A. Appendix

A.1. Proof of Unbiasedness and Convergence for JointSQ

Our quantizer selection is similar to the random uniform quantizer as referenced in [1]. We first prove the unbiasedness and convergence for the general form of Co-compressor. For a gradient vector $\mathbf{g}$, the i-th gradient element is quantized as follows:

$$\mathcal{Q}_b\left[g_i\right]=\|\mathbf{g}\| \cdot \operatorname{sgn}\left(g_i\right) \cdot \zeta\left(g_i, s\right),$$

where $\|\mathbf{g}\|$ is the l_2 norm of $\mathbf{g}$; $\operatorname{sgn}\left(g_i\right)=\{+1,-1\}$ is the sign of g_i; s is the quantization level. If we use b bits to quantize g_i, we will use one bit to represent its sign and the other $b-1$ bits to represent $\zeta\left(g_i, s\right)$, thus resulting in a quantization level $s=2^{b-1}-1$. And $\zeta\left(g_i, s\right)$ is an unbiased stochastic function that maps scalar $\left|g_i\right| /\|\mathbf{g}\|$ to one of the values in set $\{0,1/s,2/s,\ldots,s/s\}$: if $\left|g_i\right| /\|\mathbf{g}\| \in[l/s,(l+1)/s]$, we have:

$$\zeta\left(g_i, s\right)= \begin{cases}l / s, & \text { with probability } 1-p_r, \\ (l+1) / s, & \text { with probability } p_r=s \frac{\left|g_i\right|}{\|\mathbf{g}\|}-l .\end{cases}$$

So we have:

$$\begin{aligned}
\mathbb{E}\left[\zeta\left(g_i, s\right)\right]= & \frac{l}{s}\left[1-s \frac{\left|g_i\right|}{\|\mathbf{g}\|}+l\right] \\
& +\frac{l+1}{s}\left[s \frac{\left|g_i\right|}{\|\mathbf{g}\|}-l\right]=\frac{\left|g_i\right|}{\|\mathbf{g}\|} .
\end{aligned}$$

Then:

$$\begin{aligned}
\mathbb{E}\left[\zeta\left(g_i, s\right)^2\right] & =\mathbb{E}\left[\zeta\left(g_i, s\right)\right]^2+\mathbb{V}\left[\zeta\left(g_i, s\right)\right] \\
& =\frac{\left|g_i\right|^2}{\|\mathbf{g}\|^2}+\frac{1}{s^2} p(1-p) \\
& \leq \frac{\left|g_i\right|^2}{\|\mathbf{g}\|^2}+\frac{1}{4 s^2} .
\end{aligned}$$

Considering that $Q_s\left(g_i\right)=\|\mathbf{g}\| \cdot \operatorname{sgn}\left(g_i\right) \cdot \zeta\left(g_i, s\right)$, we have:

$$\begin{aligned}
\mathbb{E}\left[\left\|Q_b[\mathbf{g}]\right\|^2\right] & =\sum_{i=0}^d \mathbb{E}\left[\|\mathbf{g}\|^2 \zeta\left(g_i, s\right)^2\right] \\
& \leq \sum_{i=0}^d\|\mathbf{g}\|^2\left(\frac{\left|g_i\right|^2}{\|\mathbf{g}\|^2}+\frac{1}{4 s^2}\right) \\
& =\|\mathbf{g}\|^2+\frac{d}{4 s^2}\|\mathbf{g}\|^2 .
\end{aligned}$$

We can get:

$$\begin{aligned}
\mathbb{E}\left[Q_b[\mathbf{g}]\right] & =\mathbf{g}, \\
\mathbb{E}\left[\left\|Q_b[\mathbf{g}]\right\|^2\right] & \leq\left[1+\frac{d}{4^b}\right]\|\mathbf{g}\|^2 .
\end{aligned}$$

In the proof presented in [1], the sparsifier is set as the Rand-k sparsifier with an amplification factor of d/k. Here, we generalize it to a general unbiased sparsifier. For the stochastic gradient vector $\mathbf{g}$, with a sparsification parameter of k, we have the following expression:

$$\begin{aligned}
\mathbb{E}\left[S_k(\mathbf{g})\right] & =\mathbf{g}, \\
\mathbb{E}\left[\left\|S_k(\mathbf{g})\right\|^2\right] & \leq\|\mathbf{g}\|^2 .
\end{aligned}$$

Therefore, for the general form of the Co-compressor that utilizes uniform random quantization and unbiased sparsification, we obtain:

$$E[\hat{\mathbf{g}}]=\mathbb{E}\left[Q_b\left[S_k(\mathbf{g})\right]\right]=\mathbb{E}\left[S_k(\mathbf{g})\right]=\mathbf{g}, \tag{1}$$

$$\begin{aligned}
E\left[\|\hat{\mathbf{g}}\|^2\right]=\mathbb{E}\left[\left\|Q_b\left[S_k(\mathbf{g})\right]\right\|^2\right] & \leq\left[1+\frac{k}{4^b}\right]\left\|S_k(\mathbf{g})\right\|^2 \\
& =\left[1+\frac{k}{4^b}\right]\|\mathbf{g}\|^2 .
\end{aligned} \tag{2}$$

Eq. (17) demonstrates the unbiasedness for Co-compressor and Eq. (18) provides the convergence analysis for Co-compressor.

Our JointSQ framework treats sparsity as 0-bit quantization and introduces the idea of mixed-precision quantization. We split the gradient vector $\mathbf{g}$ into several subgradients $\mathbf{g}_i$ of length k_i and quantize them with different bit-width b_i. For example, the gradient vector $\{0.1,0.2,0.3,0.4\}$ can be split into $\{0.1,0.2\}$ and $\{0.3,0.4\}$. For ease of analysis, we set the remaining positions of the subgradients to 0 to match the length of the original gradient vector. Thus, we have $\mathbf{g}=\sum_{i=1}^n \mathbf{g}_i$, where n is the number of quantization bit levels. For JointSQ, we analyze its unbiasedness:

$$E[\hat{\mathbf{g}}]=E\left[\sum_{i=1}^n \hat{\mathbf{g}}_i\right]=\sum_{i=1}^n E\left(\hat{\mathbf{g}}_i\right) .$$

Based on the unbiasedness for the general form of the Co-compressor, as shown in Eq. (17), we know that $E\left(\hat{\mathbf{g}}_i\right)=\mathbf{g}_i$. So we have:

$$E[\hat{\mathbf{g}}]=\sum_{i=1}^n \mathbf{g}_i=\mathbf{g}. \tag{3}$$

Based on the definition of the Euclidean norm (L2 norm), we have $\|\mathbf{g}\|^2=\sum_{i=1}^n\left\|\mathbf{g}_i\right\|^2$. Therefore:

$$\begin{aligned}
E\left[\|\hat{\mathbf{g}}\|^2\right] & =E\left[\sum_{i=1}^n\left\|\mathbf{g}_i\right\|^2\right] \\
& =\sum_{i=1}^n E\left[\left\|\mathbf{g}_i\right\|^2\right] .
\end{aligned}$$

Based on the convergence for the general form of the Co-compressor, as shown in Eq. (18), we obtain:

$$E\left[\|\mathbf{g}_i\|^2\right] \leqslant \left[1+\frac{k}{4^b}\right]\|\mathbf{g}_i\|^2 .$$

Therefore:

$$E\left[\|\hat{\mathbf{g}}\|^2\right] \leq \sum_{i=1}^{n}\left[1+\frac{k}{4^b}\right]\|\mathbf{g}_i\|^2 . \tag{4}$$

Eq. (19) demonstrates the unbiasedness for JointSQ and Eq. (20) provides the convergence analysis for the JointSQ.

A.2. Proof of Improved Convergence for JointSQ

To contrast with the general form of a Co-compressor, we assume that the gradient tensor $\mathbf{g}$ is compressed using a Co-compressor with sparsity parameter k and quantization bit-width b. According to Eq. (3) in the main text, the compression noise in this case is obtained as follows:

$$\begin{aligned} h(k,b) &\triangleq \frac{k}{4^b} \\ &= \frac{k-2}{4^b}\frac{\|\mathbf{g}'\|^2}{\|\mathbf{g}\|^2}+\frac{1}{4^b}\frac{\|g_1\|^2}{\|\mathbf{g}\|^2}+\frac{1}{4^b}\frac{\|g_2\|^2}{\|\mathbf{g}\|^2}. \end{aligned}$$

In this particular case, we consider a scenario where we only change the quantization bit-width of two gradient elements in the compressed gradient. We quantize one gradient element, denoted as g_1, which is originally quantized to b bits, to $b+x$ bits, where $x \in \mathbb{N}^*$, and we quantize another gradient element, denoted as g_2, to $b-x$ bits. According to Eq. (6) in the main text, the compression noise in this case can be derived as follows:

$$h'(k,b) \triangleq \frac{k-2}{4^b}\frac{\|\mathbf{g}'\|^2}{\|\mathbf{g}\|^2}+\frac{1}{4^{b+x}}\frac{\|g_1\|^2}{\|\mathbf{g}\|^2}+\frac{1}{4^{b-x}}\frac{\|g_2\|^2}{\|\mathbf{g}\|^2}.$$

The variation of compressed noise is:

$$\Delta h' = \left(\frac{1}{4^{b+x}}-\frac{1}{4^b}\right)\frac{g_1^2}{\|\mathbf{g}\|^2}+\left(\frac{1}{4^{b-x}}-\frac{1}{4^b}\right)\frac{g_2^2}{\|\mathbf{g}\|^2}.$$

By solving the inequality $\Delta h' < 0$, we obtain the following result:

$$|g_1| > 2^x\,|g_2| . \tag{5}$$

This provides a case where the convergence for JointSQ is superior to the general form of Co-compressor. In fact, this conclusion can be generalized to reducing the bit-width of multiple gradient elements to improve the bit-width of multiple gradient elements:

$$\sum_{i=1}^{n_1}\frac{4^{x_i}-1}{4^{x_i}}\,|g_i|^2 > \sum_{j=1}^{n_2}\left(4^{y_j}-1\right)|g_j|^2 .$$

where $x_1+x_2+...x_{n_1} = y_1+y_2+...y_{n_2}$. This finding demonstrates the significant contribution of JointSQ in expanding the solution space and mitigating the occurrence of suboptimal solutions in Co-compressor.

A.3. Core Algorithm of JointSQ

A.3.1 Greedy Allocation

Algorithm 1 Greedy Allocation

Input: Assignable bit-width c, gradient vector $\mathbf{g}$.
Output: Mixed-Precision quantization mask x, gradient sorting results SP'.
$Remain\_bit \leftarrow c$ // Remain backpack capacity.
$x_{i1} \leftarrow 1$ // Default Selection of 0-bit per Group.
$B \leftarrow [0,2,4,8], b_j \in B$ // Available quantization bit widths.
$\rho_{ij} \leftarrow \frac{4^{w_{ij}}-1}{4^{w_{ij}}}\frac{g_i^2}{\|\mathbf{g}\|^2}, w_{ij} \leftarrow b_j$ // Profit and weight of items.
$SP \leftarrow argsort(\frac{p_{ij}-p_{i,j-1}}{w_{ij}-w_{i,j-1}})$ // Sort by Incremental Profit Density.
$SP' \leftarrow SP$
while $Remain\_bit > 0$ **do**
  $i, j \leftarrow SP[0], x_{i,j} \leftarrow 1, x_{i,j-1} \leftarrow 0$ // Select the
  item with the highest rank.
  $SP \leftarrow Update(SP)$ // Remove the j-th item in
  i-th group from the selection pool.
  $Remain\_bit \leftarrow Remain\_bit - w_{i,j}$
end while
return x, SP'

A.3.2 Reallocation

Algorithm 2 Reallocation

Input: Learnable Parameter R, assignable bit-width c, Mixed-Precision quantization mask x, gradient vector $\mathbf{g}$, sorting results in Greedy Allocation SP.
Output: Mixed-Precision quantization Mask x, reduction of compression noise h.
$\bar{k} \leftarrow \frac{Rc}{8}$ // Constraint Value $\bar{k}$ for Length.
$k \leftarrow \sum_{i=1}^{d}\sum_{j=2}^{4} x_{ij}$ // Get k from the Last Reallocation.
$h \leftarrow 0,\ f \leftarrow 0$ // Initialize the compression noise reduction amount h and the fine-tuning flag f.
$\Delta k = k - \bar{k}$ // Retrieve the number of fine-tuning iterations.
for $i = 1$ to $|\Delta k|$ **do**
  $x', f \leftarrow finetuning(x, SP, \Delta k, f)$ // Fine-tuning the mask according to the rules mentioned in the main text.
  $\Delta h = h(g, x') - h(g, x)$ // Calculate the difference in compression noise before and after fine-tuning.
  if $\Delta h < 0$ **then**
    $x \leftarrow x'$ // Keep only the fine-tuning attempts
    that result in a reduction of compression noise.
    $h \leftarrow h + \Delta h$ // Update the reduction of
    compression noise.
  end if
end for
return x, h

Algorithm 3 Fine-tuning

Input: Mixed-Precision quantization mask x, sorting results in Greedy Allocation SP, difference between the constraint length and the current length Δk, fine-tuning flag f.

Output: Mixed-Precision quantization mask x.

if $\Delta k < 0$ **then**
  if $f = 0$ **then**
    $x_{i_1,j_4} \leftarrow 0, x_{i_1,j_3} \leftarrow 1$
    $x_{i_2,j_2} \leftarrow 0, x_{i_2,j_3} \leftarrow 1$
    $x_{i_3,j_1} \leftarrow 0, x_{i_3,j_2} \leftarrow 1$
    $f \leftarrow 1$
  // i_1 represents the least ranked 8-bit gradient, i_2 represents the highest ranked 2-bit gradient, and i_3 represents the highest ranked 0-bit gradient.
  else
    $x_{i_1,j_3} \leftarrow 0, x_{i_1,j_2} \leftarrow 1$
    $x_{i_2,j_1} \leftarrow 0, x_{i_2,j_2} \leftarrow 1$
    $f \leftarrow 0$
  // i_1 represents the least ranked 4-bit gradient, i_2 represents the highest ranked 0-bit gradient.
  end if
else
  if $f = 0$ **then**
    $x_{i_1,j_3} \leftarrow 0, x_{i_1,j_4} \leftarrow 1$
    $x_{i_2,j_3} \leftarrow 0, x_{i_2,j_2} \leftarrow 1$
    $x_{i_3,j_2} \leftarrow 0, x_{i_3,j_1} \leftarrow 1$
    $f \leftarrow 1$
  // i_1 represents the highest ranked 4-bit gradient, i_2 represents the least ranked 4-bit gradient, and i_3 represents the least ranked 2-bit gradient.
  else
    $x_{i_1,j_2} \leftarrow 0, x_{i_2,j_3} \leftarrow 1$
    $x_{i_2,j_2} \leftarrow 0, x_{i_2,j_1} \leftarrow 1$
    $f \leftarrow 0$
  // i_1 represents the highest ranked 2-bit gradient, i_2 represents the least ranked 2-bit gradient.
  end if
end if
return x, f

A.3.3 JointSQ in Distributed Learning

Algorithm 4 JointSQ in Distributed Learning

Input: The gradients for the current iteration of training $\mathbf{g}$, the compression ratio C, the number of Reallocation performed T, the number of nodes in distributed training N, the initial value R_0, the learning rate of R δ_h.

Output: The compressed gradients $\hat{\mathbf{g}}$.

On each node:
for each layer's gradient vector $\mathbf{g}_l$ in $\mathbf{g}$ **do**
  $c \leftarrow 32 * len\,(\mathbf{g}_l) * C$
  $x, SP \leftarrow GreedyAllocation(c, \mathbf{g}_l)$
  for $i = 1$ to T **do**
    $x_i, h_i \leftarrow Reallocation(R_{i-1}, c, x_{i-1}, \mathbf{g}_l, SP)$
    $R_i \leftarrow R_{i-1} + \delta_h(h_i - h_{i-1})$.
  end for
  $\hat{\mathbf{g}}_l \leftarrow Quantize(\mathbf{g}_l, x)$
end for
All-reduce: $\hat{\mathbf{g}} \leftarrow \sum_{i=1}^{N} \hat{\mathbf{g}}$
return $\hat{\mathbf{g}}$

References

[1] Guangfeng Yan, Tan Li, Shao-Lun Huang, Tian Lan, and Linqi Song. AC-SGD: Adaptively compressed SGD for communication-efficient distributed learning. *IEEE Journal on Selected Areas in Communications*, 40(9):2678–2693, 2022. 1