Algorithm 2.1 SIMPLEX

Input. A Matrix $A \in \mathbb{R}^{m\times n}$, vectors $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and a vertex x of $P = \{x \in \mathbb{R}^n : Ax \leq b\}$.

Output. A vertex x of P attaining $\max\{c^\top x : x \in P\}$ or a vector $w \in \mathbb{R}^n$ with $Aw \leq 0$ and $c^\top w > 0$ (i.e., the LP is unbounded).

Step 1. Choose a set of n row indices I such that A_I is non-singular and $A_I x = b_I$.

Step 2. Compute $c^\top (A_I)^{-1}$ and add zeros in order to obtain a vector $y \in \mathbb{R}^m$ with $c^\top = y^\top A$ such that all entries outside I are zero. If $y \geq 0$ then return x (and y).

Step 3. Choose the minimum index i with $y_i < 0$. Let w be the column of $-(A_I)^{-1}$ with index i. Thus $A_{I-\{i\}}w = 0$ and $a_i w = -1$. If $Aw \leq 0$ then return w.

Step 4. Let

$$\lambda := \min\left\{\frac{b_j - a_j x}{a_j w} : a_j w > 0\ j = 1, \ldots, m,\right\},$$

and let j be the smallest row index attaining this minimum.

Step 5. Set $I := (I - \{i\}) \cup \{j\}$ and $x := x + \lambda w$. Go to Step 2.