Bounded Preemptive Simplex (Lexicographic Goal Programming Core)

How bounds are kept in the tableau, how LB/UB/B statuses work, how entering is chosen, and how bound hits (no pivot) differ from row pivots.

1. Problem structure (generic)

Variables $x_1, \dots, x_n$ have true physical bounds

$$\ell_i \;\le\; x_i \;\le\; u_i \qquad (i = 1, \dots, n).$$

If a variable is nonbasic, it sits exactly at one bound:

$$x_i = \ell_i \text{ (LB)} \quad \text{or} \quad x_i = u_i \text{ (UB)}.$$

If a variable is basic (B), it can lie strictly between its bounds:

$$\ell_i < x_i < u_i.$$

Bounds are kept *inside* the simplex (no substitutions like $x = \ell + \tilde{x}$).

2. Generic bounded simplex tableau

A canonical tableau (illustrative):

Basic	RHS	x_1	x_2	x_3	x_4	Status
x_3	2	$+1$	-2	1	0	B
x_4	1	0	$+1$	-1	1	B
z (max)	7	$+3$	-1	0	0	

Separately store nonbasic variables and their bound statuses:

Variable	Value	Status
x_1	$x_1 = \ell_1$	LB
x_2	$x_2 = u_2$	UB

In the bounded simplex, nonbasics sit at their *real* bounds (not at 0).

3. Reduced-cost eligibility (choosing the entering variable)

For a *maximization*:

$$\begin{aligned} \text{Nonbasic at LB}: &\quad \text{may enter by increasing if } r_j > 0. \\ \text{Nonbasic at UB}: &\quad \text{may enter by decreasing if } r_j < 0. \end{aligned}$$

For a *minimization*, flip signs:

$$\begin{aligned} \text{Nonbasic at LB}: &\quad \text{may enter if } r_j < 0. \\ \text{Nonbasic at UB}: &\quad \text{may enter if } r_j > 0. \end{aligned}$$

If no nonbasic satisfies this, the current stage is optimal.

4. Augmented ratio test (bounded feature)

Let x_e be eligible. Define the attempted move direction:

$$d_e = \begin{cases} +1, & \text{if } x_e \text{ is at LB (increase)}, \\ -1, & \text{if } x_e \text{ is at UB (decrease)}. \end{cases}$$

Let $\boldsymbol{d} = B^{-1}A_{:e}$ be the basic-direction column for x_e. Movement with step Δ is

$$\Delta x_e = d_e\,\Delta, \qquad \Delta \boldsymbol{x}_B = -\boldsymbol{d}\,\Delta.$$

Two limits can stop us:

(A) Row limit: a basic hits LB or UB. For each basic x_{B_i} with direction d_i:

$$\theta_i = \begin{cases} \dfrac{x_{B_i} - \ell_{B_i}}{d_i}, & \text{if } d_e = +1,\ d_i > 0, \\ \dfrac{u_{B_i} - x_{B_i}}{-d_i}, & \text{if } d_e = +1,\ d_i < 0. \end{cases}$$

If $d_e = -1$, LB/UB swap.

(B) Bound limit of the entering variable.

$$\theta_{\text{bound}} = \begin{cases} u_e - x_e, & \text{if } d_e = +1, \\ x_e - \ell_e, & \text{if } d_e = -1. \end{cases}$$

Step size.

$$\theta = \min\{\theta_{\text{bound}},\ \min_i \theta_i\}.$$

5. Two possible outcomes

(1) Strict bound hit of entering ($\theta = \theta_{\textbf{bound}}$)

- **Preemptive bound hit:** no pivot.
- Move x_e to the opposite bound, update basics: $\Delta \boldsymbol{x}_B = -\boldsymbol{d}\,\theta$.
- x_e stays nonbasic, flips LB $\leftrightarrow$ UB.

(2) Row pivot ($\theta = \theta_i$)

- Perform a Gauss–Jordan pivot.
- Entering becomes basic; leaving hits its bound and becomes LB or UB.

6. Example iteration

Initial:

var	value	status
x_1	0	LB
x_2	5	UB
x_3	2	B
x_4	1	B

Maximization reduced costs:

$$r_1 = +3 > 0 \Rightarrow x_1 \text{ eligible from LB}, \qquad r_2 = -1 < 0 \Rightarrow x_2 \text{ eligible from UB}.$$

Pick x_1. Suppose $\theta_{\text{bound}} = 10$ and row x_3 reaches LB at $\theta = 2$. Since $2 < 10$, pivot.

Basic	RHS	x_1	x_2	x_3	x_4	Status
x_1	2	1	-2	1	0	B
x_4	1	0	$+1$	-1	1	B
z	13	0	$+5$	-3	0	

Leaving x_3 hits LB, so $x_3 \to$ LB.

If instead $\theta_{\text{bound}} = 1 < 3$, then preemptive bound hit. x_1 moves to u_1, no pivot, status flips to UB.

7. Degeneracy

If $\theta = 0$:

- If row-limited: pivot.
- If bound-limited: flip LB $\leftrightarrow$ UB.
- Use Bland's rule to avoid cycling.

8. Difference from ordinary simplex

Standard simplex	Bounded preemptive simplex
Nonbasics at 0	Nonbasics at LB/UB
No opposite-bound hits	LB$\to$UB or UB$\to$LB possible
Movement ends in pivot	May end in bound hit with no pivot
Needs variable substitution for bounds	Bounds handled directly

9. Summary

Event	Stops movement?	Action	Status change
Bound hit (strict)	Opposite bound first	No pivot	LB $\leftrightarrow$ UB
Row pivot	Basic hits bound first (or tie)	Pivot	Enter $\to$ B, leave $\to$ LB/UB
Degenerate ($\theta = 0$)	Row or bound	Pivot or flip; Bland's	As above
No eligible r_j	—	Optimal	—

This is the mechanism used internally by lexicographic (preemptive) goal programming.