Implementation of $\mathsf{SecJudge}(\llbracket g_i \rrbracket)$

For an encrypted local gradient $\llbracket g_i \rrbracket = \{\llbracket x'_1 \rrbracket, \llbracket x'_2 \rrbracket, ..., \llbracket x'_m \rrbracket\}$ submitted to $\mathcal{S}_1$ ($i \in [1, n]$):

- @$\mathcal{S}_1$: To protect data privacy, $\mathcal{S}_1$ first chooses m random noises $r_{k\in[1,m]} \leftarrow \mathbb{Z}_N^*$, and blinds r_k to $\llbracket x'_k \rrbracket$ using Eq. 5 such that
$$\llbracket \overline{x}_k \rrbracket = \llbracket x'_k \rrbracket \cdot \llbracket r_k \rrbracket = \llbracket x'_k + r_k \rrbracket. \tag{14}$$
Then, $\mathcal{S}_1$ implements the partial decryption by calling $[\overline{x}_k]_1 \leftarrow \mathsf{PartDec}_{sk_1}(\llbracket \overline{x}_k \rrbracket)$, and sends $\{\llbracket \overline{x}_1 \rrbracket, ..., \llbracket \overline{x}_m \rrbracket\}$ and $\{[\overline{x}_1]_1, ..., [\overline{x}_m]_1\}$ to $\mathcal{S}_2$ ($k \in [1, m]$).
- @$\mathcal{S}_2$: To decrypt the blinded number $\overline{x}_k$, $\mathcal{S}_2$ calls $[\overline{x}_k]_2 \leftarrow \mathsf{PartDec}_{sk_2}(\llbracket \overline{x}_k \rrbracket)$ and $\overline{x}_k \leftarrow \mathsf{FullDec}([\overline{x}_k]_1, [\overline{x}_k]_2)$. Then, $\mathcal{S}_2$ returns $\llbracket \Sigma \overline{x}_k{}^2 \rrbracket$ to $\mathcal{S}_1$ ($k \in [1, m]$) such that $\llbracket \Sigma \overline{x}_k{}^2 \rrbracket \leftarrow \mathsf{Enc}_{pk}(\overline{x}_1{}^2 +, ..., + \overline{x}_m{}^2)$.
- @$\mathcal{S}_1$: To remove the random noises $r_{k\in[1,m]}$, $\mathcal{S}_1$ implements
$$\begin{aligned} \llbracket sum \rrbracket &= \llbracket {x'}_1^2 + ... + {x'}_m^2 \rrbracket \\ &= \llbracket \Sigma(x'_k + r_k)^2 \rrbracket \cdot (\llbracket x'_1 \rrbracket^{2r_1} \cdot, ..., \cdot \llbracket x'_m \rrbracket^{2r_m} \cdot \llbracket \Sigma r_k^2 \rrbracket)^{N-1} \\ &= \llbracket \Sigma \overline{x}_k{}^2 \rrbracket \cdot \llbracket \Sigma(2r_k x_k + r_k^2)) \rrbracket^{N-1}, \end{aligned} \tag{15}$$
and sends $\llbracket sum \rrbracket$ to $\mathcal{S}_2$.
- @$\mathcal{S}_2$: $[sum]_2 \leftarrow \mathsf{PartDec}_{sk_2}(\llbracket sum \rrbracket)$ is executed to return $\mathcal{S}_1$.
- @$\mathcal{S}_1$: To obtain the final result sum, $\mathcal{S}_1$ uses $\mathsf{PartDec}$ and $\mathsf{FullDec}$. Note that the precise degree of sum is expanded as deg^2 after $\mathsf{HE.Mul}$. If $sum/deg^2 = 1$, it means the local gradient g_i is normalized. $\mathcal{S}_1$ accepts g_i for further training. Otherwise, g_i is aborted without gradients normalization as
$$g_i = \begin{cases} \text{Accept,} & \text{If } sum/deg^2 = 1, \\ \text{Abort,} & \text{Otherwise.} \end{cases}$$