community by simplifying and accelerating Atari experiments: (R2)”, especially if “the authors also make it easy to use (R4)”. This is indeed the case, as the released version of CuLE will be characterized by a python interface which is fully compatible with OpenAI Gym, and apart from installing it, CuLE will not “require any special integration with the RL libraries (R1)”, a potential (but fortunately non-existing) drawback indicated by R1.

R1 suggests that “there aren't clear improvements over existing systems... There is also a strong focus on the evaluation of inference-only speed, which is quite easy... in any parallel system. A strong evaluation section would compare training speed against other distributed / GPU-based RL systems”. We believe that Table I actually contains such comparison, but we prefer reading it from the point of view of R4: CuLE was not designed to achieve the highest possible throughput when compared to large/costly distributed systems, but to “provide access to an accelerated training environment to researchers with limited computational capabilities” and “facilitate research in novel directions that explore thousands of agents without requiring access to a distributed system with hundreds of CPUs”. Indeed, the point that we want to make is that, with a system including 1 or at most 4 GPUs, CuLE's throughput is of the same order of magnitude of that of large (and much more expensive) distributed systems, like IMPALA [7] or a DGX-1 [23].

Regarding R2's comment that “the acceleration won't take into effect unless you use more computation... CuLE runs slower than OpenAI when using a fewer number of environments”, we want to highlight once more that CuLE leverages at best the power of the already-available computational resources by achieving high GPU occupancy and utilization. The cost (in dollars) for scaling from hundreds of CPU/GPU environments to thousands of GPU-CuLE environments is virtually null, as at least one GPU is likely to be present in any system used for RL. Although it remains true that “big companies can easily leverage CuLE to produce results faster and better (R2)”, we disagree with the additional assumption that “small labs will be unable to do so, resulting in slower progress (R2)”: we hope that CuLE can accelerate the workflow of single researchers even if using one GPU, resulting in faster progress, while the practical advantage of scaling to hundreds of thousands of environments still have to be demonstrated, especially for Atari.

The **second main contribution** is related to the “thorough analysis of the improvement over CPU implementations (R3)” and the provided insights, that are not simply intended to give an “analysis of the algorithm bottleneck (R2)”. Instead, based on the CuLE experience and by “recognizing/disentangling different factors when considering "speed" (R3)”, e.g. by analyzing the processes of data generation, transmission, storage, and consumption, we identify system level bottlenecks that have a negative impact on the performance of not only CuLE, but most parts of the existing RL frameworks, and consequently derive design principles that we applied in CuLE but can be easily generalized to design and implement effective RL training systems (including “more OpenAI Gym environments (R3)”) that make use of the available computational resources at best, especially when running on GPUs. To give an example of how these insights may be useful for the research community, R2 notices that “large batches seem required, which limits the selection of applicable algorithms”. More than interpreting this as a simple limitation of CuLE, we believe that it may be a peculiarity of any large throughput simulation system running on GPU, that is at the same time penalized in terms of frames per second per environment. We believe that porting this (and similar) observation to the attention of the researchers can only be beneficial for the design of future simulation libraries and to develop RL algorithms that leverage different learning paradigms, depending on the specific data generation pattern. Strictly related to this topic is the observation of R3 about the sample efficiency of A2C+V-Trace with CuLE: “if I can achieve 800 scores using 120 OpenAI CPU envs within 5M frames, why should I bother using CuLE with 4 GPUs and consume 18M frames?”. To answer this, we make reference to the abundant recent literature on Evolutionary Strategies for RL (e.g. see Evolution Strategies as a Scalable Alternative to Reinforcement Learning, 2017, just to mention one), highlighting that wall-clock time may be more important than sample efficiency from the practical point of view. At the same time we want to highlight once more that the sample inefficiency observed in our paper (Table III) is probably associated to the data generation / consumption pattern, and thus deserves more attention in future research to be better understood. We believe that CuLE can not only trigger, but also facilitate and speed-up this kind of research activity.

Finally, we find very interesting that R1 mentioned the Sample Factory paper in his review — this paper was published on Arxiv after our submission, nevertheless it is based on a system level analysis which is similar in spirit to ours (and can be summarized as “find and remove the bottleneck at system level”), but based on a completely different implementation based on CPU simulation and an RL algorithm using an asynchronous sampler. Assuming that this paper may have been submitted to NeurIPS as well, we believe it may be very instructive to compare the different design approaches and combine the best aspects of our and their solution.

A last note, we can add more training examples / curves in the additional material upon paper acceptance, as requested by R2.