Algorithm: Enforce_Role_Usage_and_Permission_Distribution_Constraints

1: Required: MRC_{user}, MRC_{perm}, UPA Matrix
2: Set $UserRoleCount[u] = 0$, $\forall u \in U$
3: Set $PermRoleCount[p] = 0$, $\forall p \in P$
4: U represents the set of selected users and P represents the set of selected permissions to form a role
{PHASE 1}
5: **for** each user u with uncovered incident edges AND $UserRoleCount[u] < MRC_{user}$ - 1 OR permission p with uncovered incident edges AND $PermRoleCount[p] < MRC_{perm}$ - 1 **do**
6: Set $U = \phi$, $P = \phi$
7: Select a vertex v using a suitanle heuristic
8: **if** the selected vertex is a user **then**
9: Call $Form\_Role$ procedure (Algorithm 5)
10: **else**
11: Call $Dual\ of\ Form\_Role$ procedure
12: **end if**
13: **end for**
{PHASE 2}
14: **for** each user u with uncovered incident edges AND $UserRoleCount[u] = MRC_{user}$ - 1 OR permission p with uncovered incident edges AND $PermRoleCount[p] = MRC_{perm}$ - 1 **do**
15: Set $U = \phi$, $P = \phi$
16: Select the vertex v with the maximum number of uncovered incident edges
17: **if** the selected vertex is a user **then**
18: Set $P = UC[v]$
19: **if** $PermRoleCount[p] \leq MRC_{perm}$ - 1, $\forall p \in P$ **then**
20: Call $Form\_Role$ procedure (Algorithm 5)
21: **end if**
22: **else**
23: Set $U = UC[v]$
24: **if** $UserRoleCount[u] \leq MRC_{user}$ - 1, $\forall u \in U$ **then**
25: Call $Dual\ of\ Form\_Role$ procedure
26: **end if**
27: **end if**
28: **end for**
29: **if** there is at least one vertex with uncovered edges **then**
30: "The given set of constraints cannot be enforced"
31: **end if**