

1 We thank the reviewers for carefully reading the manuscript and providing their comments.

2 **R1+R3: Novelty of results.** As the first work to our knowledge that applies coding along the temporal and “spatial”  
3 dimensions for distributed Matrix Multiplication (MM), we believe the paper could spark interest in follow-up theoretical  
4 and experimental investigations. All previous works (e.g. Polynomial Codes (PC’s)) have focused on one-shot MM  
5 that involves coding only across workers (i.e., spatial dimension). The advantages of our framework are two-fold (1)  
6 by exploiting the temporal dimension, our proposed schemes tolerate more straggler patterns than PC under the same  
7 normalized load at the workers. In particular, if PC tolerates  $S$  stragglers in each round, then under the same worker  
8 load, our proposed schemes tolerate a total of  $(T + 1) \cdot S$  stragglers in each sliding window of  $T + 1$  consecutive  
9 rounds, with delay  $T$  (please see “Motivating example” in Sec. 2.1 of supplementary material). (2) Our schemes exploit  
10 the feedback of previously occurred straggler patterns from previous rounds to adapt the computations to be performed.  
11 In particular, in the IDIP scheme, the master judiciously opts for “uncoded mini-tasks” wherever possible, based on  
12 such feedback, with the objective to reduce overall decoding complexity. Previous works did not explore the temporal  
13 dimension and hence, did not have access to this feedback. The other theoretical results in the paper include (i) showing  
14 the optimality of DIP/IDIP schemes under the  $(N, W)$ -Straggler Model (SM) and (ii) an analytical comparison of the  
15 expected run-time of various schemes (which illustrates advantage of our schemes) under an i.i.d. SM.

16 **R1: On the choice of stochastic SM’s** We emphasize that although the performance analysis of our coding schemes  
17 assumes certain stochastic models, our coding schemes can be applied irrespective of the actual pattern of stragglers.  
18 Please refer to the discussion on page 5 (last paragraph) in main paper. Thus, our coding schemes are not limited by the  
19 assumed stochastic model. The i.i.d. SM in Section 4 enables us to develop analytical performance bounds and develop  
20 insights. The use of Fritchman model in Section 5 is motivated by its ability to model occurrence of bursty stragglers. In  
21 [13], the authors note that speed variation in an Amazon EC2 credit-based instance can be closely modeled by a similar  
22 bursty stochastic model. Finally, we note that our claim in Section 4.3 of the supplementary material that  $p^{\text{IDIP}} \leq p^{\text{poly}}$ ,  
23 i.e., the master is more likely to require straggler nodes to complete their jobs in the PC scheme, does not rely on i.i.d.  
24 model, but instead holds for any stochastic model. This implies that coding across time dimension always improves  
25 upon one-shot PC, regardless of the SM.

26 **R1: Paper is light on experimental evaluation** Experimental results in Sec. 5 report encoding/decoding/processing  
27 times measured on workers and master, when training an NN. Consistent with prior work, we artificially injected  
28 stragglers during training to develop insights into performance. E.g., in Fig 5(f), we demonstrate that when bursty  
29 stragglers are introduced, the instantaneous load of IDIP scheme remains consistently lower than PC, while the DIP  
30 scheme requires high load. Thus, the proposed IDIP scheme could have significant impact in practice when bursty  
31 stragglers are reported. We note that the contribution of the paper is to propose new coding schemes along with  
32 optimality guarantees, analysis for i.i.d. SM as well as experimental evaluation to develop insights into the performance.  
33 A large scale experimental study, while interesting, is beyond the scope of the present paper.

34 **R2+R3: Applicability of the scheme in general** While we focus on training multiple NN simultaneously in the paper,  
35 the framework suits well in any application where the master is interested in finishing quickly a collection of multiple  
36 independent sequences of MM’s (dependencies are permitted within a sequence). Clearly, this is applicable if one is  
37 interested in solving multiple systems of linear equations through an iterative algorithm such as Jacobi method. Cloud  
38 platforms providing route planning, page ranking services solve multiple systems of equations in every second.

39 **R1+R3: Comparison with Streaming Codes (SC’s)** While we are aware of the literature on SC’s, there are fundamen-  
40 tal differences in the two approaches, because of which, SC constructions do not seem to be applicable to MM problems.  
41 For instance, consider the SC toy example, where packets  $p_1, p_2, p_1 + p_2$  are transmitted in rounds 1, 2, 3 (can recover  
42 any lost packet with delay 2). Extending this to our setting, suppose a worker computes  $A_1x_1, A_2x_2$  and  $A_1x_1 + A_2x_2$   
43 in successive rounds. This scheme is sub-optimal as  $A_1x_1 + A_2x_2$  involves 2 matrix-vector multiplications.

44 **R1: Additional comments** Coding across jobs may not be possible as matrices in different jobs may have incompatible  
45 dimensions or data. Moreover, for the  $(N, W)$ -SM, our proposed schemes are already optimal.

46 **R2: Additional comments** (1)  $u_i$  (subscript was missing) indicates #uncoded mini-tasks of job- $i$ . (2) We missed  
47 to mention that i.i.d. SM is a stochastic extension of the deterministic  $(N, W)$ -SM. (3) Under the  $(N, W)$ -SM, IDIP  
48 scheme does perform better than DIP. However, in Fig. 1, the performance is plotted w.r.t. i.i.d. SM. DIP scheme  
49 is allowed to increase instantaneous worker load in future rounds to handle failed mini-tasks, whereas IDIP scheme  
50 waits for stragglers to complete their mini-tasks (round duration expands by  $\alpha$ ) when non-ideal straggler patterns are  
51 encountered. Our numerical simulations indicate that under i.i.d. SM, for  $\alpha \gg 1$ , IDIP scheme gets penalized more.

52 **R3: Connections to delay-vs-throughput tradeoff, prior work** Apart from streaming codes, we do not see any  
53 connection between ours and the delay vs. throughput tradeoff framework for communication networks. While we  
54 believe we have already addressed in the paper how our coding approach differs from the existing one-shot approaches,  
55 we will include more details in the revision to bring out further, the differences and motivation for our approach.