We thank the reviewers for their thorough comments and feedback.

Reviewer #1: *The first and main concern is that, without the supplementary material, the article is not self-contained nor reproducible.*

We propose to modify the article as follows to address this concern.

One concrete example of coupling of MCMC algorithm will be moved to the main text. These couplings build on previous work but indeed are not widely known. Thus we will add a full description of a coupled kernel in Section 2.2.1.

We also propose to add some motivation and explanation (five to ten lines) for the proposed bounds, just after Theorem 2.5 in the main text. The supplementary material will still contain the full formal proof.

A secondary concern of the reviewer is a lack of discussion of some relevant articles on kernel methods.

We thank the reviewer for pointing out references which we will add. The articles mentioned provide sample quality assessments but are not directly approximating the TV or the Wasserstein distance between π_t and π. Thus, we are not aware of articles where these types of assessment are used to choose a burn-in value, or to obtain plots similar to the ones we show; our understanding is that these tools have been used to assess the approximation obtained post burn-in, or compare the bias of asymptotically biased samplers.

We propose to add some motivation for considering the TV and 1-Wasserstein distances, in Section 2, as these distances are practically useful: the TV controls the error made in approximating probability masses under the target (e.g. when plotting histograms of the target marginals, or when calculating credible intervals of posterior distributions), while the 1-Wasserstein controls all first moments. Further, TV and Wasserstein are used in most theoretical studies, which allows for comparison between our proposed bounds and established results, as we illustrate in Section 3.3.

Among other relevant papers we will also add a reference to "A simulation approach to convergence rates for Markov chain Monte Carlo algorithms" by Rosenthal & Cowles.

We will follow the reviewer's insightful suggestions on how to improve the presentation of the article and remove some redundancies. In passing $a \vee b$ denoted the maximum between a and b, which we can denote by $\max(a, b)$ instead.

Reviewer #2: *The first criticism is about the originality of the proposed bounds.*

We agree with the reviewer that the introduction of a lag of L instead of a lag of 1 as in reference [16] looks incremental from a mathematical point or view. However it is key to the obtention of practical bounds.

From the methodological point of view, the contribution is original. The idea of the upper bound on the TV was suggested in the last paragraphs of [16], but not implemented or discussed anywhere, as far as we know. The proposed method resembles Johnson's diagnostics, which we discuss in details, but it seems much more generically applicable.

The second and main concern is about the choice of L, its impact on the bounds, and its cost.

We propose to elaborate the discussion on the choice of L and the appeal of using $L > 1$. We can add an explicit discussion of the cost of obtaining $\tau^{(L)}$ as a function of L, and how the TV upper bounds becomes less vacuous for all t as $L \to \infty$. Overall we do not have strong theoretical results to guide the choice of L, and hope that the article will motivate further research on the topic. In the meantime, Figure 2 and, mostly, Figure 3 show the practical benefit of choosing $L > 1$.

A final concern is about discussing the verification of assumptions in the context of the applications. We will add comments on this but will mostly refer to [16] where such assumptions are discussed in details.

Reviewer #3:

We thank the reviewer for their supporting words regarding our efforts to point out the weaknesses of the proposed method. Regarding software implementation, designing coupling schemes for any given MCMC algorithm can require some work, nonetheless this has been done already for several popular algorithms, e.g. Hamiltonian Monte Carlo. These couplings could indeed be integrated into some software such as PyMC3, as suggested by the reviewer. We can add some discussion on this in the discussion.