Nicholl-Lee-Nicholl Line Clipping

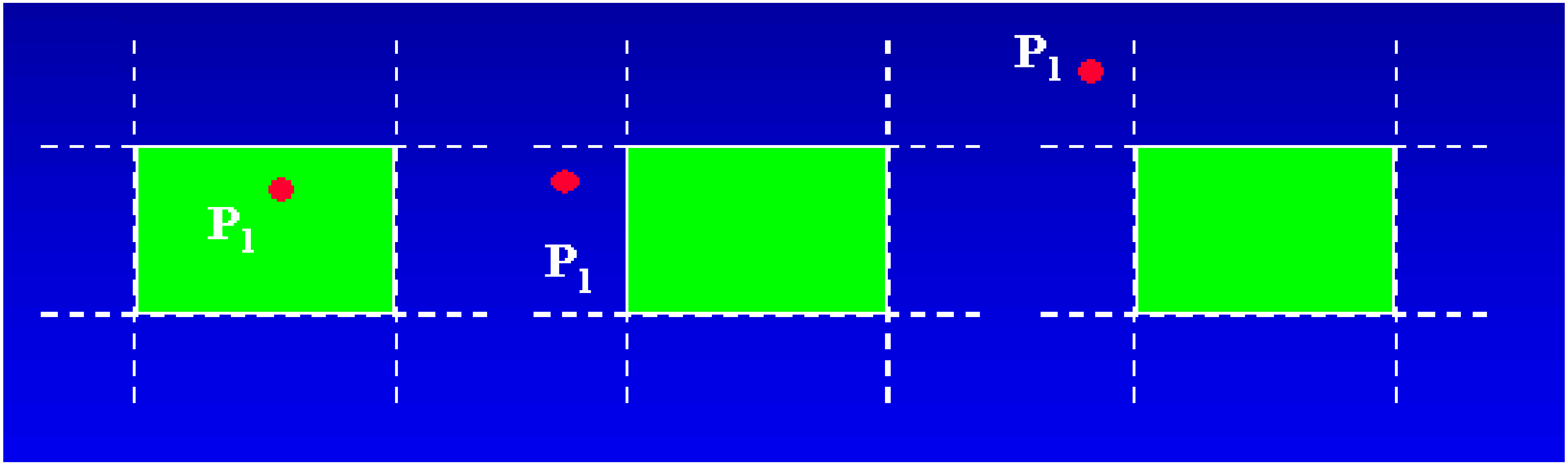
- Compared to C-S and L-B algorithms
  - NLN algorithm performs fewer comparisons and divisions.
  - NLN can only be applied to 2D clipping.
- The NLN algorithm
  - Clip a line with endpoints P_1 and P_2
  - First determine the position of P_1 for the nine possible regions.
    - Only three regions need be considered
    - The other regions using a symmetry transformation (Force Appro.Transformation)
  - Next determine the position of P_2 relative to P_1.

By Jignesh Patel

Nicholl-Lee-Nicholl Line Clipping

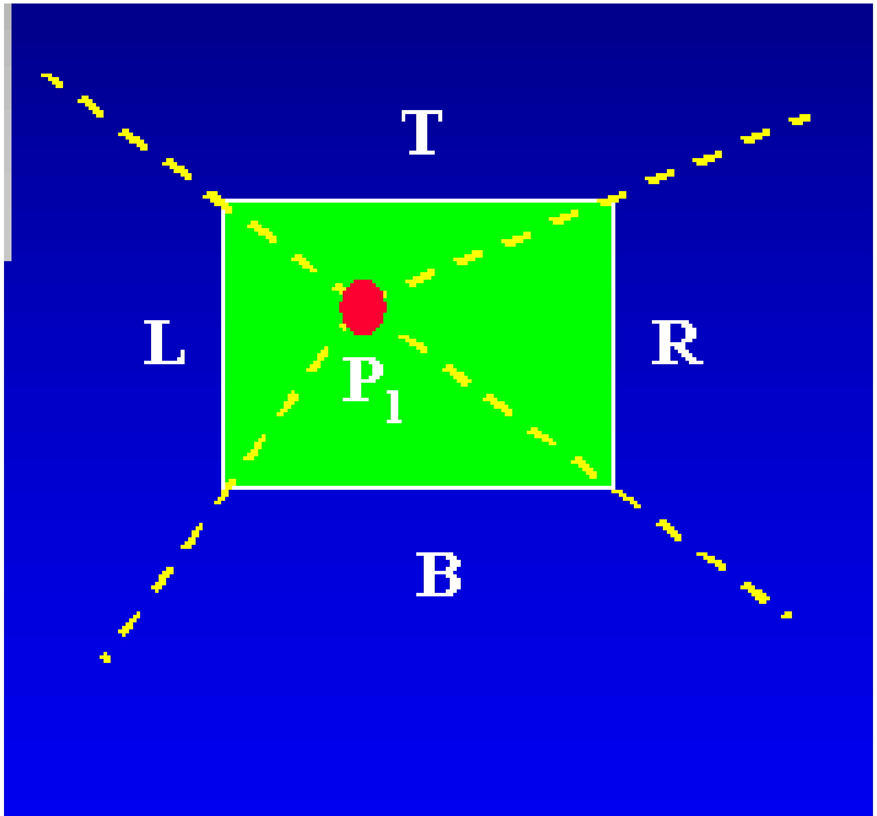
Here all 3 regions are representative of other six region on respective positions.

e.g.region above clip window can be transformed to the region left of the clip window using a reflection about the line y=-x or using 90 degree counterclockwise rotation.

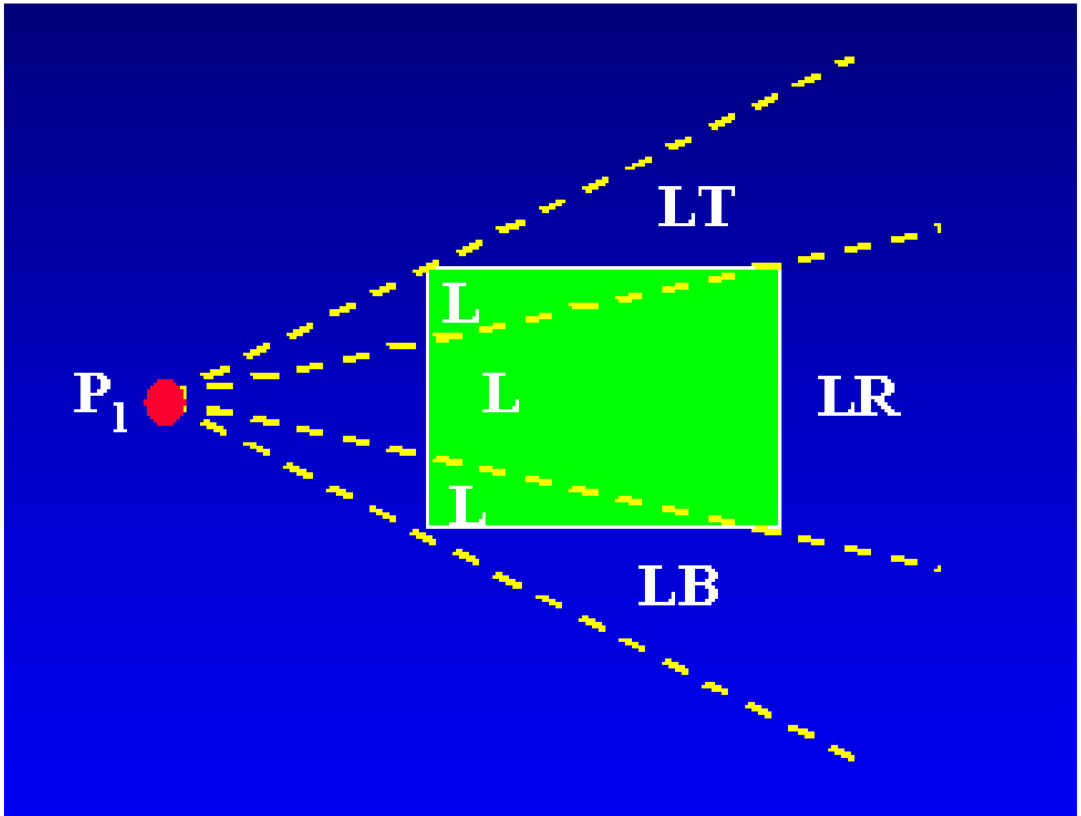


By Jignesh Patel

Nicholl-Lee-Nicholl Line Clipping



P_1 is inside the clip window and P_2 is outside

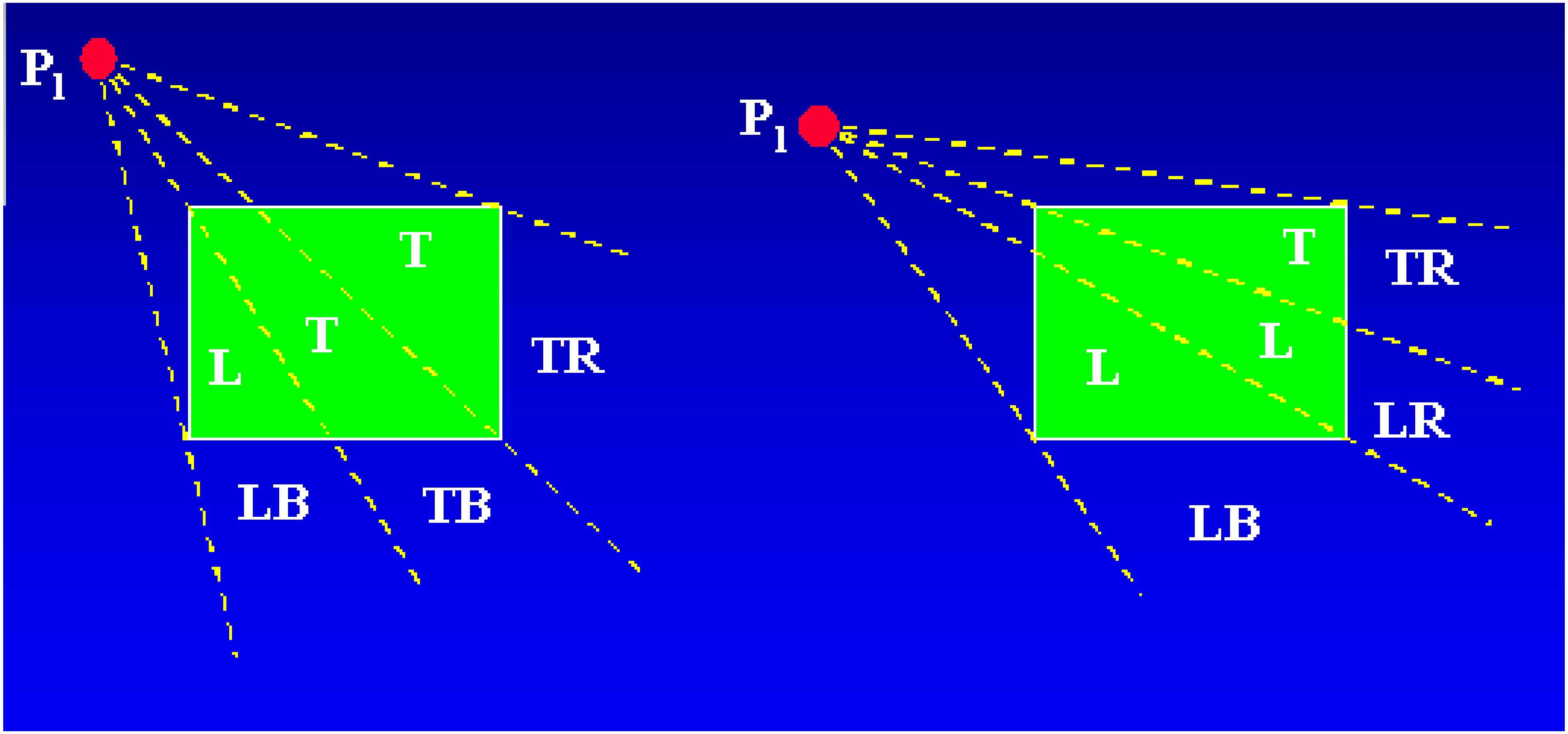


P_1 is directly to the left of the clip window

By Jignesh Patel

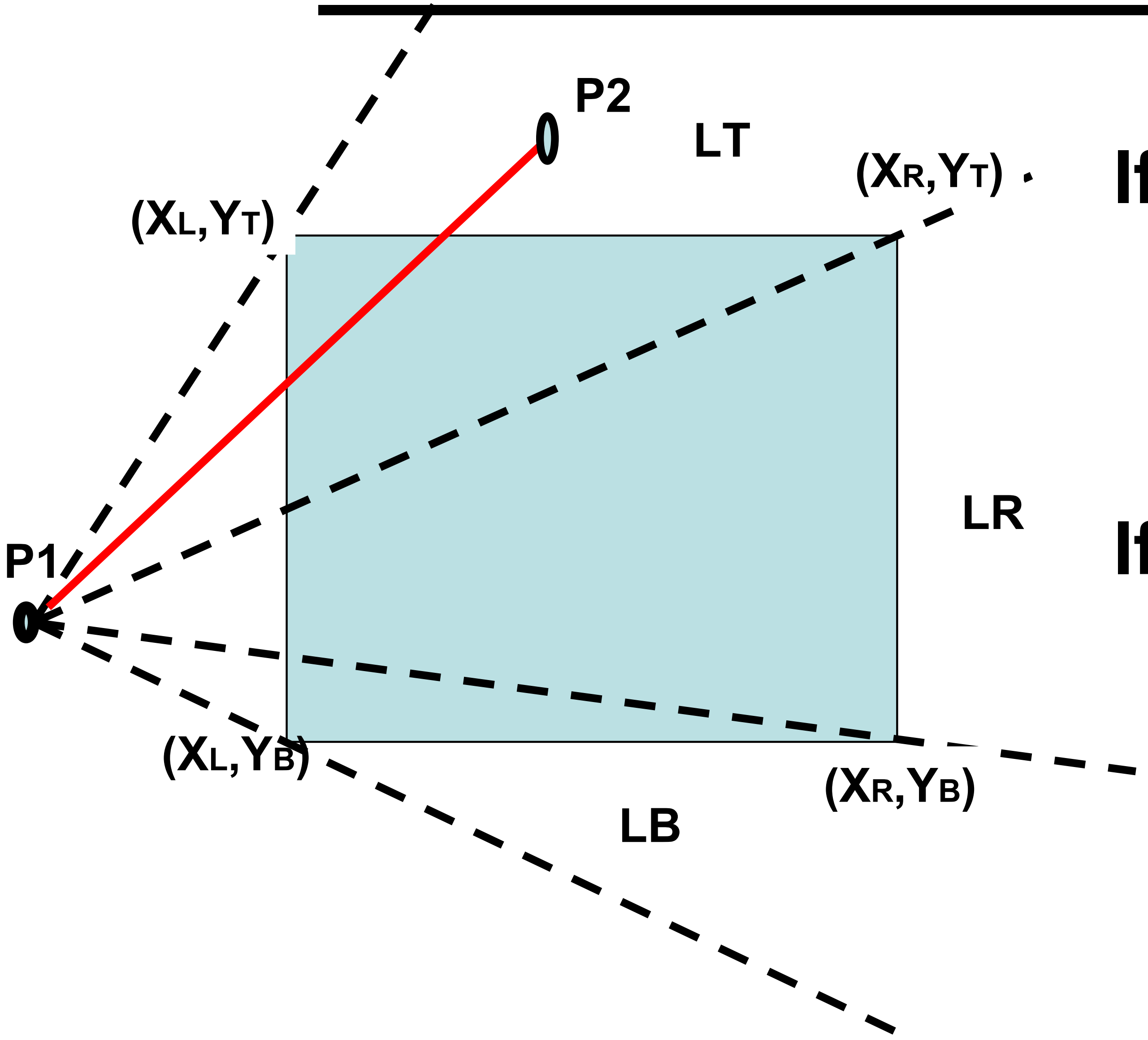
Nicholl-Lee-Nicholl Line Clipping

P1 is on top left region then 2 variations



By Jignesh Patel

P1 to the left of the window



- If P2 at LT save the line from Left boundary to Top boundary.
- If P2 is not in any of the 4 regions entire line is clipped

By Jignesh Patel

- **To determine the region in which P2 is located compare the slope of the line to the slope of the boundaries of clip regions.**
- **P1 is to Left of clip window and P2 in LT if**

$$slope\ \overline{P_1P_{TR}} < slope\ \overline{P_1P_2} < slope\ \overline{P_1P_{TL}}$$

$$\frac{y_T - y_1}{x_R - x_1} < \frac{y_2 - y_1}{x_2 - x_1} < \frac{y_T - y_1}{x_L - x_1}$$

- From parametric equations

  $x= x1 + u\ (x2-x1)$

  $y= y1 + u\ (y2-y1)$

- At x intersection position on left window boundary

  $X=XL$ $\qquad\qquad$ $u=(XL- X1)/(X2 – X1)$

  $Y=Y1 + (Y2-Y1)\ (XL-X1)\ /(X2-X1)$

- At intersection position on top boundary

  $Y=YT$ $\qquad\qquad$ $u= (YT- Y1)/(Y2 – Y1)$

  $X=X1 + (X2-X1)/(Y2-Y1)\ (YL-Y1)$