Algorithm 1: Feasibility Check

Input: $M_{(i,p)}$: The number of modules in panel p at current i;
P: Number of panels; S: Number of strings;
Q_M: The minimum number of modules for each string;
Output: Feasibility Result;
Conf: Configuration Result;

for *k = 1 to S-1,* **each** panel p **do**
  Loss[p,k] = $M_{(i_{S-k},p)}$ – $M_{(i_{S-k+1},p)}$
end
PSet = $\{p_1, p_2, ..., p_P\}$;
for *j = S to 1* **do**
  n = 1;
  for each $p \in PSet$ **do**
    $Loss^*[p] = (Loss[p, S-j+1], Loss[p, S-j+2], ..., Loss[P, S])$
  end
  Let $\alpha_1, \alpha_2, ..., \alpha_{|PSet|}$ be indices such that $Loss^*[p_{\alpha i}] \leq$ Loss$^*[p_{\alpha j}]$ holds for **any** i ≤ j
  `/* Sort Panels in lexicographical permutation by` *Loss* `*/`
  while SUM = $\sum_{k=1}^{n} M_{(i_j,\alpha_k)} < Q_M$ **do**
    if $n == PSet$ **then**
      return: Feasibility = NO
    end
    n ++;
  end
  TempConf = { p_{α_1}, p_{α_2}, p_{α_3},..., p_{α_n} };
  for *m = 1 to n* **do**
    if $M_{(j,\alpha_m)} == 0$ **then**
      TempConf = TempConf - $\{p_{\alpha_m}\}$;
    end
  end
  if SUM == Q_M or $j == 1$ **then**
    Conf [j] = TempConf
  else
    Do Panel Swap
  end
  PSet = PSet - Conf[j]
end
return: Feasibility = YES

Algorithm 2: Panel Swap

Input: $M_{(j,\alpha)}$, Q_M, SUM, TempConf
Output: Conf [j]

Over = SUM - Q_M ;
for p_{a_x} *in* {TempConf} **do**
  if $M_{(j,\alpha_x)} == Over$ **then**
    TempConf = TempConf - $\{p_{\alpha_x}\}$
    return: Conf [j] = TempConf;
  else if $\sum_x^{x+2} M_{(i_j,\alpha_x)} == 3 == M_{(j,\alpha_y)}$ **and** $M_{(j,\alpha_x)} == M_{(j,\alpha_{x+1})} == M_{(j,\alpha_{x+2})}$, $y > n$ **then**
    TempConf =TempConf - $\{p_{\alpha_x}, p_{\alpha_{x+1}}, p_{\alpha_{x+2}}\}+\{p_{\alpha_y}\}$;
  else if $\sum_x^{x+2} M_{(i_j,\alpha_x)} == 3 == M_{(j,\alpha_y)}$ $y > n$ **then**
    TempConf =TempConf - $\{p_{\alpha_x}, p_{\alpha_{x+1}}, p_{\alpha_{x+2}}\}+\{p_{\alpha_y}\}$;
  else if $\sum_x^{x+2} M_{(i_j,\alpha_x)} == 2 == M_{(j,\alpha_y)}$ $y > n$ **then**
    TempConf =TempConf - $\{p_{\alpha_x}, p_{\alpha_{x+1}}\}+\{p_{\alpha_y}\}$;
  else
    Conf [j] = TempConf ;
  end
end

Algorithm 3: find extra modules

Input:
Output: