Lighter-Communication Distributed Machine Learning via Sufficient Factor Broadcasting — Supplementary Material

Pengtao Xie, Jin Kyu Kim, Yi Zhou*, Qirong Ho†, Abhimanu Kumar§, Yaoliang Yu, Eric Xing
School of Computer Science, Carnegie Mellon University;
*Department of EECS, Syracuse University;
†Institute for Infocomm Research, A*STAR, Singapore; §Groupon

1 Proof of Convergence

Proof of Theorem 1

Proof. Let $\mathcal{F}^c := \sigma\{I_p^\tau : p = 1, \ldots, P, \tau = 1, \ldots, c\}$ be the filtration generated by the random samplings I_p^τ up to iteration counter c, *i.e*let@tokeneonedot, the information up to iteration c. Note that for all p and c, $\mathbf{W}_p^c$ and $\mathbf{W}^c$ are $\mathcal{F}^{c-1}$ measurable (since $\tau_p^q(c) \leq c-1$ by assumption), and I_p^c is independent of $\mathcal{F}^{c-1}$. Recall that the partial update generated by machine p at its c-th iteration is

$$U_p(\mathbf{W}_p^c, I_p^c) = -\eta_c |S_p| \sum_{j \in I_p^c} \nabla f_j(\mathbf{W}_p^c)$$

Then it holds that

$$U_p(\mathbf{W}_p^c) = \mathbb{E}[U_p(\mathbf{W}_p^c, I_p^c) | \mathcal{F}^{c-1}] = -\eta_c \nabla F_p(\mathbf{W}_p^c)$$

(Note that we have suppressed the dependence of U_p on the iteration counter c.)

Then, we have

$$\mathbb{E}\left[\sum_{p=1}^P U_p(\mathbf{W}_p^c, I_p^c) \mid \mathcal{F}^{c-1}\right] = \sum_{p=1}^P \mathbb{E}[U_p(\mathbf{W}_p^c, I_p^c) \mid \mathcal{F}^{c-1}] = \sum_{p=1}^P U_p(\mathbf{W}_p^c) \tag{1}$$

Similarly we have

$$\begin{aligned} &\mathbb{E}\left[\left\| \sum_{p=1}^P U_p(\mathbf{W}_p^c, I_p^c)\right\|_2^2 \mid \mathcal{F}^{c-1}\right] \\ &= \sum_{p,q=1}^P \mathbb{E}[\langle U_p(\mathbf{W}_p^c, I_p^c), U_q(\mathbf{W}_q^c, I_q^c)\rangle \mid \mathcal{F}^{c-1}] \\ &= \sum_{p,q=1}^P \langle U_p(\mathbf{W}_q^c), U_q(\mathbf{W}_q^c)\rangle + \sum_{p=1}^P \mathbb{E}\left[\|U_p(\mathbf{W}_p^c, I_p^c) - U_p(\mathbf{W}_p^c)\|_2^2 \mid \mathcal{F}^{c-1}\right] \end{aligned} \tag{2}$$

The variance term in the above equality can be bounded as

$$\begin{aligned} &\sum_{p=1}^P \mathbb{E}\left[\|U_p(\mathbf{W}_p^c, I_p^c) - U_p(\mathbf{W}_p^c)\|_2^2 \mid \mathcal{F}^{c-1}\right] \\ &= \underbrace{\eta_c^2 \sum_{p=1}^P \mathbb{E}\left[\left\| |S_p| \sum_{j \in I_p^c} \nabla f_j(\mathbf{W}_p^c) - \nabla F_p(\mathbf{W}_p^c)\right\|_2^2 \mid \mathcal{F}^{c-1}\right]}_{\hat{\sigma}^2 P} \\ &\leq \eta_c^2 \hat{\sigma}^2 P \end{aligned} \tag{3}$$

Now use the update rule $\mathbf{W}_p^{c+1} = \mathbf{W}_p^c + \sum_{p=1}^P U_p(\mathbf{W}_p^c, I_p^c)$ and the descent lemma [1], we have

$$\begin{aligned} &F(\mathbf{W}^{c+1}) - F(\mathbf{W}^c) \\ &\leq \langle \mathbf{W}^{c+1} - \mathbf{W}^c, \nabla F(\mathbf{W}^c)\rangle + \tfrac{L_F}{2}\|\mathbf{W}^{c+1} - \mathbf{W}^c\|_2^2 \\ &= \langle \textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c, I_p^c), \nabla F(\mathbf{W}^c)\rangle + \tfrac{L_F}{2}\| \textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c, I_p^c)\|_2^2 \end{aligned} \tag{4}$$

Then take expectation on both sides, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\ F(\mathbf{W}^{c+1}) - F(\mathbf{W}^c) \mid \mathcal{F}^{c-1}\right] \\
&\leq \langle \textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c), \nabla F(\mathbf{W}^c)\rangle + \frac{L_F \eta_c^2 \hat{\sigma}^2 P}{2} + \frac{L_F}{2}\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 \\
&= (\tfrac{L_F}{2} - \eta_c^{-1})\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \frac{L_F \eta_c^2 \hat{\sigma}^2 P}{2} - \eta_c^{-1}\langle \textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c), \sum_{p=1}^P [U_p(\mathbf{W}^c) - U_p(\mathbf{W}_p^c)]\rangle \\
&\leq (\tfrac{L_F}{2} - \eta_c^{-1})\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \frac{L_F \eta_c^2 \hat{\sigma}^2 P}{2} + \|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\| \sum_{p=1}^P L_p \|\mathbf{W}^c - \mathbf{W}_p^c\|
\end{aligned} \tag{5}$$

Now take expectation w.r.t all random variables, we obtain

$$\begin{aligned}
&\mathbb{E}\left[F(\mathbf{W}^{c+1}) - F(\mathbf{W}^c)\right] \\
&\leq (\tfrac{L_F}{2} - \eta_c^{-1})\mathbb{E}\left[\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2\right] + \textstyle\sum_{p=1}^P L_p \mathbb{E}\left[\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\| \|\mathbf{W}^c - \mathbf{W}_p^c\|\right] + \frac{L_F \eta_c^2 \sigma^2 P}{2}
\end{aligned} \tag{6}$$

Next we proceed to bound the term $\mathbb{E}\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\| \|\mathbf{W}^c - \mathbf{W}_p^c\|$. We list the auxiliary update rule and the local update rule here for convenience.

$$\begin{aligned}
\mathbf{W}^c &= \mathbf{W}^0 + \textstyle\sum_{q=1}^P \sum_{t=0}^{c-1} U_q(\mathbf{W}_q^t, I_q^t), \\
\mathbf{W}_p^c &= \mathbf{W}^0 + \textstyle\sum_{q=1}^P \sum_{t=0}^{\tau_p^q(c)} U_q(\mathbf{W}_q^t, I_q^t).
\end{aligned} \tag{7}$$

Now subtract the above two and use the bounded delay assumption $0 \leq (c-1) - \tau_p^q(c) \leq s$, we obtain

$$\begin{aligned}
&\|\mathbf{W}^c - \mathbf{W}_p^c\| \\
&= \|\textstyle\sum_{q=1}^P \sum_{t=\tau_p^q(c)+1}^{c-1} U_q(\mathbf{W}_q^t, I_q^t)\| \\
&\leq \|\textstyle\sum_{q=1}^P \sum_{t=c-s}^{c-1} U_q(\mathbf{W}_q^t, I_q^t)\| + \|\sum_{q=1}^P \sum_{t=c-s}^{\tau_p^q(c)} U_q(\mathbf{W}_q^t, I_q^t)\| \\
&\leq \textstyle\sum_{t=c-s}^{c-1} \|\sum_{q=1}^P U_q(\mathbf{W}_q^t, I_q^t)\| + \eta_{c-s} G
\end{aligned} \tag{8}$$

where the last inequality follows from the facts that η_c is strictly decreasing, and $\|\sum_{q=1}^P \sum_{t=c-s}^{\tau_p^q(c)} \nabla F_q(\mathbf{W}_q^t, I_q^t)\|$ is bounded by some constant G since ∇F_q is continuous and all the sequences $\mathbf{W}_p^c$ are bounded. Thus by taking expectation, we obtain

$$\begin{aligned}
&\mathbb{E}\left[\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|\ \|\mathbf{W}^c - \mathbf{W}_p^c\|\right] \\
&\leq \mathbb{E}\left[\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\| \Big(\textstyle\sum_{t=c-s}^{c-1} \|\sum_{q=1}^P U_q(\mathbf{W}_q^t, I_q^t)\| + \eta_{c-s} G \Big)\right] \\
&= \textstyle\sum_{t=c-s}^{c-1} \mathbb{E}\left[\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\| \|\sum_{q=1}^P U_q(\mathbf{W}_q^t, I_q^t)\|\right] + \eta_{c-s} G \cdot \mathbb{E}\left[\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|\right] \\
&\leq \textstyle\sum_{t=c-s}^{c-1} \mathbb{E}\left[\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \|\sum_{q=1}^P U_q(\mathbf{W}_q^t, I_q^t)\|_2^2\right] + \mathbb{E}\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_{c-s}^2 G^2 \\
&\leq (s+1)\mathbb{E}\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_{c-s}^2 G^2 + \textstyle\sum_{t=c-s}^{c-1} \left[\mathbb{E}\|\sum_{q=1}^P U_q(\mathbf{W}_q^t)\|_2^2 + \eta_t^2 \sigma^2 P\right]
\end{aligned} \tag{9}$$

Now plug this into the previous result in (6):

$$\begin{aligned}
&\mathbb{E}F(\mathbf{W}^{c+1}) - \mathbb{E}F(\mathbf{W}^c) \\
&\leq (\tfrac{L_F}{2} - \eta_c^{-1})\mathbb{E}\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + (s+1)L\mathbb{E}\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_{c-s}^2 G^2 L \\
&+ \textstyle\sum_{t=c-s}^{c-1} \left[L\mathbb{E}\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_t^2 L \sigma^2 P\right] + \frac{L_F \eta_c^2 \sigma^2 P}{2} \\
&= (\tfrac{L_F}{2} + (s+1)L - \eta_c^{-1})\mathbb{E}\|\textstyle\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_{c-s}^2 G^2 L \\
&+ \textstyle\sum_{t=c-s}^{c-1} \left[L\mathbb{E}\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2 + \eta_t^2 L \sigma^2 P\right] + \frac{L_F \eta_c^2 \sigma^2 P}{2}
\end{aligned} \tag{10}$$

Sum both sides over $c = 0, ..., C$:

$$\begin{aligned}
&\mathbb{E}F(\mathbf{W}^{C+1}) - \mathbb{E}F(\mathbf{W}^0) \\
&\leq \textstyle\sum_{c=0}^C \left[(\tfrac{L_F}{2} + (2s+1)L - \eta_c^{-1})\mathbb{E}\|\sum_{p=1}^P U_p(\mathbf{W}_p^c)\|_2^2\right] + (L\sigma^2 P s + \frac{L_F \sigma^2 P}{2}) \textstyle\sum_{c=0}^C \eta_c^2 + G^2 L \sum_{c=0}^C \eta_{c-s}^2
\end{aligned} \tag{11}$$

After rearranging terms we finally obtain

$$\begin{aligned}&\textstyle\sum_{c=0}^{C}\left[\eta_c^2(\eta_c^{-1}-\frac{L_F}{2}-2(s+1)L)\mathbb{E}\|\sum_{p=1}^{P}\nabla F_p(\mathbf{W}_p^c)\|_2^2\right]\\&\leq \mathbb{E}F(\mathbf{W}^0)-\mathbb{E}F(\mathbf{W}^{C+1})+(L\sigma^2Ps+\tfrac{L_F\sigma^2P}{2})\textstyle\sum_{c=0}^{C}\eta_c^2+G^2L\sum_{c=0}^{C}\eta_{c-s}^2\\&\leq F(\mathbf{W}^0)-\inf F+(L\sigma^2Ps+LG^2+\tfrac{L_F\sigma^2P}{2})\textstyle\sum_{c=0}^{C}\eta_c^2.\end{aligned}\tag{12}$$

It then follows that

$$\begin{aligned}&\min_{c=1,\ldots,C}\mathbb{E}\left[\|\textstyle\sum_{p=1}^{P}\nabla F_p(\mathbf{W}_p^c)\|_2^2\right]\\&\leq\frac{F(\mathbf{W}^0)-\inf F+(L\sigma^2Ps+LG^2+\frac{L_F\sigma^2P}{2})\sum_{c=0}^{C}\eta_c^2}{\sum_{c=0}^{C}\eta_c-(L_F/2+2L(s+1))\eta_c^2}\\&\approx\frac{F(\mathbf{W}^0)-\inf F+(L\sigma^2Ps+LG^2+\frac{L_F\sigma^2P}{2})\sum_{c=0}^{C}\eta_c^2}{\sum_{c=0}^{C}\eta_c},\end{aligned}$$

where we ignore the higher order term $(L_F/2+2L(s+1))\eta_c^2$ in the last equation for simplicity, and this does not affect the order of the final estimate since we will use a diminishing stepsize $\eta_c=O(1/\sqrt{c})$. Now we can apply [2, Theorem 4.2] to the last equation to conclude that

$$\begin{aligned}&\min_{c=1,\ldots,C}\mathbb{E}\left[\|\textstyle\sum_{p=1}^{P}\nabla F_p(\mathbf{W}_p^c)\|_2^2\right]\\&\leq\sqrt{\frac{(F(\mathbf{W}^0)-\inf F)[2L(\sigma^2Ps+G^2)+L_F\sigma^2P]}{2C}}\end{aligned}$$

with the choice of stepsize

$$\eta_c=\sqrt{\frac{8(F(\mathbf{W}^0)-\inf F)}{2L(\sigma^2Ps+G^2)+L_F\sigma^2P}}\frac{1}{\sqrt{c}}.$$

Hence, we must have

$$\liminf_{c\to\infty}\mathbb{E}\|\textstyle\sum_{p=1}^{P}\nabla F_p(\mathbf{W}_p^c)\|=0\tag{13}$$

proving the first claim.

On the other hand, the bound of $\|\mathbf{W}^c-\mathbf{W}_p^c\|$ in (8) gives

$$\|\mathbf{W}^c-\mathbf{W}_p^c\|\leq\textstyle\sum_{t=c-s}^{c-1}\eta_t\|\sum_{q=1}^{P}|S_q|\sum_{j\in I_q^t}\nabla f_j(\mathbf{W}_q^t)\|+\eta_{c-s}G\tag{14}$$

By assumption the sequences $\{\mathbf{W}_p^c\}_{p,c}$ and $\{\mathbf{W}^c\}_c$ are bounded and the gradient of f_j is continuous, thus $\nabla f_j(\mathbf{W}_q^t)$ is bounded. Now take $c\to\infty$ in the above inequality and notice that $\lim_{c\to\infty}\eta_c=0$, we have $\lim_{c\to\infty}\|\mathbf{W}^c-\mathbf{W}_p^c\|=0$ almost surely, proving the second claim.

Lastly, the Lipschitz continuity of ∇F_p further implies

$$0=\liminf_{c\to\infty}\mathbb{E}\|\textstyle\sum_{p=1}^{P}\nabla F_p(\mathbf{W}_p^c)\|\geq\liminf_{c\to\infty}\mathbb{E}\|\sum_{p=1}^{P}\nabla F_p(\mathbf{W}^c)\|=\liminf_{c\to\infty}\mathbb{E}\|\nabla F(\mathbf{W}^c)\|=0\tag{15}$$

Thus there exists a common limit point of $\mathbf{W}^c$, $\mathbf{W}_p^c$ that is a stationary point almost surely. The proof is now complete. □

2 Sample Code for Sparse Coding

Figure 2 shows the sample code of implementing sparse coding in SFB. D is the feature dimensionality of data and J is the dictionary size. Users need to write a SF computation function to specify how to compute the sufficient factors: for each data sample $\mathbf{x}_i$, we first compute its sparse code $\mathbf{a}$ based on the dictionary $\mathbf{B}$ stored in the parameter matrix sc.para_mat. Given $\mathbf{a}$, the sufficient factor $\mathbf{u}$ can be computed as $\mathbf{Ba}-\mathbf{x}_i$ and the sufficient factor $\mathbf{v}$ is simply $\mathbf{a}$. In addition, users need to provide a proximal operator function to specify how to project $\mathbf{B}$ to the ℓ_2 ball constraint set.

3 Implementation Details

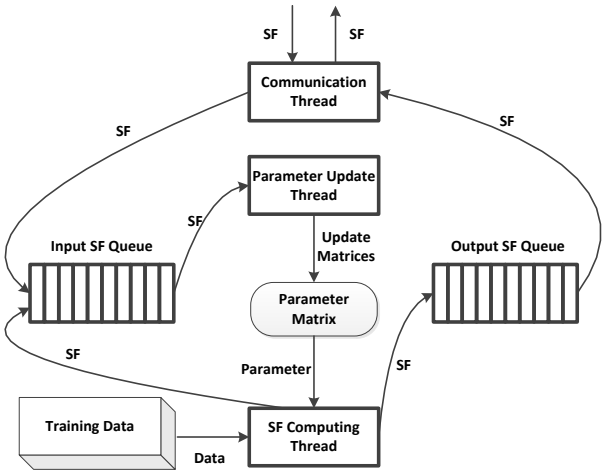

Figure 2: Implementation details on each worker in SFB.

Figure 2 shows the implementation details on each worker in SFB. Each worker maintains three threads: SF computing thread, parameter update thread and communication thread. Each worker holds a local copy of the parameter matrix and a partition of the training data. It also maintains an input SF queue which stores the sufficient factors computed locally and received remotely and an output SF queue which stores SFs to be sent to other workers. In each iteration, the SF computing thread checks the consistency policy detailed in the main paper. If permitted, this thread randomly chooses a minibatch of samples from the training data, computes the SFs and pushes them to the input and output SF queue. The parameter update thread fetches SFs from the input SF queue and uses them to update the parameter matrix. In proximal-SGD/SDCA, the proximal/dual operator function (provided by the user) is automatically called by this thread as a function pointer. The communication thread receives SFs from other workers and pushes them into the input SF queue and broadcasts SFs in the output SF queue to other workers. One worker is in charge of measuring the objective value. Once the algorithm converges, this worker notifies all other workers to terminate the job. We implemented SFB in C++. OpenMPI was used for communication between workers and OpenMP was used for multicore parallelization within each machine.

```
sfb_app sc ( int D, int J , int staleness)
//SF computation function
function compute_sf ( sfb_app sc ):
  while ( ! converge ):
    X=sample_minibatch ()
    foreach x_i in X:
      //compute sparse code
      a = compute_sparse_code ( sc.para_mat, x_i )
      //sufficient factor u_i
      sc.sf_list[i].write_u ( sc.para_mat * a-x_i )
      //sufficient factor v_i
      sc.sf_list[i].write_v ( a )
    commit()
//Proximal operator function
function prox ( sfb_app sc ):
  foreach column b_i in sc.para_mat:
    if ||b_i||_2 > 1:
      b_i = b_i / ||b_i||_2
```

Figure 1: Sample code of sparse coding in SFB

The decentralized architecture of SFB makes it robust to machine failures. If one worker fails, the rest of workers can continue to compute and broadcast the sufficient factors among themselves. In addition, SFB possesses high elasticity [3]: new workers can be added and existing workers can be taken offline, without restarting the running framework. A thorough study of fault tolerance and elasticity will be left for future work.

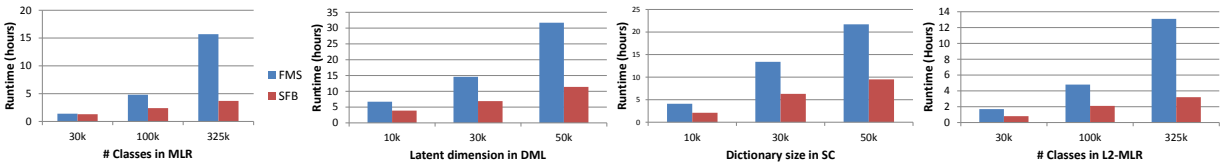

Figure 3: Convergence time versus model size for MLR, DML, SC, L2-MLR (left to right), under SSP with staleness=20.

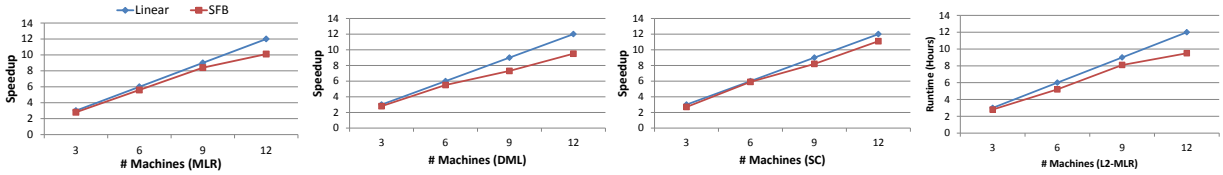

Figure 4: SFB scalability with varying machines, for MLR, DML, SC, L2-MLR (left to right), under SSP with staleness=20.

4 Additional Experiment Results

Figure 3 shows the convergence time versus model size for MLR, DML, SC, L2-MLR, under SSP with staleness=20. Figure 4 shows SFB scalability with varying machines under SSP with staleness=20, for MLR, DML, SC, L2-MLR. Figure 5 shows the iteration throughput (left) and iteration quality (right) for MLR, under SSP (staleness=20). The minibatch size was set to 100 for both SFB and FMS. As can be seen from the rightmost graph, SFB has a lightly worse iteration quality than FMS. The reason we conjecture is that the centralized architecture of FMS is more robust and stable than the decentralized architecture of SFB. On the other hand, the iteration throughput of SFB is much higher than FMS as shown in the leftmost graph. Figure 6 shows the convergence time of MLR and L2-MLR versus varying Q in partial broadcasting, under SSP (staleness=20). Figure 7 shows the communication volume of four models under BSP. As shown in the figure, the communication volume of SFB is significantly lower than FMS and Spark. Under the BSP consistency model, SFB and FMS share the same iteration quality, hence need the same number of iterations to converge. Within each iteration, SFB communicates vectors while FMS transmits matrices. As a result, the communication volume of SFB is much lower than FMS.

References

[1] Dimitri P. Bertsekas and John N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, 1989.

[2] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167 – 175, 2003.

[3] Mu Li, David G Andersen, Jun Woo Park, Alexander J Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J Shekita, and Bor-Yiing Su. Scaling distributed machine learning with the parameter server. In *USENIX Symposium on Operating Systems Design and Implementation*, 2014.

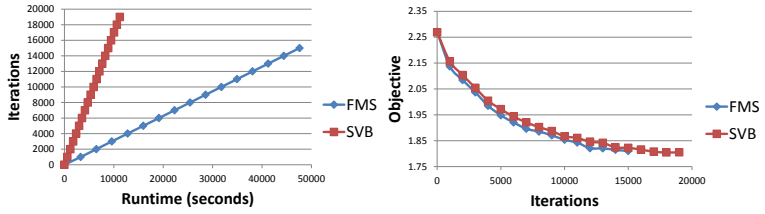

Figure 5: MLR iteration throughput (left) and iteration quality (right) for MLR under SSP (staleness=20).

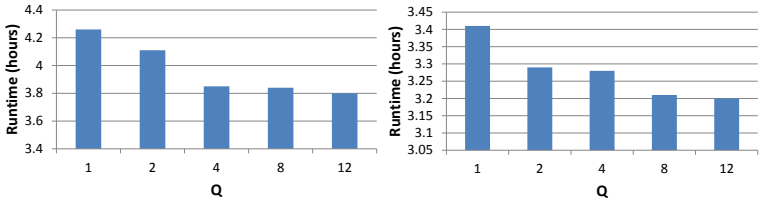

Figure 6: Convergence time versus Q in partial broadcasting for MLR (left) and L2-MLR (right), under SSP (staleness=20).

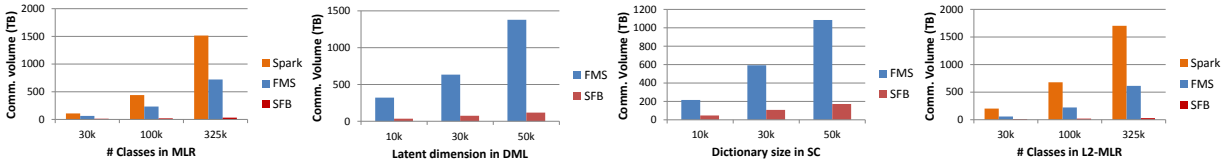

Figure 7: Communication volume for MLR, DML, SC, L2-MLR (left to right) under BSP.