We thank Reviewers (R) 1, 2 and 3 (who gave us marks 6, 7, and 8, respectively) for their pertinent remarks.

R1+R2+R3. More details on monotone operators. We will provide more details on the monotone operator framework (for instance more elaboration around Eq. 6) allowing us to prove the theorems (we shall use the 9th page of the camera ready if our paper is accepted). We already provided some intuitions in the appendix, e.g. in Appendix C. Moreover, we want to recall that the definition of the space E is provided in Appendix C. We will move it to the beginning of Section 4.1 for better understanding.

R1+R3. Straightforward combination of existing techniques? As R3 says, our work closes the theoretical gap by providing the first (and so far the only) optimal first order algorithm for smooth strongly convex decentralized optimization. We obtained this algorithm by "combining existing approaches", but the combination was far from straightforward.

• (Scaman et al.'17) obtained MSDA by simply applying Nesterov acceleration to the dual problem. We instead build upon recent results on the minimization of strongly convex functions under linear constraints (by a first order algorithm, without projecting on the constraints space), see (Salim et. al.'20). Surprisingly, there are only a few algorithms that can solve such problems at a linear rate, and they were proposed only recently.

• APAPC is obtained by applying Nesterov acceleration to the generalized forward-backward algorithm (5) for a sum of operators A and B (see page 5, line 208). Although we managed to do this, this was not an easy task (to say the least), because Nesterov acceleration does not apply to general monotone operators. Even if it it did, a naive approach would lead to a sublinear rate $\mathcal{O}\left(\frac{1}{k^2}\right)$ because $A+B$ is not strongly monotone. On the contrary, we obtained an accelerated linear rate (complexity $\mathcal{O}\left(\left(\sqrt{\chi\kappa}+\chi\right)\log\frac{1}{\epsilon}\right)$) in Theorem 2, which requires careful and deep theoretical analysis of APAPC. Finally, we had to carefully design our generalized Forward Backward algorithm by choosing the space E (see Appendix C), its inner product, and the matrix P as functions of the gossip matrix W.

• In Appendix F we provided an algorithm provably optimal in "# of communication rounds", without using Chebyshev acceleration. The development and analysis of this method required substantial innovation, as we explain in the paper.

Finally, we believe that the *apparent* simplicity of our approach is due to us spending a lot of time making sure the explanations are as intuitive as possible. Many of these intuitions only became clear to us after we have done the analysis; and we provide them for the benefit of the reader. Hence, we view the simplicity as a strength!

R1. Experiments. The networks chosen for evaluating the decentralized method are the ones that were used in (Scaman et al.'17): 10×10 grid and Erdös-Rényi random network with parameter $p = 0.06$. We have now added this detail to the paper. **Regarding the wall-clock-time comparison:** The design of our experiments was very similar to those in (Scaman et al.'17), who assumed that local gradient computation takes one unit of time and communication with neighbors takes time τ. It's easy to observe that in this case [wall clock time] $=$ [# of gradient calls] $+ \tau \times$ [# of communication rounds]. Scaman et al. (2017) used 2 regimes in their experiments: $\tau \gg 1$ (high communication time) and $\tau \ll 1$ (low communication time), which more or less correspond to our plots "# of communication rounds" and "# of gradient calls". This controlled setup is sufficient to verify numerically that our theory (which expresses the optimality of OPAPC in terms of "# of communication rounds" and "# of gradient calls") has predictive power for experiments. Note that our "# of communication rounds" and "# of gradient calls" plots give understanding of how the algorithms will behave with any possible τ (even if we do not know it in practice). We will also produce plots providing "wall clock time", but these will be implementation dependent. Our focus here was not to produce highly performing and fine-tuned software to be benchmarked in this way.

R3. Minor comments. We will put Table 1 in the main paper using the 9th page if the paper is accepted.

R3. Open remark on optimality w.r.t. W. In the line of works on optimal distributed algorithms by Scaman, Hendrikx, Xiao, Bubeck, Bach and Massoulié, lower bounds are obtained by proving the existence of a "bad" gossip matrix and "bad" functions that cannot be optimized faster than the lower bounds by any decentralized algorithm. This includes the decentralized algorithms using the gossip matrix W. Therefore, if one pick a family of functions and a gossip matrix W, they could be "bad" in the above sense, and one cannot beat the lower bounds by a decentralized algorithm using W. However, we agree that, perhaps, the lower bounds theory for these distributed algorithms could be a bit improved by providing lower bounds involving intrinsic properties of the graph, as in the centralized case (there are indeed many gossip matrices for one graph).

R3. Open remark w.r.t. real-life application. Since OPAPC is practical and optimal, the use of OPAPC at scale is definitely the next step in the study of OPAPC. Obviously, this goes beyond the scope of this paper, which contains algorithm development, analysis and testing, but does not and was not supposed to have a software/system development element. But obviously we are also very curious about its performance at scale, and plan to work on this in the future. We are optimistic and confident that OPAPC will outperform existing approaches on average, as indicated by our experiments.