We thank the reviewers for carefully reading the manuscript and providing us with valuable feedback. While the essence of our results seem to be well understood by the reviewers, we address below some specific points that they have raised.

Note that from our perspective *the BC-RED algorithm*, *Theorem 1*, *Theorem 2*, and *the numerical experiments* are equally important as contributions, providing new insights into using *denoisers* (including those based on *deep neural nets*) as priors within block-coordinate estimation. Our manuscript shows that all these ingredients combine synergistically in a novel methodology that is both theoretically rigorous and practically relevant. We made effort to give credit to all the prior work on the topic and will include citations to all publications mentioned by the reviewers.

Reviewer 1. As you correctly inferred, *geometric convergence* can be obtained by strengthening *Assumption 2* to say that g is *strongly convex*. This was omitted from the submitted manuscript due to space. In the context of BC-RED, *separable regularizers* correspond to *separable denoisers*, such as *pixel-wise* or *patch-wise* denoisers. Figure 1 (Left) shows that while DnCNN^* is *not* fully separable, it only requires 5 px padding for optimal performance. We will fix both typos you mention in the revision. We would like to highlight two original contributions in the manuscript, namely the infusion of *deep neural nets* into block-coordinate algorithms in a mathematically rigorous way and establishing an explicit connection between the RED framework and nonsmooth optimization. The revised manuscript will better explain that the traditional analysis from nonsmooth optimization does *not* simply carry over to Theorem 1, since we assume *no objective function* (to accommodate *deep neural net* denoisers, not associated with any regularizer h).

Reviewer 2. Note that Assumption 4 holds for a large number of popular regularizers, including ℓ_2, ℓ_1, and TV penalties. Theorem 1 implies that $\mathbb{E}[\|\mathsf{G}(\boldsymbol{x}^k)\|^2]$ is summable and $\mathbb{E}[\|\mathsf{G}(\boldsymbol{x}^k)\|] \rightarrow 0$, which is the best we can establish for a *convex* g and a generic denoiser. A stronger result – convergence of the iterates to a unique point $\boldsymbol{x}^* \in \mathsf{zer}(\mathsf{G})$ – can be established when g is *strongly convex*. We will clarify L220 to make this more precise. We will clarify L175 to say that we consider a *generic* proximal operator. *No constants blow up*: it is possible to progressively take $\tau \rightarrow \infty$, as in eq. (13), but this leads to a progressive reduction in the step-size γ (see L187). Instead, we empirically found the benefit of tuning τ as a free parameter. We would have loved to be more specific in citations and have a more detailed literature review, but we were dealing with a significant space shortage (we are fully using all the 8-pages allowed by NeurIPS). However, we will certainly include citations to both Danielyan and Tseng in the manuscript. We will clarify that in general PnP and RED are *not* minimizing any functional. We hope that our (possibly suboptimal) notation for U and our schematic illustrations won't preclude the reviewer from considering other merits of our manuscript. We will clarify L17 to say that the true prior might be unknown for *certain* signals, such as natural images. We will release our code with its documentation to GitHub after the reviews; Dropbox was used as a mechanism for anonymous code sharing.

Reviewer 3. While [21] is a great work, it neither analyzes block-coordinate algorithms nor provides an explicit convergence rate. The latter is important for precisely quantifying the computational complexity of BC-RED. The conceptual leap from the traditional RED to our analysis of BC-RED is comparable to the leap from the traditional *gradient descent* to the Nesterov's analysis of *coordinate descent methods* [23], which is certainly not minor. We share your enthusiasm for Theorem 2, but it will be challenging to find more space without significant revisions. We provide some *time comparisons* in Figure 1 (Center and Right), showing that on our machine (see Section F in the supplement) an efficient implementation of BC-RED *can* be much faster than RED, where the speed depends on the structure of the measurement matrix and the denoiser. However, the *speed* is only one of many potential advantages of BC-RED, as it can offer *scalability* through other mechanisms, such as effective memory management and distributed implementation.

Padding	SNR
0 px	27.64 dB
5 px	28.11 dB
10 px	28.12 dB
20 px	28.12 dB
40 px	28.12 dB

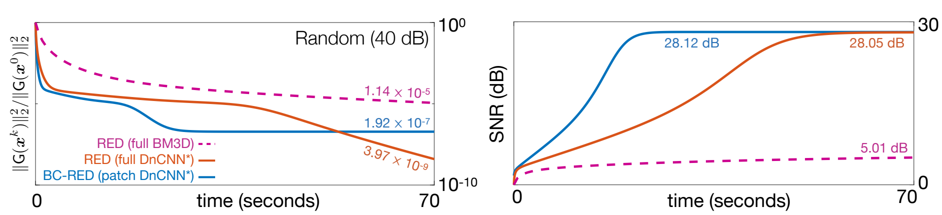

Figure 1: ***Left:*** The performance of BC-RED for the Random matrix with 40 dB noise and *patch-wise* DnCNN^*, where the denoiser input includes an additional padding around the patch, while the output has the size of the patch. The lower SNR for 0 px suggests *non-separability* of DnCNN^*; yet, a small 5 px padding is sufficient for matching the performance of the *full-image* DnCNN^*. ***Center and Right:*** The convergence speed of BC-RED under *patch-wise* DnCNN^* with 40 px padding for the same setting as *Left*. Distance to $\mathsf{zer}(\mathsf{G})$ – corresponding to the *full-image* denoiser – and SNR are plotted against time. As a reference, we provide the convergence of RED using the full-image DnCNN^* and BM3D denoisers. Since the patch-wise denoiser only *approximates* the full-image denoiser, the final accuracy of BC-RED to $\mathsf{zer}(\mathsf{G})$ is 1.92×10^{-7}. Yet, BC-RED still matches the SNR performance of the full-gradient RED and does this substantially faster due to its better convergence rate and reduced denoising complexity (due to patch-wise denoising). Note also the slow convergence of RED using the full-image BM3D, due to high complexity of denoising.