Algorithm 1: Preemption-Aware Scheduling Policy (PAP).

Input: $\bar{\Lambda}_j$,θ_j,ω_j,λ_j,τ_j,μ_j, for all $1 \leq j \leq N$.
Output: $(\hat{\Lambda}_j)$ load distribution of grid requests to different clusters, for all $1 \leq j \leq N$.

for $j \leftarrow 1$ **to** N **do**
  $\psi_j = \frac{\lambda_j \mu_j}{2(1-\rho_j)^2} + \frac{\theta_j}{(1-\rho_j)}$;
`Sort` (ψ);
$k \leftarrow 1$;
while $k < N$ **do**
  if $\sum_{j=1}^{k} \phi_j(\psi_k) \geq \left(\sum_{j=1}^{k} \frac{(1-\rho_j)}{\theta_j} \right) - \Lambda$ **then**
    break;
  else
    $k \leftarrow k + 1$;
$lb \leftarrow \psi_k$;
$ub = 2 * lb$;
while $\sum_{j=1}^{k} \phi_j(ub) > \left(\sum_{j=1}^{k} \frac{(1-\rho_j)}{\theta_j} \right) - \Lambda$ **do**
  $ub = 2 * ub$;
while $ub - lb > \epsilon$ **do**
  $z \leftarrow (lb + ub)/2$;
  if $\sum_{j=1}^{k} \phi_j(z) \geq \left(\sum_{j=1}^{k} \frac{(1-\rho_j)}{\theta_j} \right) - \Lambda$ **then**
    $lb \leftarrow z$;
  else
    $ub \leftarrow z$;
for $j \leftarrow 1$ **to** k **do**
  $\hat{\Lambda}_j = \frac{(1-\rho_j)}{\theta_j} - \frac{1}{\theta_j} \sqrt{\frac{(1-\rho_j)(\omega_j(1-\rho_j)) + \theta_j \lambda_j \mu_j}{2\theta_j(1-\rho_j)z + (\omega_j - 2\theta_j^2)}}$;
for $j \leftarrow k+1$ **to** N **do**
  $\hat{\Lambda}_j = 0$;