APPENDIX
PROOF OF THE NP-HARDNESS OF P(1) IN OFFLINE SETTING

In Problem (1), we let e_{ihk} denote the throughput of job i under number configuration h for GPU type k, rather than expressing it as the product of the number of GPUs used and the per-GPU throughput. This is because, in distributed training, the throughput is nonlinear in relation to the number of GPUs due to communication overhead. Let us consider a simplified problem where the throughput scales linearly with the number of GPUs, and each job is restricted to use only one GPU type throughout its training. Under this assumption, the formulation of simplified Problem (1) is as follows.

$$\text{(P7):} \quad \text{maximize} \sum_i \alpha_i \tag{7}$$

$$\text{s.t.} \sum_{t=arr_i}^{ddl_i} \sum_k x_{it}^k e_{ik} \geq \alpha_i E_i, \forall i \tag{7a}$$

$$x_{it}^k \leq s_i y_{ik}, \forall i, k, t \tag{7b}$$

$$\sum_k y_{ik} = \alpha_i, \forall i \tag{7c}$$

$$\sum_i x_{it}^k \leq C^k, \forall k, t \tag{7d}$$

In this formulation, binary variables α_i denote whether job i is accepted or not. Integer variables x_{it}^k represent the number of type-k GPUs allocated to job i in timeslot t, and binary variables y_{ik} indicate whether job i uses type-k GPUs. Let e_{ik} denote the number of iterations that job i can complete with one type-k GPU in a timeslot.

Next, we show that the bin packing problem can be reduced to the simplified problem, thereby demonstrating that the simplified problem is NP-hard. The bin packing problem is well-known to be NP-complete [1] and can be formulated as follows.

$$\text{maximize} \quad \sum_i z_i$$

$$\text{s.t.} \sum_j \beta_{ij} = z_i, \forall i$$

$$\sum_j w_i \beta_{ij} \leq C, \forall j$$

Binary variables z_i denote whether item i is placed into bins and binary variables β_{ij} denote whether item i is placed into bin j. Constant w_i and C denote the weight of item i and the capacity of a bin, respectively.

In P(7), we can set $\forall i, arr_i = 0, ddl_i = 1, s_i = E_i = w_i$, $\forall i, k, e_{ik} = 1$, $\forall k, C^k = C$. Then binary variables x_{it}^k can be replaced by $w_i y_{ik}$. Based on this, constraints (7a) and (7b) can be removed. The remaining two constraints (7c) and (7d) are consistent with those of the bin packing problem. This mapping can be completed in polynomial time, hence P(7) is NP-hard. Since P(7) is no harder than P(1), we can conclude that P(1) is NP-hard even in offline setting.

REFERENCES

[1] J. T. Leung and D. Ting, "Bin packing: Maximizing the number of pieces packed," *Acta Informatica*, vol. 9, pp. 263–271, 1978.