1 Incremental Algorithm

Definition 1 Let $X = [(\mathbf{a}_1, b_1), \ldots, (\mathbf{a}_N, b_N)]$, where $\mathbf{a}_i \in \mathbb{R}^n, b_i \in \mathbb{R}$ for $i = 1, \ldots, N$. Let $V = \{i_1, \ldots, i_s\} \subseteq \{1, \ldots, n\}$ for some $1 \leq i_1 < i_2 < \cdots < i_s \leq n$, and define the projection

$$\pi_V : \mathbb{R}^n \longrightarrow \mathbb{R}^s, \quad \pi_V(x_1, \ldots, x_n) = (x_{i_1}, \ldots, x_{i_s}).$$

We say that the set V is *separating* for X if for all $1 \leq i, j \leq N$

$$b_i \neq b_j \quad \Rightarrow \quad \pi_V(\mathbf{a}_i) \neq \pi_V(\mathbf{a}_j).$$

We say that V is a *minimal separating set* for X if V is separating but no proper subset of V is separating for X.

Algorithm 2 MinSepSet (incremental)
INPUT: $X = [(\mathbf{a}_1, b_1), \ldots, (\mathbf{a}_N, b_N)]$, where $\mathbf{a}_i \in \mathbb{R}^n, b_i \in \mathbb{R}$ for $i = 1, \ldots, N$.
OUTPUT: $\mathcal{S} = [V_1, \ldots, V_t]$ such that the entries of $\mathcal{S}$ form the set of all minimal separating sets for X. Also, $|V_i| \leq |V_j|$ if $i < j$.

```
if N = 1 then
  return [∅]
end if
S := [ ]
T := MinSepSet([(a_1, b_1), ..., (a_{N-1}, b_{N-1})])
for i=1 to |T| do
  V := T[i]
  G_V := ∅
  for j=1 to N − 1 do
    if b_j ≠ b_N and π_V(a_j) = π_V(a_N) then
      G_V := G_V ∪ {j}
    end if
  end for
  if G_V = ∅ then
    S := Insert&Reduce(S, V)
  end if
  if G_V = {j_1} then
    for k=1 to n do
      if π_{k}(a_{j_1}) ≠ π_{k}(a_N) then
        S := Insert&Reduce(S, V ∪ {k})
      end if
    end for
  end if
  if G_V = {j_1, ..., j_s} and s ≥ 2 then
    V̄ := {1, ..., n} − V
```

$\mathcal{P} :=$ MinSepSet$([(\pi_{\bar{V}}(\mathbf{a}_N), b_N), (\pi_{\bar{V}}(\mathbf{a}_{j_1}), b_{j_1}), \ldots, (\pi_{\bar{V}}(\mathbf{a}_{j_s}), b_{j_1})])$
for j=1 to $|\mathcal{P}|$ **do**
$\mathcal{S} :=$ Insert&Reduce$(\mathcal{S}, V \cup \mathcal{P}[i])$
end for
end if
end for
return $\mathcal{S}$

Definition 3 Let $\mathcal{S} = [S_1, \ldots, S_t]$ be a list of finite sets. We say that $\mathcal{S}$ is *reduced* if for all $i < j$ we have $|S_i| \leq |S_j|$ and $S_i \nsubseteq S_j$.
For $\mathcal{S}$ as above, V a finite set and $i \in \{1, \ldots, t, t+1\}$, the subroutine Insert$(\mathcal{S}, V, i)$ inserts V to the i-th place of $\mathcal{S}$, moving the $S_i, \ldots, S_t$ one to the right. For $i = t+1$ it inserts V to the end of the list $\mathcal{S}$.
The subroutine Remove$(\mathcal{S}, S_i)$ finds and removes S_i form $\mathcal{S}$, keeping the order of the rest of the list.

Algorithm 4 Insert&Reduce
INPUT: $\mathcal{S} = [S_1, \ldots, S_t]$ a reduced list of finite sets, and V a finite set.
OUTPUT: If $\exists\, S_i$ such that $S_i \subseteq V$ then return $\mathcal{S}$, otherwise the elements of $\{V\} \cup \{S_i \,:\, V \nsubseteq S_i\}$ ordered by their cardinality.

if $|S| = 0$ **then**
return $[V]$
end if
flag $:= 0$
$j := 1$
for i=1 **to** $|S|$ **do**
if $S_i \subseteq V$ **then**
return $\mathcal{S}$
end if
if flag $= 0$ **and** $|S_i| \geq |V|$ **then**
$\mathcal{S}'_j := V$
flag $:= 1$
end if
if flag $= 1$ **and** $V \nsubseteq S_i$ **then**
$\mathcal{S}'_j := S_i$
end if
$j := j + 1$
end for
if flag $= 0$ **then**
$\mathcal{S}'_{j+1} := V$
end if
return $\mathcal{S}'$