Holling

Holling type III

In[∘]:=
```
Series[aij yj^k/(1 + aij hij yj^k), {hij, 0, 2}]
```

Out[∘]=

$$aij\ yj^k - aij^2\ yj^{2k}\ hij + aij^3\ yj^{3k}\ hij^2 + O[hij]^3$$

We will be focusing on Holling type II

In[∘]:=
```
Series[aij yj/(1 + aij hij yj), {hij, 0, 2}]
```

Out[∘]=

$$aij\ yj - aij^2\ yj^2\ hij + aij^3\ yj^3\ hij^2 + O[hij]^3$$

In[∘]:=
```
a = {{3.15, 0.6}, {2.5, 2.9}};
h = {{-0.135, -1.5}, {-0.1, -0.23}};
r1 = 0.1;
r2 = 0.25;
```

In[∘]:=
```
n = 2;
```

In[∘]:=

$$\left\{y_1'[t] == y_1[t]\left(r1 + \sum_{j=1}^{n}\left(a[[1, j]]\ y_j[t] + a[[1, j]]^2\ y_j[t]^2\ h[[1, j]]\right)\right),\right.$$
$$\left. y_2'[t] == y_2[t]\left(r2 + \sum_{j=1}^{n}\left(a[[2, j]]\ y_j[t] + a[[2, j]]^2\ y_j[t]^2\ h[[2, j]]\right)\right)\right\}$$

Out[∘]=

$$\{y_1'[t] == y_1[t]\ (0.1 + 3.15\ y_1[t] - 1.33954\ y_1[t]^2 + 0.6\ y_2[t] - 0.54\ y_2[t]^2),$$
$$y_2'[t] == y_2[t]\ (0.25 + 2.5\ y_1[t] - 0.625\ y_1[t]^2 + 2.9\ y_2[t] - 1.9343\ y_2[t]^2)\}$$

In[∘]:=
```
int = {y1, y2} /. Solve[
    {y1 (r1 + a[[1, 1]] y1 + a[[1, 1]]^2 h[[1, 1]] y1^2 + a[[1, 2]] y2 + a[[1, 2]]^2 h[[1, 2]] y2^2) == 0 &&
      y2 (r2 + a[[2, 1]] y1 + a[[2, 1]]^2 h[[2, 1]] y1^2 + a[[2, 2]] y2 + a[[2, 2]]^2 h[[2, 2]] y2^2) == 0},
    {y1, y2}, Assumptions → {y1 > 0 && y2 > 0}];
```

··· Solve : Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

In[ ]:=

$$\text{predprey} = \text{NDSolve}\Big[\Big\{$$

$$y_1{}'[t] == y_1[t]\left(r1 + \sum_{j=1}^{n}\left(a[\![1, j]\!]\, y_j[t] + a[\![1, j]\!]^2\, y_j[t]^2\, h[\![1, j]\!]\right)\right),$$

$$y_2{}'[t] == y_2[t]\left(r2 + \sum_{j=1}^{n}\left(a[\![2, j]\!]\, y_j[t] + a[\![2, j]\!]^2\, y_j[t]^2\, h[\![2, j]\!]\right)\right),$$

$$y_1[0] == 0.5,\ y_2[0] == 0.5\Big\},\ \{y_1, y_2\},\ \{t, 0, 5, 0.1\}\Big]$$

Out[ ]=

$$\Big\{\Big\{y_1 \to \text{InterpolatingFunction}\Big[$$ Domain: {{0., 5. }} Output: scalar $$\Big],$$

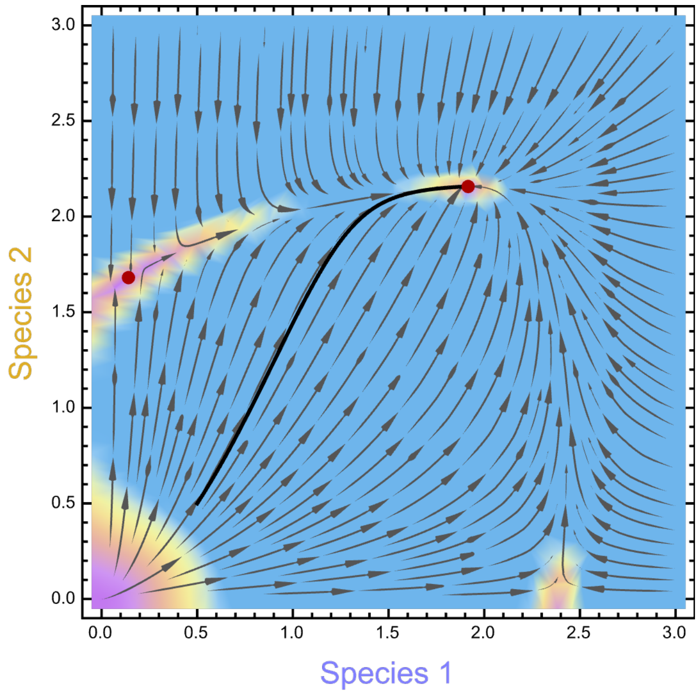
$$y_2 \to \text{InterpolatingFunction}\Big[$$ Domain: {{0., 5. }} Output: scalar $$\Big]\Big\}\Big\}$$

In[ ]:=

```
trajectory = ListPlot[
  Table[Evaluate[{y_1[t], y_2[t]} /. predprey][[1]], {t, 0, 5, 0.005}], Joined → True,
  Axes → None, PlotStyle → Black, PlotStyle → Black, PlotStyle → Automatic,
  Frame → True, FrameStyle → Directive[Black, Thickness[0.004]],
  PlotRange → {{-0.5, 3}, {-0.5, 3}},
  FrameLabel → {Style["Species 1", 16, RGBColor["#807FFF"]],
    Style["Species 2", 16, RGBColor["#DEAD26"]]},
  AspectRatio → 1, Epilog → {Red, PointSize[Medium], Point[int]}];
```

In[ ]:=
```
Show[StreamDensityPlot[
   {y1 (r1 + a[[1, 1]] y1 + a[[1, 1]]^2 h[[1, 1]] y1^2 + a[[1, 2]] y2 + a[[1, 2]]^2 h[[1, 2]] y2^2),
    y2 (r2 + a[[2, 1]] y1 + a[[2, 1]]^2 h[[2, 1]] y1^2 + a[[2, 2]] y2 + a[[2, 2]]^2 h[[2, 2]] y2^2)},
   {y1, 0, 3}, {y2, 0, 3}, Frame → True, ColorFunction → "Pastel",
   ColorFunctionScaling → False, StreamStyle → Darker[Gray],
   StreamColorFunction → None, StreamPoints → Fine, StreamMarkers → {"PinDart"},
   StreamScale → Large, FrameLabel → {Style["Species 1", 16, RGBColor["#807FFF"]],
     Style["Species 2", 16, RGBColor["#DEAD26"]]},
   AspectRatio → 1, PlotRange → {{-0.1, 3.1}, {-0.1, 3.1}}], trajectory,
  Epilog → {Darker[Red], PointSize[Large], Point[int]},
  FrameStyle → Directive[Black, Thickness[0.004]]]
```

Out[ ]=



Using the notation as used in the manuscript

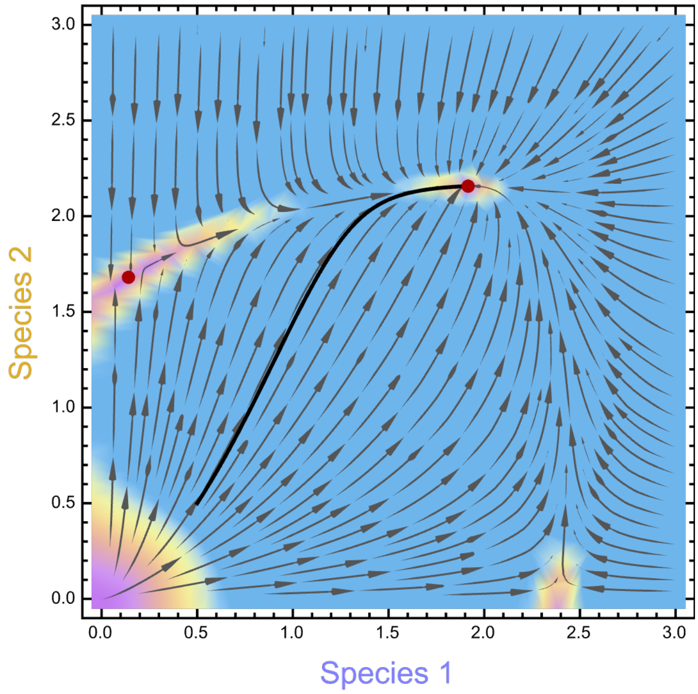
In[ ]:=
```
c =.;
bigc = Table[c_{i,j,k}, {i, 1, 2}, {j, 1, 2}, {k, 1, 2}];
```

In[ ]:=
```
c_{1,1,1} = h[[1, 1]] a[[1, 1]]^2; c_{1,1,2} = 0;
c_{1,2,2} = h[[1, 2]] a[[1, 2]]^2; c_{1,2,1} = 0;
```

In[ ]:=
```
c_{2,1,1} = h[[2, 1]] a[[2, 1]]^2; c_{2,1,2} = 0;
c_{2,2,2} = h[[2, 2]] a[[2, 2]]^2; c_{2,2,1} = 0;
```

In[∘]:=
```
Show[StreamDensityPlot[{y1 (r1 + a[[1, 1]] y1 +
      a[[1, 2]] y2 + c1,1,1 y1 y1 + c1,1,2 y1 y2 + c1,2,1 y2 y1 + c1,2,2 y2 y2 ),
    y2 (r2 + a[[2, 1]] y1 + a[[2, 2]] y2 + c2,1,1 y1 y1 +
      c2,1,2 y1 y2 + c2,2,1 y2 y1 + c2,2,2 y2 y2)}, {y1, 0, 3}, {y2, 0, 3},
   Frame → True, ColorFunction → "Pastel", ColorFunctionScaling → False,
   StreamStyle → Darker[Gray], StreamColorFunction → None,
   StreamPoints → Fine, StreamMarkers → {"PinDart"}, StreamScale → Large,
   FrameLabel → {Style["Species 1", 16, RGBColor["#807FFF"]],
     Style["Species 2", 16, RGBColor["#DEAD26"]]},
   AspectRatio → 1, PlotRange → {{-0.1, 3.1}, {-0.1, 3.1}}], trajectory,
  Epilog → {Darker[Red], PointSize[Large], Point[int]},
  FrameStyle → Directive[Black, Thickness[0.004]]]
```

Out[∘]=



Converting to Replicator

LV equations using the replicator matrix look like

In[ ]:=
```
m = 3;
(*Table[b_{i,j,k}=.,{i,1,m},{j,1,m},{k,1,m}]*)
Table[
 y_i'[t] == y_i[t] ((b_{i,m,m} - b_{m,m,m}) + Sum[(b_{i,j,m} - b_{m,j,m}) y_j[t], {j,1,n}] + Sum[(b_{i,m,k} - b_{m,m,k}) y_k[t], {k,1,n}] +
   (Sum[Sum[(b_{i,j,k} - b_{m,j,k}) y_j[t] y_k[t], {j,1,n}], {k,1,n}])), {i, 1, 2, 1}]
```

$$y_i{}'[t] = y_i[t]\left((b_{i,m,m}-b_{m,m,m})+\sum_{j=1}^{n}(b_{i,j,m}-b_{m,j,m})\,y_j[t]+\sum_{k=1}^{n}(b_{i,m,k}-b_{m,m,k})\,y_k[t]+\left(\sum_{j=1}^{n}\sum_{k=1}^{n}(b_{i,j,k}-b_{m,j,k})\,y_j[t]\,y_k[t]\right)\right),\ \{i,1,2,1\}$$

Out[ ]=

$$\{y_1'[t] == y_1[t]\ (b_{1,3,3}-b_{3,3,3}+(b_{1,1,3}-b_{3,1,3})\,y_1[t]+(b_{1,3,1}-b_{3,3,1})\,y_1[t]+(b_{1,1,1}-b_{3,1,1})\,y_1[t]^2+(b_{1,2,3}-b_{3,2,3})\,y_2[t]+(b_{1,3,2}-b_{3,3,2})\,y_2[t]+(b_{1,1,2}-b_{3,1,2})\,y_1[t]\,y_2[t]+(b_{1,2,1}-b_{3,2,1})\,y_1[t]\,y_2[t]+(b_{1,2,2}-b_{3,2,2})\,y_2[t]^2),$$

$$y_2'[t] == y_2[t]\ (b_{2,3,3}-b_{3,3,3}+(b_{2,1,3}-b_{3,1,3})\,y_1[t]+(b_{2,3,1}-b_{3,3,1})\,y_1[t]+(b_{2,1,1}-b_{3,1,1})\,y_1[t]^2+(b_{2,2,3}-b_{3,2,3})\,y_2[t]+(b_{2,3,2}-b_{3,3,2})\,y_2[t]+(b_{2,1,2}-b_{3,1,2})\,y_1[t]\,y_2[t]+(b_{2,2,1}-b_{3,2,1})\,y_1[t]\,y_2[t]+(b_{2,2,2}-b_{3,2,2})\,y_2[t]^2)\}$$

In[ ]:=
```
coeff1 =
  CoefficientList[Simplify[(b_{1,3,3} - b_{3,3,3} + (b_{1,1,3} - b_{3,1,3}) y1 + (b_{1,3,1} - b_{3,3,1}) y1 +
      (b_{1,1,1} - b_{3,1,1}) y1^2 + (b_{1,2,3} - b_{3,2,3}) y2 + (b_{1,3,2} - b_{3,3,2}) y2 + (b_{1,1,2} - b_{3,1,2})
       y1 y2 + (b_{1,2,1} - b_{3,2,1}) y1 y2 + (b_{1,2,2} - b_{3,2,2}) y2^2)], {y1, y2}] // Simplify;
```

In[ ]:=
```
coeff1rhs = CoefficientList[(r1 + a[[1, 1]] y1 + a[[1, 2]] y2 +
     c_{1,1,1} y1 y1 + c_{1,1,2} y1 y2 + c_{1,2,1} y2 y1 + c_{1,2,2} y2 y2 ), {y1, y2}];
```

In[ ]:=
```
coeff2 =
  CoefficientList[Simplify[(b_{2,3,3} - b_{3,3,3} + (b_{2,1,3} - b_{3,1,3}) y1 + (b_{2,3,1} - b_{3,3,1}) y1 +
     (b_{2,1,1} - b_{3,1,1}) y1^2 + (b_{2,2,3} - b_{3,2,3}) y2 + (b_{2,3,2} - b_{3,3,2}) y2 +
     (b_{2,1,2} - b_{3,1,2}) y1 y2 + (b_{2,2,1} - b_{3,2,1}) y1 y2 + (b_{2,2,2} - b_{3,2,2}) y2^2)], {y1, y2}];
```

In[ ]:=
```
coeff2rhs = CoefficientList[(r2 + a[[2, 1]] y1 + a[[2, 2]] y2 +
     c_{2,1,1} y1 y1 + c_{2,1,2} y1 y2 + c_{2,2,1} y2 y1 + c_{2,2,2} y2 y2), {y1, y2}];
```

Let us fill the last strategy to be 0s

In[ ]:=
```
Table[b_{m,i,j} = 0, {i, 1, m}, {j, 1, m}];
```

In[ ]:=
```
coeff1
```

Out[ ]=

$$\{\{b_{1,3,3},\ b_{1,2,3}+b_{1,3,2},\ b_{1,2,2}\},\ \{b_{1,1,3}+b_{1,3,1},\ b_{1,1,2}+b_{1,2,1},\ 0\},\ \{b_{1,1,1},\ 0,\ 0\}\}$$

In[ ]:=
```
coeff1rhs
```

Out[ ]=

```
{{0.1, 0.6, -0.54}, {3.15, 0, 0}, {-1.33954, 0, 0}}
```

In[ ]:=
```
b1,3,3 = coeff1rhs[[1, 1]];
b1,2,3 = coeff1rhs[[1, 2]];
b1,3,2 = 0;
b1,2,2 = coeff1rhs[[1, 3]];
b1,1,3 = coeff1rhs[[2, 1]];
b1,3,1 = 0;
b1,1,2 = coeff1rhs[[2, 2]];
b1,2,1 = 0;
b1,1,1 = coeff1rhs[[3, 1]];
```

In[ ]:=
```
coeff2
```

Out[ ]=

$\{\{b_{2,3,3}, b_{2,2,3}+b_{2,3,2}, b_{2,2,2}\}, \{b_{2,1,3}+b_{2,3,1}, b_{2,1,2}+b_{2,2,1}, 0\}, \{b_{2,1,1}, 0, 0\}\}$

In[ ]:=
```
coeff2rhs
```

Out[ ]=

```
{{0.25, 2.9, -1.9343}, {2.5, 0, 0}, {-0.625, 0, 0}}
```

In[ ]:=
```
b2,3,3 = coeff2rhs[[1, 1]];
b2,2,3 = coeff2rhs[[1, 2]];
b2,3,2 = 0;
b2,2,2 = coeff2rhs[[1, 3]];
b2,1,3 = coeff2rhs[[2, 1]];
b2,3,1 = 0;
b2,1,2 = coeff2rhs[[2, 2]];
b2,2,1 = 0;
b2,1,1 = coeff2rhs[[3, 1]];
```

In[ ]:=
```
B = Table[bi,j,k, {i, 1, m}, {j, 1, m}, {k, 1, m}]
```

Out[ ]=

```
{{{-1.33954, 0, 3.15}, {0, -0.54, 0.6}, {0, 0, 0.1}},
 {{-0.625, 0, 2.5}, {0, -1.9343, 2.9}, {0, 0, 0.25}},
 {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}}}
```

In[ ]:=
```
B // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} \begin{pmatrix} -1.33954 \\ 0 \\ 3.15 \end{pmatrix} & \begin{pmatrix} 0 \\ -0.54 \\ 0.6 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 0.1 \end{pmatrix} \\ \begin{pmatrix} -0.625 \\ 0 \\ 2.5 \end{pmatrix} & \begin{pmatrix} 0 \\ -1.9343 \\ 2.9 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 0.25 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \end{pmatrix}$$

Plotting the game dynamics

Simplex Plot

In[ ]:=
```
(* Geometric transformation to simplex *)
{err, trans} = FindGeometricTransform[
   {{1, Tan[Pi / 3]} / 2, {0, 0}, {1, 0}}, {{0, 0}, {0, 1}, {1, 0}}];
(* Edges of simplex *)
triangle = Graphics[{Thickness[0.005], Darker[Gray],
    GeometricTransformation[Line[{{0, 0}, {0, 1}, {1, 0}, {0, 0}}], trans]}];
(* Some random data *)
dummyData = Select[RandomReal[1, {100, 2}], Total[#] ≤ 1 &];
(* Plot the points *)
points = ListPlot[dummyData, PlotStyle → PointSize[0.03]];
(* Or plot the lines *)
lines = ListLinePlot[dummyData, PlotStyle → Black];
(* Show all together *)
(* The trick is to extract the "First" part of the plots, and transform it *)
```

In[ ]:=
```
payoffs = B;
fits[x_, y_] :=
  Table[Sum[Sum[(payoffs[[i, j, k]] x_j x_k), {k, 1, 3}], {j, 1, 3}], {i, 1, 3}] /. {x_1 → x, x_2 → y, x_3 → 1 - x - y};
```

$$\text{Table}\left[\sum_{j=1}^{3}\left(\sum_{k=1}^{3}(\text{payoffs}[[i,j,k]]\,x_j\,x_k)\right),\{i,1,3\}\right] /. \{x_1 \to x, x_2 \to y, x_3 \to 1-x-y\};$$

In[ ]:=
```
π1[x_, y_, z_] := fits[x, y][[1]];

π2[x_, y_, z_] := fits[x, y][[2]];

π3[x_, y_, z_] := fits[x, y][[3]];
πbar[x_, y_, z_] := x π1[x, y, z] + y π2[x, y, z] + z π3[x, y, z];
dx[x_, y_, z_] := x (π1[x, y, z] - πbar[x, y, z]);
dy[x_, y_, z_] := y (π2[x, y, z] - πbar[x, y, z]);
dz[x_, y_, z_] := z (π3[x, y, z] - πbar[x, y, z]);

Fermi[x_] := {
   dx[x[[1]], x[[2]], x[[3]]],
   dy[x[[1]], x[[2]], x[[3]]],
   dz[x[[1]], x[[2]], x[[3]]]
  }
x =.; y =.; z = 1 - x - y;
fa = fits[x, y][[1]];
fb = fits[x, y][[2]];
fc = fits[x, y][[3]];
```

In[ ]:=
```
x =.
y =.
sol =
  NSolve[{dx[x, y, 1 - x - y] == 0, dy[x, y, 1 - x - y] == 0}, {x, y}, NonNegativeReals];
```

```
In[◦]:= data = {x, y} /. sol;
relData = Select[data, Total[#] ≤ 1 && Total[#] ≥ 0 &];
p1 = ListPlot[relData, PlotStyle → {Black, PointSize[0.025]}];
p2 = ListPlot[relData, PlotStyle → {White, PointSize[0.015]}];
p3 = ListPlot[{{1, 0}, {0, 1}, {0, 0}}, PlotStyle → {Red, PointSize[0.03]}];
p4 = ListPlot[{{1, 0}, {0, 1}, {0, 0}}, PlotStyle → {White, PointSize[0.015]}];
points = Show[p1, p2, p3, p4];
```

Plotting isoclines

```
In[◦]:= eq1 = fa - fc;
eq2 = fb - fc;

In[◦]:= cplt = ContourPlot[eq1 == 0, {x, 0, 1},
    {y, 0, 1}, RegionFunction → Function[{x, y}, x + y ≤ 1],
    PlotRange → All, ContourStyle → Darker[Red]];
cplt2 = ContourPlot[eq2 == 0, {x, 0, 1},
    {y, 0, 1}, RegionFunction → Function[{x, y}, x + y ≤ 1],
    PlotRange → All, ContourStyle → Darker[Blue]];

In[◦]:= sp = StreamDensityPlot[{dx[x, y, 1 - x - y], dy[x, y, 1 - x - y]}, {x, 0, 1}, {y, 0, 1},
    Frame → True, ColorFunction → "Pastel", ColorFunctionScaling → False,
    StreamStyle → Darker[Gray], StreamColorFunction → None,
    StreamPoints → Fine, StreamMarkers → {"PinDart"}, StreamScale → Large,
    RegionFunction → Function[{x, y, vx, vy, n}, x + y ≤ 1 && x ≥ 0 && y ≥ 0]];

In[◦]:= trajdata = NDSolve[
    {x'[t] == dx[x[t], y[t], 1 - x[t] - y[t]], y'[t] == dy[x[t], y[t], 1 - x[t] - y[t]],
     x[0] == 0.45, y[0] == 0.08}, {x, y}, {t, 0, 50}];
parpl = ParametricPlot[Evaluate[{x[t], y[t]} /. trajdata],
    {t, 0, 50}, PlotRange → All, PlotStyle → Directive[Black, Dashed]];
```

In[ ]:=
```
psim = Show[
  Graphics[GeometricTransformation[First[Show[sp]], trans]], triangle,
  Graphics[GeometricTransformation[First[Show[points]], trans]],
  Graphics[GeometricTransformation[First[Show[cplt, cplt2]], trans]],
  Graphics[GeometricTransformation[First[parpl], trans]](*,
  Graphics[GeometricTransformation[Text["label",{0,1}],trans]]*)
 ]
```

Out[ ]=