```
Clear["Global`*"];

Quit[]

LaunchKernels[]
```

Figures Bistab sigmaprime / sigma

```
params[ss_, mm_] =
   {c → 0.05, mu → 1, β₀ → 10, k → 0.01, sig → ss, eps → 1, rm → 2, cup → 0, sigp → mm};

β[a_] = β₀ a / (1 + a); (* parasite trade-off*)

s[x_] =      s_max         /. c ->  sigp  s_max   /. b →  s_max  - 1 /. s_max → 10 // Simplify;
          ─────────────             ────  ─────              ─────
          1 + b Exp[-c x]           sig²    b               sig
(* superinfection function*)

rS[ga_] = rm / (1 + c ga) ; (* susceptible hosts trade-off*)

rI[ga_] = eps rS[ga]; (*infected hosts trade-off*)

getres[aa_, gg_] =
 Solve[{0 == (rS[gg] x + rI[gg] y) (1 - k (x + y)) - (mu + (β[aa] y)) x + gg y,
      0 == β[aa] x y - (mu + aa + gg) y}, {x, y}][[4]] // Simplify;
(* epidemiological system*)

para[a_, ga_] = Block[{}, res = getres[a, ga];
   Seq = x /. res;
   Ieq = y /. res;
   β'[a] - β[a] (1 - 2 β[a] Ieq sigp) / (mu + a + ga + sig β[a] Ieq)];
(* parasite selection gradient*)

hote[a_, ga_] = Block[{}, res = getres[a, ga];
    Seq = x /. res;
    Ieq = y /. res;
    rS'[ga] (1 - k (Seq + Ieq)) + (rI'[ga] (1 - k (Seq + Ieq)) β[a] Ieq) / (mu + a + ga) +
     (rS[ga] (1 - k (Seq + Ieq)) - mu) / (mu + a + ga)] // Simplify;
(*host selection gradient*)

getcoess[sig_, sigp_] :=
 Solve[{0 == para[a, ga], 0 == hote[a, ga]} /. params[sig, sigp], {a, ga}]

list = Table[{sig, sigp, getcoess[sig, sigp]},
    {sig, 0.0, 1.2, 0.01}, {sigp, -0.4, 0.4, 0.01}];

list = ParallelTable[{sig, sigp, getcoess[sig, sigp]},
    {sig, 0.0, 1.2, 0.005}, {sigp, -0.4, 0.4, 0.005}];
```

```
dataAl = {};
dataAl2 = {};
For[i = 1, i ≤ Dimensions[list][[2]], i++,
 For[j = 1, j ≤ Dimensions[list][[1]], j++,
  mysol = Select[{a, ga} /. list[[j, i, 3]], Element[#[[1]], Reals] &&
      #[[1]] > 0 && Element[#[[2]], Reals] && #[[2]] > 0 &];
  AppendTo[dataAl, {list[[j, i, 2]], list[[j, i, 1]], Length[mysol]}];
  AppendTo[dataAl2, {list[[j, i, 2]], list[[j, i, 1]], mysol}];
  Export["datasigmapsigma2.csv", dataAl, "CSV"];
  Export["datasigmapsigmadat2.csv", dataAl2, "CSV"];
 ]
]
Export["listbrutsigmap.csv", list, "CSV"];
```

```
Export["listbrutsigmap.csv", list, "CSV"];
```

```
list = Table[{sig, sigp, getcoess[sig, sigp]},
    {sig, 0.0, 1.2, 0.01}, {sigp, -0.4, 0.4, 0.01}];
```

```
dataAll = {};
dataEqI = {};
dataEqS = {};
dataAllEqI = {};
dataAllEqS = {};
dataEqIsigmap = {};
dataEqSsigmap = {};
dataAll2 = {};
For[i = 1, i ≤ Dimensions[list][[2]], i++,
 For[j = 1, j ≤ Dimensions[list][[1]], j++,
  mysol = Select[{a, ga} /. list[[j, i, 3]], Element[#[[1]], Reals] &&
      #[[1]] > 0 && Element[#[[2]], Reals] && #[[2]] > 0 &];

  For[l = 1, l ≤ Length[mysol], l++,
   findEqI = Ieq /. res /. params[list[[j, i, 1]], list[[j, i, 2]]] /.
      a → mysol[[l, 1]] /. ga → mysol[[l, 2]];
   findEqS = Seq /. res /. params[list[[j, i, 1]], list[[j, i, 2]]] /.
      a → mysol[[l, 1]] /. ga → mysol[[l, 2]];
   AppendTo[dataEqI, {list[[j, i, 2]], list[[j, i, 1]], findEqI}];
   AppendTo[dataEqS, {list[[j, i, 2]], list[[j, i, 1]], findEqS}];
  ];

  EqI = Select[dataEqI[[All, 3]], # > 0 &] ;
  EqS = Select[dataEqS[[All, 3]], # < 100 &] ;
  AppendTo[dataAllEqS, {list[[j, i, 2]], list[[j, i, 1]], EqS}];
  AppendTo[dataAllEqI, {list[[j, i, 2]], list[[j, i, 1]], EqI}];

  (*AppendTo[dataAll,{list[[j,i,2]],list[[j,i,1]],Length[mysol]}];
  AppendTo[dataAll2,{list[[j,i,2]],list[[j,i,1]],mysol}];*)

  AppendTo[dataEqSsigmap, {list[[j, i, 2]], list[[j, i, 1]], Length[EqS]}];
  AppendTo[dataEqIsigmap, {list[[j, i, 2]], list[[j, i, 1]], Length[EqI]}];
  Export["datasigmapsigmaEqS.csv", dataEqSsigmap, "CSV"];
  Export["datasigmapsigmaEqI.csv", dataEqIsigmap, "CSV"];
  dataEqI = {}; dataEqS = {};
 ]
]
Export["datasigmapsigmaEqI-brut.csv", dataAllEqI, "CSV"];
Export["datasigmapsigmaEqS-brut.csv", dataAllEqS, "CSV"];
(*Export["datasigmapsigma2.csv",dataAll,"CSV"];
Export["datasigmapsigmadat2.csv",dataAll2,"CSV"];*)
```

Part::partw : Part 18 of {{0., −0.4, {{a → −0.650303, ga → 118.59}, {a → 1388.55, ga → 152.169}}}, ≪15≫, {0., 0.4, {{a → 10.7185
+ 2.75015 i, ga → −2.92572 − 3.22958 i}, {a → 10.7185
− 2.75015 i, ga → −2.92572 + 3.22958 i}, {a → 0.513827 + 0.330348 i, ga → 2.52615
− 15.5988 i}, {a → 0.513827 − 0.330348 i, ga → 2.52615 + 15.5988 i}}}} does not exist. ≫

ReplaceAll::reps :

{{{{0., −0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.3, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, ≪12≫, {0., 0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., 0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}}, {≪1≫}, ≪21≫, {≪1≫}, {≪1≫}}[[1, 18, 3]]} is

neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing. ≫

ReplaceAll::argt : ReplaceAll called with 0 arguments; 1 or 2 arguments are expected. ≫

Part::partw : Part 19 of {{0., −0.4, {{a → −0.650303, ga → 118.59}, {a → 1388.55, ga → 152.169}}}, ≪15≫, {0., 0.4, {{a → 10.7185 + 2.75015 i, ga → −2.92572 − 3.22958 i}, {a → 10.7185 − 2.75015 i, ga → −2.92572 + 3.22958 i}, {a → 0.513827 + 0.330348 i, ga → 2.52615 − 15.5988 i}, {a → 0.513827 − 0.330348 i, ga → 2.52615 + 15.5988 i}}}} does not exist. ≫

ReplaceAll::reps :

{{{{0., −0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.3, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, ≪12≫, {0., 0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., 0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}}, {≪1≫}, ≪21≫, {≪1≫}, {≪1≫}}[[1, 19, 3]]} is

neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing. ≫

ReplaceAll::argt : ReplaceAll called with 0 arguments; 1 or 2 arguments are expected. ≫

Part::partw : Part 20 of {{0., −0.4, {{a → −0.650303, ga → 118.59}, {a → 1388.55, ga → 152.169}}}, ≪15≫, {0., 0.4, {{a → 10.7185 + 2.75015 i, ga → −2.92572 − 3.22958 i}, {a → 10.7185 − 2.75015 i, ga → −2.92572 + 3.22958 i}, {a → 0.513827 + 0.330348 i, ga → 2.52615 − 15.5988 i}, {a → 0.513827 − 0.330348 i, ga → 2.52615 + 15.5988 i}}}} does not exist. ≫

General::stop : Further output of Part::partw will be suppressed during this calculation. ≫

ReplaceAll::reps :

{{{{0., −0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., −0.3, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, ≪12≫, {0., 0.35, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}, {0., 0.4, {{Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}, {Rule[≪2≫], Rule[≪2≫]}}}}, {≪1≫}, ≪21≫, {≪1≫}, {≪1≫}}[[1, 20, 3]]} is

neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing. ≫

General::stop : Further output of ReplaceAll::reps will be suppressed during this calculation. ≫

ReplaceAll::argt : ReplaceAll called with 0 arguments; 1 or 2 arguments are expected. ≫

General::stop : Further output of ReplaceAll::argt will be suppressed during this calculation. ≫