Theorem 6: b_l = 1/2

In[1]:=
```
(* See Mathematica codes at https://bit.ly/3vY8Bw1 *)
(* Define mu *)
mu =
  (-8 pf p^3 (-1 + bl) bl (-1 + br) br + 2 p^4 (-1 + bl) bl (-1 + br) br - 2 p (-2 + br + 3 pf br - pf
          br^2 + pf bl^2 (-1 + 4 pf^2 (-1 + br) br) + bl (1 + pf (3 - 4 br) - 4 pf^3 (-1 + br) br)) +
      pf (-br (-2 + pf + pf br) + pf bl^2 (-1 + 2 pf^2 (-1 + br) br) -
         bl (-2 + pf + 4 pf br + 2 pf^3 (-1 + br) br)) + p^2 (-8 + 7 br - br^2 +
         bl^2 (-1 + 12 pf^2 (-1 + br) br) + bl (7 + 4 (-1 + 3 pf^2) br - 12 pf^2 br^2))) /
   (1 + pf^4 (-1 + bl) bl (-1 + br) br - 4 pf p^3 (-1 + bl) bl (-1 + br) br +
      p^4 (-1 + bl) bl (-1 + br) br - pf^2 (bl + br + 2 bl br) -
      2 pf p (bl + br + 2 (-1 + pf^2) bl br + 2 pf^2 bl^2 (-1 + br) br - 2 pf^2 bl br^2) +
      p^2 (-4 + 3 br + 6 pf^2 bl^2 (-1 + br) br + bl (3 - 2 br + 6 pf^2 br - 6 pf^2 br^2)));
(* Take second-order derivative w.r.t. b_r *)
secondDerivative = D[mu, {br, 2}];
(* Take b_l = 1/2 *)
SOD = Simplify[secondDerivative /. {bl → 1/2}];
(* Specify range of parameters *)
conditions = 0 < br < 1 && 0 ≤ p < pf ≤ 1/2;
(* Verify if it is possible to have SOD ≥ 0;
returns false if SOD < 0 for all parameters within the range *)
Reduce[SOD ≥ 0 && conditions, {br, bl, pf, p}]
```

Out[5]= False

Theorem 6: b_l = 0

In[6]:=
```
(* Take second-order derivative w.r.t. b_r *)
secondDerivative = D[mu, {br, 2}];
(* Take b_l = 0 *)
SOD = Simplify[secondDerivative /. {bl → 0}];
(* Specify range of parameters *)
conditions = 0 < br < 1 && 0 ≤ p < pf ≤ 1/2;
(* Verify if it is possible to have SOD ≥ 0;
returns false if SOD < 0 for all parameters within the range *)
Reduce[SOD ≥ 0 && conditions, {br, bl, pf, p}]
```

Out[9]= False