# 1 Problem statement

## 1.1 Version 1

Given a set of tasks $\mathcal{T}$, a set of resources $\mathcal{R}$, GPU capacity $c^{(GPU)}$, major frame length $h$ and characteristics of each task $T_i \in \mathcal{T}$, the goal is to find a feasible schedule for tasks to be run on available resources which minimizes energy consumption. The process of finding such a schedule consists of two phases:

**Phase 1:** assigning tasks to resources.

**Phase 2:** scheduling tasks into isolation windows.

### 1.1.1 Phase 1

**Input:**

- Set $\mathcal{R} = (R_1, R_2, ..., R_m)$ representing available resources, where each resource $R_k \in \mathcal{R}$ has capacity $c_k$.
- Value $c^{(GPU)}$ representing GPU capacity.
- Set of tasks $\mathcal{T} = (T_1, T_2, ..., T_n)$.
- Values $p_{i,k}$ and $E_{i,k}$ representing processing time and energy consumption of task $T_i$ on resource $R_k$. If task $T_i$ cannot be executed on resource $R_k$, then $p_{i,k} = \infty$.
- Values $p_{i,k}^{(GPU)}$ and $E_{i,k}^{(GPU)}$ representing processing time and energy consumption of task $T_i$ on resource $R_k$ if the task is offloaded to GPU. If task $T_i$ cannot be executed on resource $R_k$, or cannot be offloaded to GPU, then $p_{i,k}^{(GPU)} = \infty$.
- Value $h$ representing major frame length.

**Output:** The output consists of resource assignment function $a_r : \mathcal{T} \to \mathcal{R}$ and GPU offloading function $a_g : \mathcal{T} \to \{0, 1\}$.

**ILP model:**

$$\min \sum_{T_i \in \mathcal{T}} \sum_{R_k \in \mathcal{R}} (a_{i,k} \cdot E_{i,k} + a_{i,k}^{(GPU)} \cdot E_{i,k}^{(GPU)}) \quad \text{subject to:} \tag{1}$$

$$\sum_{R_k \in \mathcal{R}} (a_{i,k} + a_{i,k}^{(GPU)}) = 1 \quad \forall T_i \in \mathcal{T} \tag{2}$$

$$\sum_{T_i \in \mathcal{T}} (a_{i,k} \cdot p_{i,k} + a_{i,k}^{(GPU)} \cdot p_{i,k}^{(GPU)}) \leq h \cdot c_k \quad \forall R_k \in \mathcal{R} \tag{3}$$

$$\sum_{T_i \in \mathcal{T}} \sum_{R_k \in \mathcal{R}} a_{i,k}^{(GPU)} \cdot p_{i,k}^{(GPU)} \leq h \cdot c^{(GPU)} \tag{4}$$

$$a_{i,k} = 0 \quad \forall T_i \in \mathcal{T}, \forall R_k \in \mathcal{R} : p_{i,k} = \infty \tag{5}$$

$$a_{i,k}^{(GPU)} = 0 \quad \forall T_i \in \mathcal{T}, \forall R_k \in \mathcal{R} : p_{i,k}^{(GPU)} = \infty \tag{6}$$

$$a_{i,k} \in \{0,1\}, a_{i,k}^{(GPU)} \in \{0,1\} \quad \forall T_i \in \mathcal{T}, \forall R_k \in \mathcal{R} \tag{7}$$

Note that a feasible solution does not imply that it will be possible to find a feasible schedule in phase 2.

### 1.1.2 Phase 2

**Input:**

- Set $\mathcal{R} = (R_1, R_2, ..., R_m)$ representing available resources, where each resource $R_k \in \mathcal{R}$ has capacity $c_k$.
- Value $c^{(GPU)}$ representing GPU capacity.
- Set $\mathcal{T} = (T_1, T_2, ..., T_n)$ representing tasks to be scheduled. Each task $T_i \in \mathcal{T}$ has processing time $p_i$.
- Resource assignment function $a_r : \mathcal{T} \to \mathcal{R}$ and GPU offloading function $a_g : \mathcal{T} \to \{0,1\}$.
- Value $h$ representing major frame length.

**Output:** Let $\mathcal{W} = (W_1, W_2, ..., W_\ell)$ be a set of isolation windows, where each window $W_j \in \mathcal{W}$ has length $l_j$. The goal is to find a triplet $S = (\ell, \boldsymbol{l}, a_w)$, where $\ell$ is the number of windows, $\boldsymbol{l}$ is the vector capturing the windows' lengths, and $a_w : \mathcal{T} \to \mathcal{W}$ is the window assignment function. Solution $S$ is feasible, if the following constraints are satisfied:

$$l_j \geq \frac{\max_{T_i \in \mathcal{T} : a_w(T_i) = W_j} \{p_i\}}{0.6} \quad \forall W_j \in \mathcal{W} \tag{8}$$

$$\sum_{W_j \in \mathcal{W}} l_j \leq h \tag{9}$$

$$\sum_{T_i \in \mathcal{T} : a_w(T_i) = W_j} \mathbb{1}_{[a_r(T_i) = R_k]} \leq c_k \quad \forall R_k \in \mathcal{R}, \forall W_j \in \mathcal{W} \tag{10}$$

$$\sum_{T_i \in \mathcal{T} : a_w(T_i) = W_j} a_g(T_i) \leq c^{(GPU)} \quad \forall W_j \in \mathcal{W} \tag{11}$$

Solution $S$ is optimal, if it minimizes the total length of the schedule $\sum_{W_j \in \mathcal{W}} l_j$.
**ILP model:**

$$\min \sum_{W_j \in \mathcal{W}} l_j \quad \text{subject to:} \tag{12}$$

$$\sum_{W_j \in \mathcal{W}} l_j \leq h \tag{13}$$

$$l_j \geq \frac{w_{i,j} \cdot p_i}{0.6} \quad \forall T_i \in \mathcal{T}, \forall W_j \in \mathcal{W} \tag{14}$$

$$\sum_{W_j \in \mathcal{W}} w_{i,j} = 1 \quad \forall T_i \in \mathcal{T} \tag{15}$$

$$\sum_{T_i \in \mathcal{T} : a_r(T_i) = R_k} w_{i,j} \leq c_k \quad \forall W_j \in \mathcal{W}, \forall R_k \in \mathcal{R} \tag{16}$$

$$\sum_{T_i \in \mathcal{T}} w_{i,j} \cdot a_g(T_i) \leq g \quad \forall W_j \in \mathcal{W} \tag{17}$$

$$l_j \in \mathbb{N} \quad \forall W_j \in \mathcal{W} \tag{18}$$

$$w_{i,j} \in \{0, 1\} \quad \forall T_i \in \mathcal{T}, \forall W_j \in \mathcal{W} \tag{19}$$