Cooperation increases robustness to ecological disturbance in microbial cross-feeding networks

Generating Random Networks

Functions to calculate Entropy and Assortativity

Entropy

Assortativity

1. Colimitation model

Solving the system of ODE

The function “fNewSaitoK” solves the ODE system and gives the population at steady state of the system. The function “fNewSaitoK” receives a network and a disturbance value as arguments.

```
fNewSaitoK[Net_, Dh_] := (

  dB1 =
   B1[t] (-B1[t] κ1 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
        M5[t]/(denK + M5[t])) - (c1,1 + c1,2 + c1,3 + c1,4 + c1,5 + Dh) B1[t];
  dB2 = B2[t] (-B2[t] κ2 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
        M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c2,1 + c2,2 + c2,3 + c2,4 + c2,5 + Dh) B2[t];
  dB3 = B3[t] (-B3[t] κ3 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
```

```
      M4[t]        M5[t]
   ---------- * ----------) - (c3,1 + c3,2 + c3,3 + c3,4 + c3,5 + Dh) B3[t];
   denK + M4[t]  denK + M5[t]
dB4 = B4[t] (-B4[t] κ4 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c4,1 + c4,2 + c4,3 + c4,4 + c4,5 + Dh) B4[t];
dB5 = B5[t] (-B5[t] κ5 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c5,1 + c5,2 + c5,3 + c5,4 + c5,5 + Dh) B5[t];

dM1 = -M1[t] q1 +
   (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t]))
    (-B1[t] d1,1 - B2[t] d1,2 - B3[t] d1,3 - B4[t] d1,4 - B5[t] d1,5) +
   B1[t] Ω1,1 + B2[t] Ω1,2 + B3[t] Ω1,3 + B4[t] Ω1,4 + B5[t] Ω1,5;
dM2 = -M2[t] q2 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d2,1 - B2[t] d2,2 - B3[t] d2,3 - B4[t] d2,4 - B5[t] d2,5) +
   B1[t] Ω2,1 + B2[t] Ω2,2 + B3[t] Ω2,3 + B4[t] Ω2,4 + B5[t] Ω2,5;
dM3 = -M3[t] q3 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d3,1 - B2[t] d3,2 - B3[t] d3,3 - B4[t] d3,4 - B5[t] d3,5) +
   B1[t] Ω3,1 + B2[t] Ω3,2 + B3[t] Ω3,3 + B4[t] Ω3,4 + B5[t] Ω3,5;
dM4 = -M4[t] q4 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d4,1 - B2[t] d4,2 - B3[t] d4,3 - B4[t] d4,4 - B5[t] d4,5) +
   B1[t] Ω4,1 + B2[t] Ω4,2 + B3[t] Ω4,3 + B4[t] Ω4,4 + B5[t] Ω4,5;
dM5 = -M5[t] q5 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d5,1 - B2[t] d5,2 - B3[t] d5,3 - B4[t] d5,4 - B5[t] d5,5) +
   B1[t] Ω5,1 + B2[t] Ω5,2 + B3[t] Ω5,3 + B4[t] Ω5,4 + B5[t] Ω5,5;

KK = 0.2;
cc = 0.05;
qq = 0.3;
dd = 0.00015;
OM = 1;
nu = 1500;
den = 2;
```

```
tmax = 1000;
par = {
  κ1 → KK, κ2 → KK, κ3 → KK, κ4 → KK, κ5 → KK,

  c1,1 → cc Net[[1]][[1]], c1,2 → cc Net[[1]][[2]],
  c1,3 → cc Net[[1]][[3]], c1,4 → cc Net[[1]][[4]], c1,5 → cc Net[[1]][[5]],
  c2,1 → cc Net[[2]][[1]], c2,2 → cc Net[[2]][[2]], c2,3 → cc Net[[2]][[3]],
  c2,4 → cc Net[[2]][[4]], c2,5 → cc Net[[2]][[5]],
  c3,1 → cc Net[[3]][[1]], c3,2 → cc Net[[3]][[2]], c3,3 → cc Net[[3]][[3]],
  c3,4 → cc Net[[3]][[4]], c3,5 → cc Net[[3]][[5]],
  c4,1 → cc Net[[4]][[1]], c4,2 → cc Net[[4]][[2]], c4,3 → cc Net[[4]][[3]],
  c4,4 → cc Net[[4]][[4]], c4,5 → cc Net[[4]][[5]],
  c5,1 → cc Net[[5]][[1]], c5,2 → cc Net[[5]][[2]], c5,3 → cc Net[[5]][[3]],
  c5,4 → cc Net[[5]][[4]], c5,5 → cc Net[[5]][[5]],

  q1 → qq, q2 → qq, q3 → qq, q4 → qq, q5 → qq,

  d1,1 → dd, d1,2 → dd, d1,3 → dd, d1,4 → dd, d1,5 → dd,
  d2,1 → dd, d2,2 → dd, d2,3 → dd, d2,4 → dd, d2,5 → dd,
  d3,1 → dd, d3,2 → dd, d3,3 → dd, d3,4 → dd, d3,5 → dd,
  d4,1 → dd, d4,2 → dd, d4,3 → dd, d4,4 → dd, d4,5 → dd,
  d5,1 → dd, d5,2 → dd, d5,3 → dd, d5,4 → dd, d5,5 → dd,

  Ω1,1 → OM Net[[1]][[1]], Ω1,2 → OM Net[[1]][[2]],
  Ω1,3 → OM Net[[1]][[3]], Ω1,4 → OM Net[[1]][[4]], Ω1,5 → OM Net[[1]][[5]],
  Ω2,1 → OM Net[[2]][[1]], Ω2,2 → OM Net[[2]][[2]], Ω2,3 → OM Net[[2]][[3]],
  Ω2,4 → OM Net[[2]][[4]], Ω2,5 → OM Net[[2]][[5]],
  Ω3,1 → OM Net[[3]][[1]], Ω3,2 → OM Net[[3]][[2]], Ω3,3 → OM Net[[3]][[3]],
  Ω3,4 → OM Net[[3]][[4]], Ω3,5 → OM Net[[3]][[5]],
  Ω4,1 → OM Net[[4]][[1]], Ω4,2 → OM Net[[4]][[2]], Ω4,3 → OM Net[[4]][[3]],
  Ω4,4 → OM Net[[4]][[4]], Ω4,5 → OM Net[[4]][[5]],
  Ω5,1 → OM Net[[5]][[1]], Ω5,2 → OM Net[[5]][[2]], Ω5,3 → OM Net[[5]][[3]],
  Ω5,4 → OM Net[[5]][[4]], Ω5,5 → OM Net[[5]][[5]],
  nuK → nu,
  denK → den


 };


B10 = 1500;
B20 = 1500;
B30 = 1500;
```

```
B40 = 1500;
B50 = 1500;
M10 = 10;
M20 = 10;
M30 = 10;
M40 = 10;
M50 = 10;


sol =
 NDSolve[
  {
    B1'[t] == dB1,
    B2'[t] == dB2,
    B3'[t] == dB3,
    B4'[t] == dB4,
    B5'[t] == dB5,

    M1'[t] == dM1,
    M2'[t] == dM2,
    M3'[t] == dM3,
    M4'[t] == dM4,
    M5'[t] == dM5,



    B1[0] == B10,
    B2[0] == B20,
    B3[0] == B30,
    B4[0] == B40,
    B5[0] == B50,
    M1[0] == M10,
    M2[0] == M20,
    M3[0] == M30,
    M4[0] == M40,
    M5[0] == M50

   } /. par,
  {B1, B2, B3, B4, B5, M1, M2, M3, M4, M5},
  {t, 0, tmax}];

{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax],
   M1[tmax], M2[tmax], M3[tmax], M4[tmax], M5[tmax]} /. sol /. par

(*Min[{B1[tmax],B2[tmax],B3[tmax],B4[tmax],B5[tmax]}/.sol/.par]*)
```

```
)
```

As an example let's take the following Network

```
In[●]:= NetK = {
    {0, 1, 0, 1, 0},
    {1, 0, 1, 1, 0},
    {1, 0, 1, 0, 1},
    {0, 1, 0, 1, 0},
    {0, 0, 0, 0, 1}
  };

In[4050]:= fNewSaitoK[NetK, 0]
fNewSaitoK[NetK, 1]

Out[4050]= {{6661.68, 6661.43, 6661.43, 6661.68,
   6661.93, 22219.9, 44425.5, 44426.3, 22219.9, 15.9422}}

Out[4051]= {{6656.68, 6656.43, 6656.43, 6656.68,
   6656.93, 22203.2, 44392.1, 44393., 22203.2, 15.9421}}
```

The function "fNewSaito" solves the ODE system and gives the lowest microbial population size (this is used to calculate the Robustness). The function "fNewSaito" receives a network and a disturbance value as arguments.

```
fNewSaito[Net_, Dh_] := (
```

$$dB_1 = B_1[t]\left(-B_1[t]\,\kappa_1 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{1,1} + c_{1,2} + c_{1,3} + c_{1,4} + c_{1,5} + Dh\right) B_1[t];$$

$$dB_2 = B_2[t]\left(-B_2[t]\,\kappa_2 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{2,1} + c_{2,2} + c_{2,3} + c_{2,4} + c_{2,5} + Dh\right) B_2[t];$$

$$dB_3 = B_3[t]\left(-B_3[t]\,\kappa_3 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{3,1} + c_{3,2} + c_{3,3} + c_{3,4} + c_{3,5} + Dh\right) B_3[t];$$

$$dB_4 = B_4[t]\left(-B_4[t]\,\kappa_4 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} *\right.$$

```
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c4,1 + c4,2 + c4,3 + c4,4 + c4,5 + Dh) B4[t];
dB5 = B5[t] (-B5[t] κ5 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c5,1 + c5,2 + c5,3 + c5,4 + c5,5 + Dh) B5[t];

dM1 = -M1[t] q1 +
   (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t]))
    (-B1[t] d1,1 - B2[t] d1,2 - B3[t] d1,3 - B4[t] d1,4 - B5[t] d1,5) +
   B1[t] Ω1,1 + B2[t] Ω1,2 + B3[t] Ω1,3 + B4[t] Ω1,4 + B5[t] Ω1,5;
dM2 = -M2[t] q2 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d2,1 - B2[t] d2,2 - B3[t] d2,3 - B4[t] d2,4 - B5[t] d2,5) +
   B1[t] Ω2,1 + B2[t] Ω2,2 + B3[t] Ω2,3 + B4[t] Ω2,4 + B5[t] Ω2,5;
dM3 = -M3[t] q3 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d3,1 - B2[t] d3,2 - B3[t] d3,3 - B4[t] d3,4 - B5[t] d3,5) +
   B1[t] Ω3,1 + B2[t] Ω3,2 + B3[t] Ω3,3 + B4[t] Ω3,4 + B5[t] Ω3,5;
dM4 = -M4[t] q4 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d4,1 - B2[t] d4,2 - B3[t] d4,3 - B4[t] d4,4 - B5[t] d4,5) +
   B1[t] Ω4,1 + B2[t] Ω4,2 + B3[t] Ω4,3 + B4[t] Ω4,4 + B5[t] Ω4,5;
dM5 = -M5[t] q5 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d5,1 - B2[t] d5,2 - B3[t] d5,3 - B4[t] d5,4 - B5[t] d5,5) +
   B1[t] Ω5,1 + B2[t] Ω5,2 + B3[t] Ω5,3 + B4[t] Ω5,4 + B5[t] Ω5,5;

KK = 0.2;
cc = 0.05;
qq = 0.3;
dd = 0.00015;
OM = 1;
nu = 1500;
den = 2;

tmax = 1000;
par = {
   κ1 → KK, κ2 → KK, κ3 → KK, κ4 → KK, κ5 → KK,
```

```
  c1,1 → cc Net[[1]][[1]], c1,2 → cc Net[[1]][[2]],
  c1,3 → cc Net[[1]][[3]], c1,4 → cc Net[[1]][[4]], c1,5 → cc Net[[1]][[5]],
  c2,1 → cc Net[[2]][[1]], c2,2 → cc Net[[2]][[2]], c2,3 → cc Net[[2]][[3]],
  c2,4 → cc Net[[2]][[4]], c2,5 → cc Net[[2]][[5]],
  c3,1 → cc Net[[3]][[1]], c3,2 → cc Net[[3]][[2]], c3,3 → cc Net[[3]][[3]],
  c3,4 → cc Net[[3]][[4]], c3,5 → cc Net[[3]][[5]],
  c4,1 → cc Net[[4]][[1]], c4,2 → cc Net[[4]][[2]], c4,3 → cc Net[[4]][[3]],
  c4,4 → cc Net[[4]][[4]], c4,5 → cc Net[[4]][[5]],
  c5,1 → cc Net[[5]][[1]], c5,2 → cc Net[[5]][[2]], c5,3 → cc Net[[5]][[3]],
  c5,4 → cc Net[[5]][[4]], c5,5 → cc Net[[5]][[5]],

  q1 → qq, q2 → qq, q3 → qq, q4 → qq, q5 → qq,

  d1,1 → dd, d1,2 → dd, d1,3 → dd, d1,4 → dd, d1,5 → dd,
  d2,1 → dd, d2,2 → dd, d2,3 → dd, d2,4 → dd, d2,5 → dd,
  d3,1 → dd, d3,2 → dd, d3,3 → dd, d3,4 → dd, d3,5 → dd,
  d4,1 → dd, d4,2 → dd, d4,3 → dd, d4,4 → dd, d4,5 → dd,
  d5,1 → dd, d5,2 → dd, d5,3 → dd, d5,4 → dd, d5,5 → dd,

  Ω1,1 → OM Net[[1]][[1]], Ω1,2 → OM Net[[1]][[2]],
  Ω1,3 → OM Net[[1]][[3]], Ω1,4 → OM Net[[1]][[4]], Ω1,5 → OM Net[[1]][[5]],
  Ω2,1 → OM Net[[2]][[1]], Ω2,2 → OM Net[[2]][[2]], Ω2,3 → OM Net[[2]][[3]],
  Ω2,4 → OM Net[[2]][[4]], Ω2,5 → OM Net[[2]][[5]],
  Ω3,1 → OM Net[[3]][[1]], Ω3,2 → OM Net[[3]][[2]], Ω3,3 → OM Net[[3]][[3]],
  Ω3,4 → OM Net[[3]][[4]], Ω3,5 → OM Net[[3]][[5]],
  Ω4,1 → OM Net[[4]][[1]], Ω4,2 → OM Net[[4]][[2]], Ω4,3 → OM Net[[4]][[3]],
  Ω4,4 → OM Net[[4]][[4]], Ω4,5 → OM Net[[4]][[5]],
  Ω5,1 → OM Net[[5]][[1]], Ω5,2 → OM Net[[5]][[2]], Ω5,3 → OM Net[[5]][[3]],
  Ω5,4 → OM Net[[5]][[4]], Ω5,5 → OM Net[[5]][[5]],
  nuK → nu,
  denK → den


 };


B10 = 1500;
B20 = 1500;
B30 = 1500;
B40 = 1500;
B50 = 1500;
M10 = 10;
M20 = 10;
```

```
M30 = 10;
M40 = 10;
M50 = 10;


sol =
 NDSolve[
  {
    B1'[t] == dB1,
    B2'[t] == dB2,
    B3'[t] == dB3,
    B4'[t] == dB4,
    B5'[t] == dB5,

    M1'[t] == dM1,
    M2'[t] == dM2,
    M3'[t] == dM3,
    M4'[t] == dM4,
    M5'[t] == dM5,


    B1[0] == B10,
    B2[0] == B20,
    B3[0] == B30,
    B4[0] == B40,
    B5[0] == B50,
    M1[0] == M10,
    M2[0] == M20,
    M3[0] == M30,
    M4[0] == M40,
    M5[0] == M50

   } /. par,
  {B1, B2, B3, B4, B5, M1, M2, M3, M4, M5},
  {t, 0, tmax}];

{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax],
   M1[tmax], M2[tmax], M3[tmax], M4[tmax], M5[tmax]} /. sol /. par;

Min[{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax]} /. sol /. par]

)
```

The function “robustnessNewSaito” uses the previous function “fNewSaito” and calculates the Robustness. The function “robustnessNewSaito” simply receives a network as an argument.

In[4053]:=
```
robustnessNewSaito[NetTop_] := (
  n1 = 1;
  n2 = 5000;
  mid = (n1 + n2) / 2;

  While[(n1 ≠ mid && n2 ≠ mid),
   (If[fNewSaito[NetTop, mid] < 1, n2 = mid, n1 = mid];
    mid = Floor[N[(n1 + n2) / 2]];); {n1, n2, mid}]; mid

 )
```

As an example let’s take the following Network

In[4054]:=
```
NetK = {
    {0, 1, 0, 1, 0},
    {1, 0, 1, 1, 0},
    {1, 0, 1, 0, 1},
    {0, 1, 0, 1, 0},
    {0, 0, 0, 0, 1}
  };
```

Using the function fNewSaito we can calculate the smallest value of a bacterial population in the community for a given disturbance vale. For example, let’s take Disturbance value 1 and 500:

In[4055]:=
```
fNewSaito[NetK, 0]
```

Out[4055]= 6661.43

In[4072]:=
```
fNewSaito[NetK, 500]
```

Out[4072]= 4158.75

Using the function fNewSaito we can calculate Robustness of the Network:

In[4073]:=
```
robustnessNewSaito[NetK]
```

Out[4073]= 924

We can calculate the (Relative) Entropy and the Assortativity:

In[4047]:=
```
RelatEntrop5[NetK]
```
Out[4047]=
```
0.960956
```

In[4048]:=
```
assortativity[NetK]
```
Out[4048]=
```
-0.113228
```

We can calculate the robustness of the previously generated random networks with different number of auxotrophies:

In[4084]:=
```
AuxoComm6 = robustnessNewSaito /@ hk6;
AuxoComm7 = robustnessNewSaito /@ hk7;
AuxoComm8 = robustnessNewSaito /@ hk8;
AuxoComm9 = robustnessNewSaito /@ hk9;
AuxoComm10 = robustnessNewSaito /@ hk10;
AuxoComm11 = robustnessNewSaito /@ hk11;
AuxoComm12 = robustnessNewSaito /@ hk12;
AuxoComm13 = robustnessNewSaito /@ hk13;
AuxoComm14 = robustnessNewSaito /@ hk14;
AuxoComm15 = robustnessNewSaito /@ hk15;
AuxoComm16 = robustnessNewSaito /@ hk16;
AuxoComm17 = robustnessNewSaito /@ hk17;
```

In[4190]:=
```
Lik = {AuxoComm6, AuxoComm7, AuxoComm8, AuxoComm9, AuxoComm10, AuxoComm11,
   AuxoComm12, AuxoComm13, AuxoComm14, AuxoComm15, AuxoComm16, AuxoComm17};
```

In[4191]:=
```
Needs["HypothesisTesting`"]
```

In[4192]:=
```
meCI = MeanCI[#, ConfidenceLevel → .95] & /@ Lik
```
Out[4192]=
```
{{1054.37, 1056.81}, {1038.49, 1041.59}, {1025.5, 1028.94}, {1010.44, 1013.76},
 {992.199, 996.361}, {976.85, 981.43}, {957.029, 961.511}, {940.573, 944.667},
 {919.209, 923.531}, {899.701, 903.539}, {877.074, 881.286}, {851.202, 854.318}}
```

In[4193]:=
```
wk[x_] :=
 (Around[Mean[{x[[1]], x[[2]]}], Max[{x[[1]], x[[2]]} - Mean[{x[[1]], x[[2]]}]]])
```

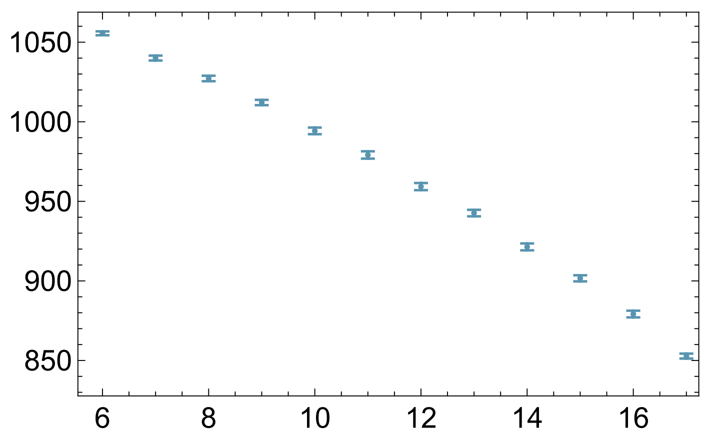
In[4194]:=
```
Rob5 = wk /@ meCI
```

Out[4194]=
```
{1055.6 ± 1.2, 1040.0 ± 1.5, 1027.2 ± 1.7, 1012.1 ± 1.7, 994.3 ± 2.1, 979.1 ± 2.3,
 959.3 ± 2.2, 942.6 ± 2.0, 921.4 ± 2.2, 901.6 ± 1.9, 879.2 ± 2.1, 852.8 ± 1.6}
```

In[4195]:=
```
xAuxo = Range[6, 17];
```

In[4196]:=
```
ListPlot[Partition[Riffle[xAuxo, Rob5], {2}], Frame → True , PlotStyle →
  {RGBColor[0.34509803921568627, 0.5803921568627451, 0.6901960784313725],
   Thickness[0.004]}, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4196]=



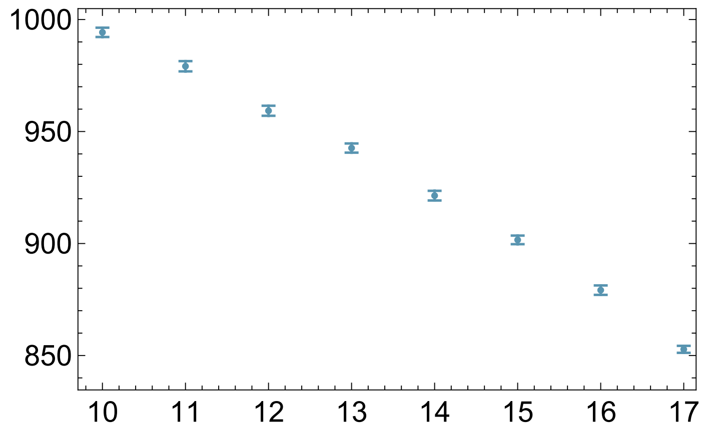
In[4197]:=
```
Length[xAuxo]
Length[Rob5]
```

Out[4197]= 12

Out[4198]= 12

In[4199]:=
```
ListPlot[Partition[Riffle[xAuxo[[5 ;; 12]], Rob5[[5 ;; 12]]], {2}],
 Frame → True , PlotStyle →
  {RGBColor[0.34509803921568627, 0.5803921568627451, 0.6901960784313725],
   Thickness[0.004]}, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4199]=

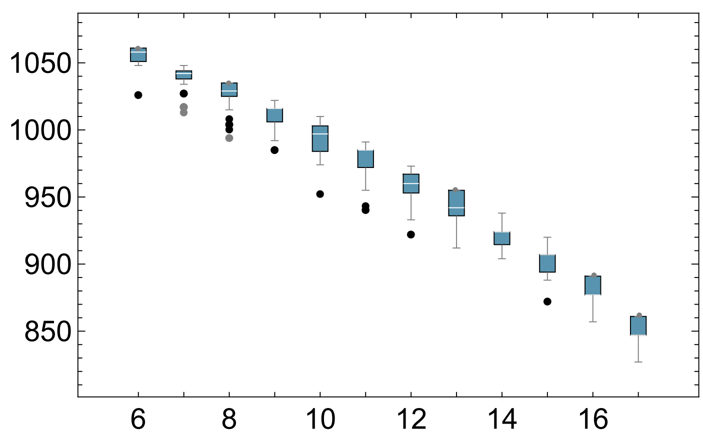


In[4200]:=
```
coco = RGBColor[0.34509803921568627, 0.5803921568627451, 0.6901960784313725]
```

Out[4200]=

In[4201]:=
```
BoxWhiskerChart[Lik, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{coco}}, Frame → True,
 ChartLabels → {"6", "", "8", "", "10", "", "12", "", "14", "", "16", ""},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4201]=



In[4152]:=
```
AuxoComm7
```

Out[4152]=
```
{1038, 1038, 1048, 1038, 1038, 1038, 1014, 1048, 1018, 1018, 1048, 1044,
 1038, 1034, 1044, 1038, 1048, 1048, 1044, 1038, 1034, 1044, 1044, 1034,
 1048, 1038, 1044, 1028, 1034, 1028, 1038, 1044, 1044, 1018, 1048, 1048,
 1048, 1044, 1038, 1044, 1048, 1044, 1034, 1044, 1044, 1048, 1034, 1048,
 1044, 1044, 1044, 1048, 1038, 1048, 1038, 1048, 1044, 1038, 1044, 1034, 1038,
 1042, 1038, 1038, 1048, 1038, 1044, 1048, 1048, 1048, 1038, 1048, 1038, 1048,
 1034, 1038, 1042, 1044, 1038, 1042, 1038, 1038, 1018, 1044, 1048, 1038, 1038,
 1038, 1042, 1044, 1044, 1028, 1048, 1028, 1038, 1048, 1044, 1018, 1038, 1038}
```

We can study the correlation between Relative entropy and assortativity with Robustness for Networks with 7 auxotrophies.

In[4153]:=
```
Entropy7 = RelatEntrop5 /@ hk7;
```

In[4154]:=
```
Assort7 = assortativity /@ hk7;
```

In[4155]:=
```
RobustNewSaito7b = AuxoComm7;
```

In[4156]:=
```
Length[Entropy7]
Length[Assort7]
Length[RobustNewSaito7b]
```

Out[4156]= 100

Out[4157]= 100

Out[4158]= 100

In[4159]:=
```
{Min[Entropy7], Max[Entropy7]}
{Min[Assort7], Max[Assort7]}
```
Out[4159]= {0.935154, 0.994118}

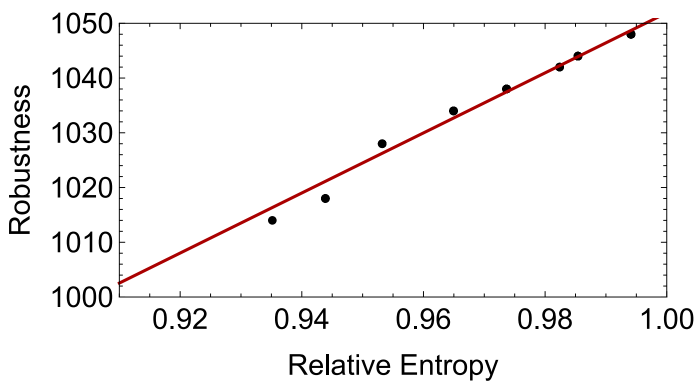
Out[4160]= {-0.416667, 0.25}

In[4161]:=
```
Position[Entropy7, Min[Entropy7]]
```
Out[4161]= {{7}}

In[4162]:=
```
RobustNewSaito7b[[#]] & /@ {1, 2, 24}
```
Out[4162]= {1038, 1038, 1034}

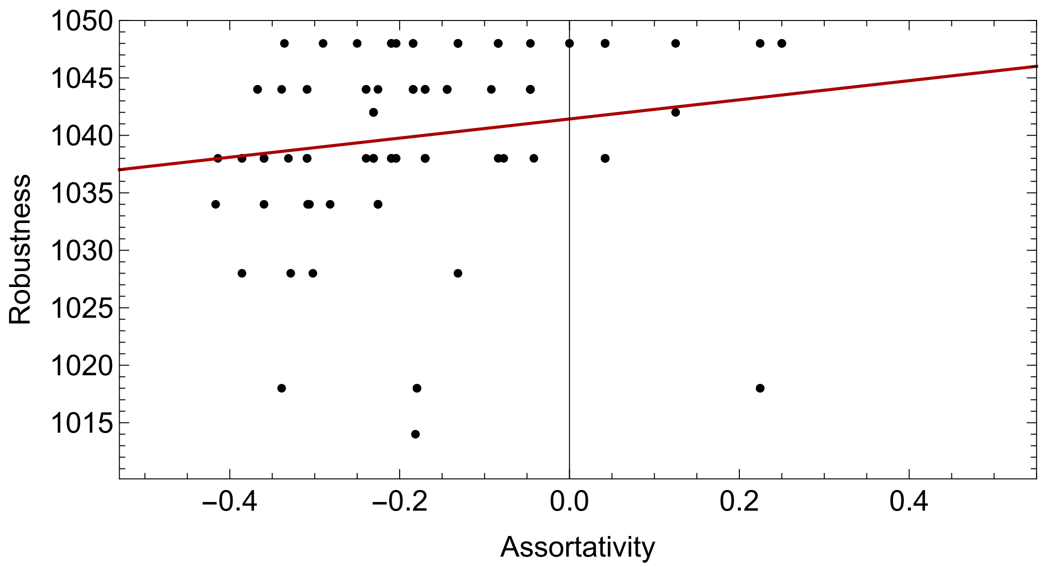
In[4163]:=
```
{Min[RobustNewSaito7b], {Max[RobustNewSaito7b]}}
```
Out[4163]= {1014, {1048}}

In[4164]:=
```
linerobustnessNewSaito25 =
  Fit[Partition[Riffle[Entropy7, RobustNewSaito7b], {2}], {1, x}, x];
Show[ListPlot[Partition[Riffle[Entropy7, RobustNewSaito7b], {2}],
  Frame → True, FrameLabel → {"Relative Entropy", "Robustness"},
  FrameStyle → Directive[Black, FontSize → 15],
  PlotStyle → {Black, PointSize[Medium]}, PlotRange → {{0.91, 1}, {1000, 1050}},
  AspectRatio → 0.5], Plot[linerobustnessNewSaito25,
  {x, 0.91, 1}, AspectRatio → 0.5, PlotStyle → Darker[Red]]]
```
Out[4165]=



In[4166]:=
```
lineAssoRobrobustnessNewSaito25 =
  Fit[Partition[Riffle[Assort7, RobustNewSaito7b], {2}], {1, x}, x];
Show[ListPlot[Partition[Riffle[Assort7, RobustNewSaito7b], {2}],
  Frame → True, FrameLabel → {"Assortativity", "Robustness"},
  FrameStyle → Directive[Black, FontSize → 15],
  PlotStyle → {Black, PointSize[Medium]}, PlotRange → {{-0.53, 0.55}, Automatic},
  AspectRatio → 0.5], Plot[lineAssoRobrobustnessNewSaito25,
  {x, -0.53, 0.55}, AspectRatio → 0.5, PlotStyle → Darker[Red]]]
```

Out[4167]=



In[4173]:=
```
SpearmanRankTest[Entropy7, RobustNewSaito7b, "TestDataTable"]
```

Out[4173]=

	Statistic	P-Value
Spearman Rank	1.	0.

In[4174]:=
```
SpearmanRankTest[Assort7, RobustNewSaito7b, "TestDataTable"]
```

Out[4174]=

	Statistic	P-Value
Spearman Rank	0.351095	0.000341587

Solving the system of ODE with Overproduction

```
fNewSaitoOVx[Net_, Dh_, coop_] := (
```

```
dB1 =
  B1[t] (-B1[t] κ1 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
       M5[t]/(denK + M5[t])) - (c1,1 + c1,2 + c1,3 + c1,4 + c1,5 + Dh) B1[t];
dB2 = B2[t] (-B2[t] κ2 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
       M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c2,1 + c2,2 + c2,3 + c2,4 + c2,5 + Dh) B2[t];
dB3 = B3[t] (-B3[t] κ3 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
       M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c3,1 + c3,2 + c3,3 + c3,4 + c3,5 + Dh) B3[t];
dB4 = B4[t] (-B4[t] κ4 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
       M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c4,1 + c4,2 + c4,3 + c4,4 + c4,5 + Dh) B4[t];
dB5 = B5[t] (-B5[t] κ5 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
       M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c5,1 + c5,2 + c5,3 + c5,4 + c5,5 + Dh) B5[t];

dM1 = -M1[t] q1 +
   (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t]))
    (-B1[t] d1,1 - B2[t] d1,2 - B3[t] d1,3 - B4[t] d1,4 - B5[t] d1,5) +
   B1[t] Ω1,1 + B2[t] Ω1,2 + B3[t] Ω1,3 + B4[t] Ω1,4 + B5[t] Ω1,5;
dM2 = -M2[t] q2 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
       M5[t]/(denK + M5[t])) (-B1[t] d2,1 - B2[t] d2,2 - B3[t] d2,3 - B4[t] d2,4 - B5[t] d2,5) +
   B1[t] Ω2,1 + B2[t] Ω2,2 + B3[t] Ω2,3 + B4[t] Ω2,4 + B5[t] Ω2,5;
dM3 = -M3[t] q3 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
       M5[t]/(denK + M5[t])) (-B1[t] d3,1 - B2[t] d3,2 - B3[t] d3,3 - B4[t] d3,4 - B5[t] d3,5) +
   B1[t] Ω3,1 + B2[t] Ω3,2 + B3[t] Ω3,3 + B4[t] Ω3,4 + B5[t] Ω3,5;
dM4 = -M4[t] q4 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
       M5[t]/(denK + M5[t])) (-B1[t] d4,1 - B2[t] d4,2 - B3[t] d4,3 - B4[t] d4,4 - B5[t] d4,5) +
   B1[t] Ω4,1 + B2[t] Ω4,2 + B3[t] Ω4,3 + B4[t] Ω4,4 + B5[t] Ω4,5;
dM5 = -M5[t] q5 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
```

```
        M₅[t]
      ─────────── ) (-B₁[t] d₅,₁ - B₂[t] d₅,₂ - B₃[t] d₅,₃ - B₄[t] d₅,₄ - B₅[t] d₅,₅) +
      denK + M₅[t]
  B₁[t] Ω₅,₁ + B₂[t] Ω₅,₂ + B₃[t] Ω₅,₃ + B₄[t] Ω₅,₄ + B₅[t] Ω₅,₅;

KK = 0.2;
cc = 0.05;
qq = 0.3;
dd = 0.00015;
OM = 1;
nu = 1500;
den = 2;


op = coop; (*Number of links with overExpression*)
posNe = Position[Net, 1];
(*Positions in the matrix where there are links (=1)*)
RaN = RandomSample[posNe, op];
(*Random sample of op links that will be overproduced*)

costincr = 1.3; (*Term multiplying the cost link*)
overprodincr = 1.15;
(*Term multiplying the overproduction link*)


NewNetCost = Net cc;
Table[NewNetCost[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] =
  NewNetCost[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] * costincr, {i, Length[RaN]}];

NewNetOvProd = Net OM;
Table[NewNetOvProd[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] =
  NewNetOvProd[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] * overprodincr, {i,
  Length[RaN]}];


tmax = 1000;
par = {
  κ₁ → KK, κ₂ → KK, κ₃ → KK, κ₄ → KK, κ₅ → KK,

  c₁,₁ → NewNetCost[[1]][[1]],
  c₁,₂ → NewNetCost[[1]][[2]], c₁,₃ → NewNetCost[[1]][[3]],
  c₁,₄ → NewNetCost[[1]][[4]], c₁,₅ → NewNetCost[[1]][[5]],
  c₂,₁ → NewNetCost[[2]][[1]], c₂,₂ → NewNetCost[[2]][[2]],
  c₂,₃ → NewNetCost[[2]][[3]], c₂,₄ → NewNetCost[[2]][[4]],
  c₂,₅ → NewNetCost[[2]][[5]],
```

```
    c3,1 → NewNetCost[[3]][[1]], c3,2 → NewNetCost[[3]][[2]],
    c3,3 → NewNetCost[[3]][[3]], c3,4 → NewNetCost[[3]][[4]],
    c3,5 → NewNetCost[[3]][[5]],
    c4,1 → NewNetCost[[4]][[1]], c4,2 → NewNetCost[[4]][[2]],
    c4,3 → NewNetCost[[4]][[3]], c4,4 → NewNetCost[[4]][[4]],
    c4,5 → NewNetCost[[4]][[5]],
    c5,1 → NewNetCost[[5]][[1]], c5,2 → NewNetCost[[5]][[2]],
    c5,3 → NewNetCost[[5]][[3]], c5,4 → NewNetCost[[5]][[4]],
    c5,5 → NewNetCost[[5]][[5]],

    q1 → qq, q2 → qq, q3 → qq, q4 → qq, q5 → qq,

    d1,1 → dd, d1,2 → dd, d1,3 → dd, d1,4 → dd, d1,5 → dd,
    d2,1 → dd, d2,2 → dd, d2,3 → dd, d2,4 → dd, d2,5 → dd,
    d3,1 → dd, d3,2 → dd, d3,3 → dd, d3,4 → dd, d3,5 → dd,
    d4,1 → dd, d4,2 → dd, d4,3 → dd, d4,4 → dd, d4,5 → dd,
    d5,1 → dd, d5,2 → dd, d5,3 → dd, d5,4 → dd, d5,5 → dd,

    Ω1,1 → NewNetOvProd[[1]][[1]],
    Ω1,2 → NewNetOvProd[[1]][[2]], Ω1,3 → NewNetOvProd[[1]][[3]],
    Ω1,4 → NewNetOvProd[[1]][[4]], Ω1,5 → NewNetOvProd[[1]][[5]],
    Ω2,1 → NewNetOvProd[[2]][[1]], Ω2,2 → NewNetOvProd[[2]][[2]],
    Ω2,3 → NewNetOvProd[[2]][[3]], Ω2,4 → NewNetOvProd[[2]][[4]],
    Ω2,5 → NewNetOvProd[[2]][[5]],
    Ω3,1 → NewNetOvProd[[3]][[1]], Ω3,2 → NewNetOvProd[[3]][[2]],
    Ω3,3 → NewNetOvProd[[3]][[3]], Ω3,4 → NewNetOvProd[[3]][[4]],
    Ω3,5 → NewNetOvProd[[3]][[5]],
    Ω4,1 → NewNetOvProd[[4]][[1]], Ω4,2 → NewNetOvProd[[4]][[2]],
    Ω4,3 → NewNetOvProd[[4]][[3]], Ω4,4 → NewNetOvProd[[4]][[4]],
    Ω4,5 → NewNetOvProd[[4]][[5]],
    Ω5,1 → NewNetOvProd[[5]][[1]], Ω5,2 → NewNetOvProd[[5]][[2]],
    Ω5,3 → NewNetOvProd[[5]][[3]], Ω5,4 → NewNetOvProd[[5]][[4]],
    Ω5,5 → NewNetOvProd[[5]][[5]],
    nuK → nu,
    denK → den


   };


  B10 = 1500;
  B20 = 1500;
  B30 = 1500;
  B40 = 1500;
```

```
B50 = 1500;
M10 = 10;
M20 = 10;
M30 = 10;
M40 = 10;
M50 = 10;


sol =
 NDSolve[
  {
    B1'[t] == dB1,
    B2'[t] == dB2,
    B3'[t] == dB3,
    B4'[t] == dB4,
    B5'[t] == dB5,

    M1'[t] == dM1,
    M2'[t] == dM2,
    M3'[t] == dM3,
    M4'[t] == dM4,
    M5'[t] == dM5,


    B1[0] == B10,
    B2[0] == B20,
    B3[0] == B30,
    B4[0] == B40,
    B5[0] == B50,
    M1[0] == M10,
    M2[0] == M20,
    M3[0] == M30,
    M4[0] == M40,
    M5[0] == M50

   } /. par,
  {B1, B2, B3, B4, B5, M1, M2, M3, M4, M5},
  {t, 0, tmax}];

{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax],
   M1[tmax], M2[tmax], M3[tmax], M4[tmax], M5[tmax]} /. sol /. par;

Min[{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax]} /. sol /. par]
```

```
)
```

In[4203]:=
```
robustnessNewSaitoOVx[NetTop_, coop_] := (
  n1 = 1;
  n2 = 5000;
  mid = (n1 + n2) / 2;

  While[(n1 ≠ mid && n2 ≠ mid ),
   (If[fNewSaitoOVx[NetTop, mid, coop] < 1, n2 = mid, n1 = mid];
    mid = Floor[N[(n1 + n2) / 2]];); {n1, n2, mid}]; mid

 )
```

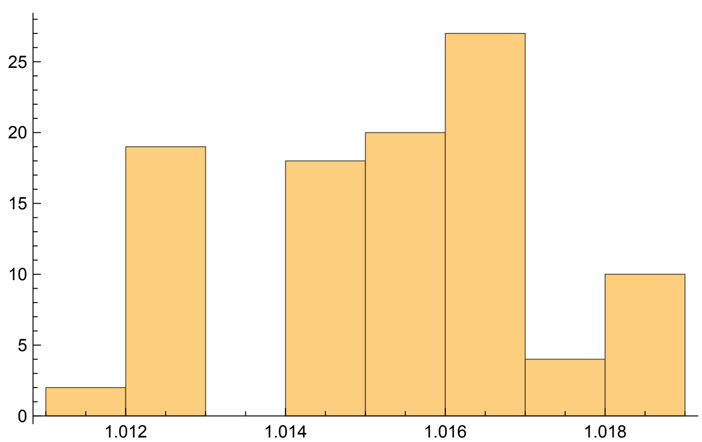
In[4205]:=
```
robustnessNewSaitoOVx[NetK, 5]
```

Out[4205]= 941

In[4210]:=
```
coopn = Table[robustnessNewSaitoOVx[NetK, 5], {100}]
```

Out[4210]=
```
{936, 935, 940, 938, 936, 936, 939, 940, 941, 938, 936, 939, 936, 939, 938, 937, 941, 939,
 938, 938, 938, 937, 935, 939, 936, 939, 937, 939, 940, 939, 941, 936, 937, 938, 937,
 937, 937, 938, 939, 939, 937, 937, 941, 939, 939, 937, 937, 936, 936, 937, 939, 939,
 939, 937, 939, 936, 936, 936, 936, 941, 938, 938, 938, 936, 937, 939, 938, 941,
 941, 939, 936, 938, 941, 939, 941, 936, 936, 938, 937, 938, 936, 937, 937, 938,
 938, 939, 939, 937, 939, 939, 939, 940, 936, 938, 938, 939, 938, 939, 939, 941}
```

In[4211]:=
```
Histogram[coopn / 924]
```

Out[4211]=


In[4212]:=
```
Select[N[coopn / 924], # ≥ 1 &] // Length
```

Out[4212]= 100

In[4213]:=
```
Mean[N[coopn / 924]]
```

Out[4213]= 1.01518

In[4214]:=
```
{Min[N[coopn / 924]], Max[N[coopn / 924]]}
```
Out[4214]=
```
{1.0119, 1.0184}
```

In[4215]:=
```
SignedRankTest[coopn / 924, 1]
```
Out[4215]= 2.67154×10^{-18}

In[4249]:=
```
coop5to15 =
  {Table[robustnessNewSaitoOVx[#, 5], {20}], Table[robustnessNewSaitoOVx[#, 10],
     {20}], Table[robustnessNewSaitoOVx[#, 15], {20}]} &;
```

In[4254]:=
```
wf6 = coop5to15 /@ hk6;
wf7 = coop5to15 /@ hk7;
wf8 = coop5to15 /@ hk8;
```

In[4235]:=
```
robustnessNewSaito[hk6[[1]]]
```
Out[4235]=
```
1058
```

In[4237]:= wf / 1058.

Out[4237]= {{1.00851, 1.00851, 1.00945, 1.00851, 1.00851, 1.00756, 1.00851, 1.00851, 1.00756,
1.00851, 1.00756, 1.00851, 1.00851, 1.00756, 1.00756, 1.00851, 1.00756, 1.00756,
1.00851, 1.00567, 1.00851, 1.00851, 1.00945, 1.00851, 1.00756, 1.00662,
1.00851, 1.00851, 1.00662, 1.00851, 1.00567, 1.00756, 1.00756, 1.00851,
1.00756, 1.00756, 1.00756, 1.00756, 1.00851, 1.00756, 1.00851, 1.00756,
1.00756, 1.00945, 1.00756, 1.00662, 1.00851, 1.00851, 1.00945, 1.00851},
{1.01512, 1.01607, 1.01512, 1.01607, 1.01607, 1.01607, 1.01512, 1.01607,
1.01607, 1.01607, 1.01701, 1.01607, 1.01607, 1.01607, 1.01607, 1.01607, 1.01701,
1.01512, 1.01607, 1.01512, 1.01607, 1.01607, 1.01512, 1.01607, 1.01607, 1.01323,
1.01701, 1.01607, 1.01512, 1.01607, 1.01701, 1.01418, 1.01607, 1.01607,
1.01512, 1.01607, 1.01701, 1.01323, 1.01607, 1.01701, 1.01607, 1.01607,
1.01607, 1.01512, 1.01607, 1.01607, 1.01607, 1.01607, 1.01607, 1.01512},
{1.02268, 1.02457, 1.02363, 1.02363, 1.02363, 1.02363, 1.02268, 1.02363,
1.02268, 1.02268, 1.02363, 1.02268, 1.02268, 1.02363, 1.02457, 1.02457, 1.02363,
1.02363, 1.02363, 1.02268, 1.02363, 1.02363, 1.02363, 1.02363, 1.02363, 1.02268,
1.02268, 1.02363, 1.02268, 1.02363, 1.02363, 1.02363, 1.02268, 1.02363,
1.02268, 1.02268, 1.02363, 1.02363, 1.02268, 1.02268, 1.02457, 1.02363,
1.02363, 1.02268, 1.02268, 1.02363, 1.02363, 1.02363, 1.02363, 1.02268}}

In[4238]:= SignedRankTest[wf[[1]] / 1058, 1]

Out[4238]= 3.85519×10^{-10}

In[4239]:= SignedRankTest[wf[[2]] / 1058, 1]

Out[4239]= 2.25707×10^{-10}

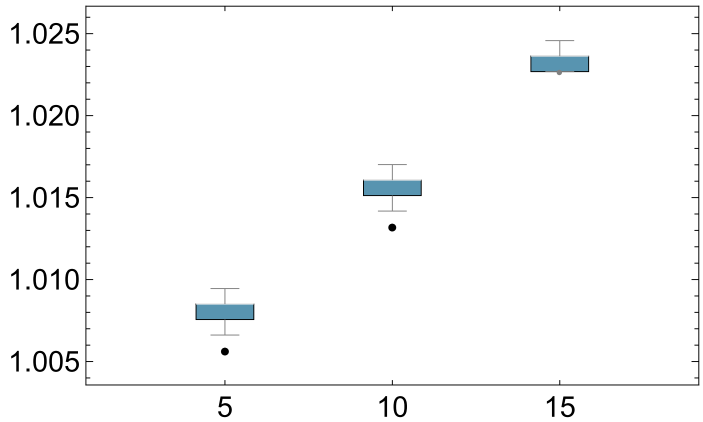
In[4240]:= SignedRankTest[wf[[3]] / 1058, 1]

Out[4240]= 2.58042×10^{-10}

In[4246]:= wf1 = Join[{ConstantArray[1, {50}]}, wf / robustnessNewSaito[hk6[[1]]]];

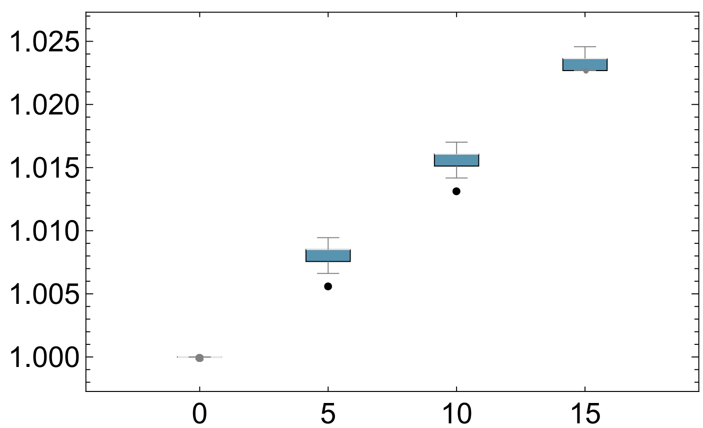
In[4241]:=
```
BoxWhiskerChart[wf / 1058., "Outliers", ChartBaseStyle → EdgeForm[Dashing[0.99]],
 ChartStyle → {{coco}}, Frame → True, ChartLabels → {"5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4241]=



In[4248]:=
```
BoxWhiskerChart[wf1, "Outliers", ChartBaseStyle → EdgeForm[Dashing[0.99]],
 ChartStyle → {{coco}}, Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4248]=



In[4257]:=
```
wf6 // Length
```

Out[4257]=
```
100
```

In[4261]:=
```
N[wf6[[1]] / AuxoComm6[[1]]]
```

Out[4261]=
```
{{1.00756, 1.00756, 1.00851, 1.00851, 1.00756, 1.00756,
  1.00662, 1.00567, 1.00756, 1.00756, 1.00851, 1.00756, 1.00851,
  1.00756, 1.00756, 1.00756, 1.00756, 1.00851, 1.00662, 1.00662},
 {1.01607, 1.01512, 1.01701, 1.01512, 1.01607, 1.01607, 1.01607,
  1.01512, 1.01607, 1.01701, 1.01512, 1.01607, 1.01512, 1.01512,
  1.01607, 1.01512, 1.01607, 1.01512, 1.01418, 1.01607},
 {1.02268, 1.02363, 1.02268, 1.02457, 1.02363, 1.02363, 1.02268,
  1.02363, 1.02363, 1.02363, 1.02457, 1.02363, 1.02363, 1.02268,
  1.02363, 1.02457, 1.02363, 1.02363, 1.02363, 1.02268}}
```

In[4263]:=
```
wf6Normalized = N[wf6[[#]] / AuxoComm6[[#]]] & /@ Range[100]
```

In[4264]:=
```
wf7Normalized = N[wf7[[#]] / AuxoComm7[[#]]] & /@ Range[100]
```

In[4265]:=
```
wf8Normalized = N[wf8[[#]] / AuxoComm8[[#]]] & /@ Range[100]
```

In[4290]:=
```
wf6Normalized // Length
```

Out[4290]=
```
100
```

In[4291]:=
```
wf6Normalized[[1]]
```

Out[4291]=
```
{{1.00756, 1.00756, 1.00851, 1.00851, 1.00756, 1.00756,
  1.00662, 1.00567, 1.00756, 1.00756, 1.00851, 1.00756, 1.00851,
  1.00756, 1.00756, 1.00756, 1.00756, 1.00851, 1.00662, 1.00662},
 {1.01607, 1.01512, 1.01701, 1.01512, 1.01607, 1.01607, 1.01607,
  1.01512, 1.01607, 1.01701, 1.01512, 1.01607, 1.01512, 1.01512,
  1.01607, 1.01512, 1.01607, 1.01512, 1.01418, 1.01607},
 {1.02268, 1.02363, 1.02268, 1.02457, 1.02363, 1.02363, 1.02268,
  1.02363, 1.02363, 1.02363, 1.02457, 1.02363, 1.02363, 1.02268,
  1.02363, 1.02457, 1.02363, 1.02363, 1.02363, 1.02268}}
```

In[4268]:=
```
wf6NormalizedWith5Coop = wf6Normalized[[#]][[1]] & /@ Range[100]
```

In[4269]:=
```
wf6NormalizedWith10Coop = wf6Normalized[[#]][[2]] & /@ Range[100]
```

In[4270]:=
```
wf6NormalizedWith15Coop = wf6Normalized[[#]][[3]] & /@ Range[100]
```

In[4271]:=
```
wf7NormalizedWith5Coop = wf7Normalized[[#]][[1]] & /@ Range[100]
```

In[4272]:=
```
wf7NormalizedWith10Coop = wf7Normalized[[#]][[2]] & /@ Range[100]
```

In[4273]:=
```
wf7NormalizedWith15Coop = wf7Normalized[[#]][[3]] & /@ Range[100]
```

In[4274]:=
```
wf8NormalizedWith5Coop = wf8Normalized[[#]][[1]] & /@ Range[100]
```

In[4275]:=
```
wf8NormalizedWith10Coop = wf8Normalized[[#]][[2]] & /@ Range[100]
```

In[4276]:=
```
wf8NormalizedWith15Coop = wf8Normalized[[#]][[3]] & /@ Range[100]
```

In[4292]:=
```
allcoopWith6Auxo = {Flatten[wf6NormalizedWith5Coop],
  Flatten[wf6NormalizedWith10Coop], Flatten[wf6NormalizedWith15Coop]}
```

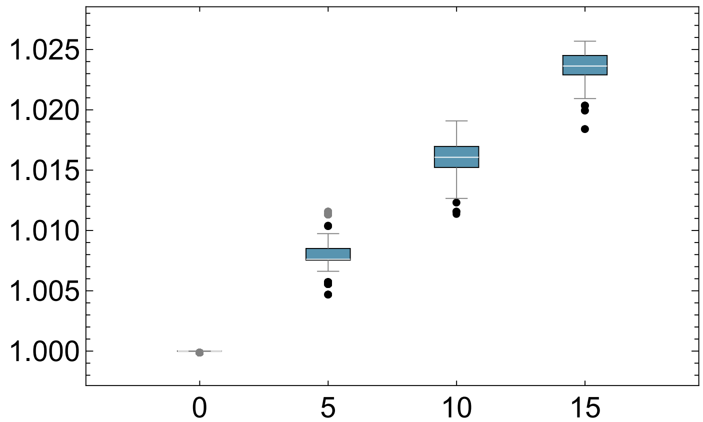
In[4293]:=
```
Length[Flatten[wf6NormalizedWith5Coop]]
```

Out[4293]= 2000

In[4294]:=
```
allcoopWith6AuxoPlusAuxo = Join[{ConstantArray[1, {2000}]}, allcoopWith6Auxo]
```

In[4295]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{coco}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4295]=



In[4296]:=
```
RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178, 0.6]
```

Out[4296]=

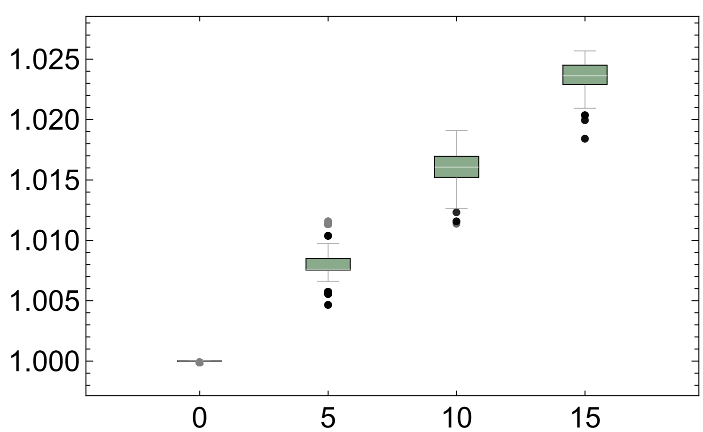
In[4297]:=
```
RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178]
```

Out[4297]=

In[4298]:=
```
gree1 =
  RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178, 0.6];
gree2 = RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178];
```

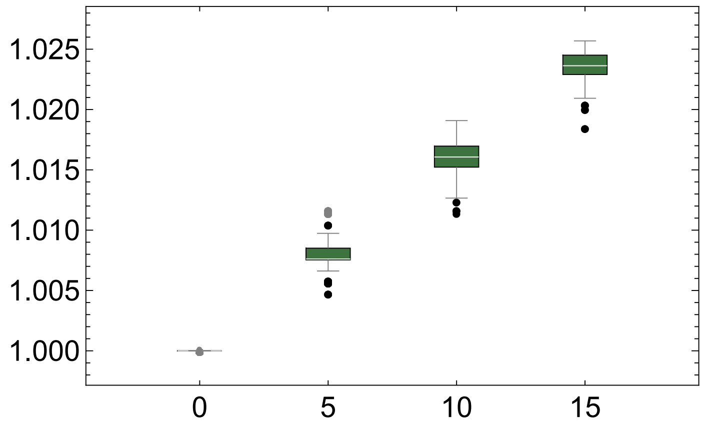
In[4300]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4300]=



In[4301]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree2}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4301]=



In[4405]:=
```
SignedRankTest[allcoopWith6AuxoPlusAuxo[[2]], 1]
SignedRankTest[allcoopWith6AuxoPlusAuxo[[3]], 1]
SignedRankTest[allcoopWith6AuxoPlusAuxo[[4]], 1]
```

Out[4405]= 0.

Out[4406]= 0.

Out[4407]= 0.

In[4282]:=
```
allcoopWith6AuxoPlusAuxo // Length
```

Out[4282]= 4

In[4284]:=
```
allcoopWith6AuxoPlusAuxo[[2]]
```

In[4285]:=
```
wf1
```

In[4287]:=
```
wf1 // Length
```

Out[4287]= 4

In[4288]:=
```
wf1[[1]] // Length
```

Out[4288]= 50

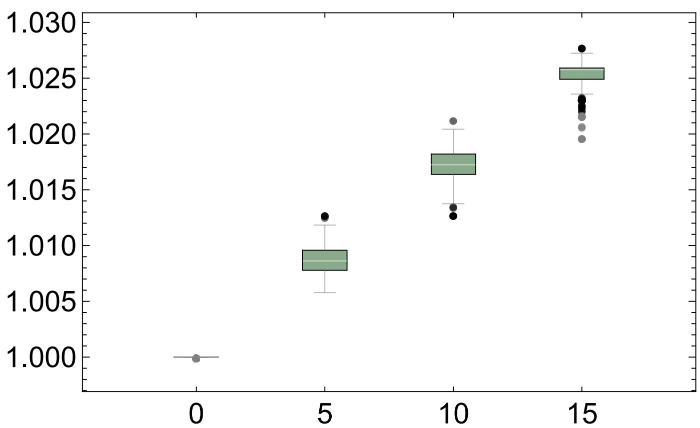
```
(*For 7 auxotrophies networks*)
```

In[4408]:=
```
allcoopWith7Auxo = {Flatten[wf7NormalizedWith5Coop],
  Flatten[wf7NormalizedWith10Coop], Flatten[wf7NormalizedWith15Coop]}
```

In[4409]:=
```
allcoopWith7AuxoPlusAuxo = Join[{ConstantArray[1, {2000}]}, allcoopWith7Auxo]
```

In[4410]:=
```
BoxWhiskerChart[allcoopWith7AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4410]=

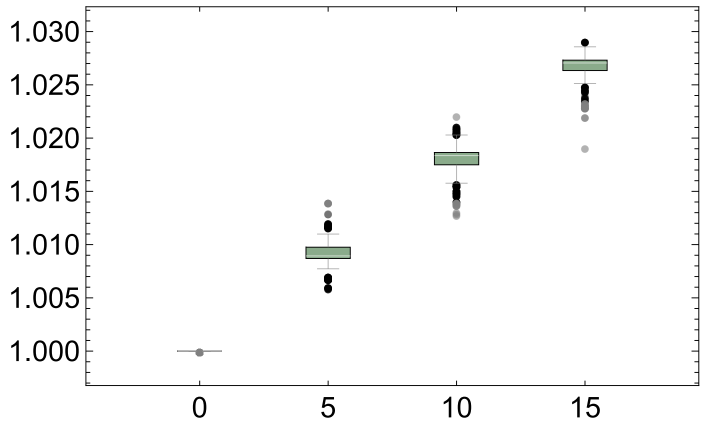



```
(*For 8 auxotrophies networks*)
```

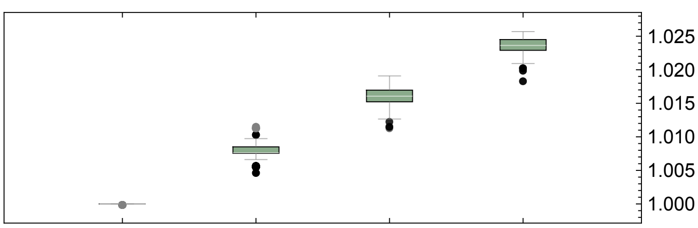
In[4411]:=
```
allcoopWith8Auxo = {Flatten[wf8NormalizedWith5Coop],
  Flatten[wf8NormalizedWith10Coop], Flatten[wf8NormalizedWith15Coop]}
```

In[4412]:=
```
allcoopWith8AuxoPlusAuxo = Join[{ConstantArray[1, {2000}]}, allcoopWith8Auxo]
```

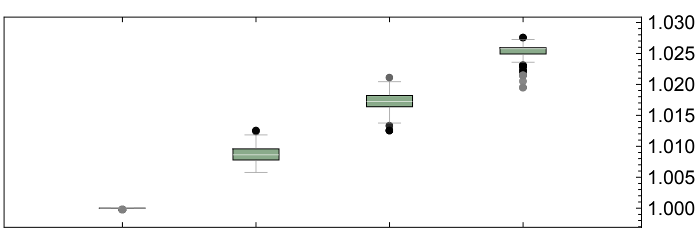
In[4413]:=
```
BoxWhiskerChart[allcoopWith8AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[4413]=


In[4448]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, FrameTicks → {{None, All}, {None, All}}, BarSpacing → 1.9,
 FrameStyle → Directive[Black, FontSize → 10], AspectRatio → 0.33]
```

Out[4448]=


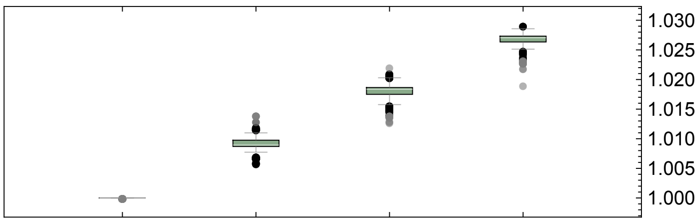
In[4449]:=
```
BoxWhiskerChart[allcoopWith7AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, FrameTicks → {{None, All}, {None, All}}, BarSpacing → 1.9,
 FrameStyle → Directive[Black, FontSize → 10], AspectRatio → 0.33]
```

Out[4449]=


In[4450]:=

```
BoxWhiskerChart[allcoopWith8AuxoPlusAuxo, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, FrameTicks → {{None, All}, {None, All}}, BarSpacing → 1.9,
 FrameStyle → Directive[Black, FontSize → 10], AspectRatio → 0.33]
```

Out[4450]=



Solving the system of ODE Random parametrization

In[5070]:=
```
Knum = 0.2;
ccrnum = 0.05;
qqrnum = 0.3;
ddrnum = 0.00015;
OMrnum = 1;
nurum = 1500;
den2rum = 2;

corrpar0 = 10^3;
corrpar1 = 10^4;
corrpar2 = 10^6;

KKr := RandomVariate[
    GammaDistribution[corrpar0 Sqrt[Knum], (1/corrpar0) Sqrt[Knum]], 1][[1]];
ccr := RandomVariate[GammaDistribution[corrpar1 Sqrt[ccrnum],
     (1/corrpar1) Sqrt[ccrnum]], 1][[1]];
qqr := RandomVariate[GammaDistribution[corrpar0 Sqrt[qqrnum],
     (1/corrpar0) Sqrt[qqrnum]], 1][[1]];
ddr := RandomVariate[GammaDistribution[corrpar1 Sqrt[ddrnum],
     (1/corrpar1) Sqrt[ddrnum]], 1][[1]];
OMr := RandomVariate[GammaDistribution[corrpar0 Sqrt[OMrnum],
     (1/corrpar0) Sqrt[OMrnum]], 1][[1]];
nur := (*nurum*) RandomVariate[GammaDistribution[
     corrpar2 Sqrt[nurum], (1/corrpar2) Sqrt[nurum]], 1][[1]] ;
denr2 := (*den2rum*) RandomVariate[GammaDistribution[
     corrpar2 Sqrt[den2rum], (1/corrpar2) Sqrt[den2rum]], 1][[1]] ;


parR = Join[Table[KKr, {5}], Table[ccr, {25}],
  Table[qqr, {5}], Table[ddr, {25}], Table[OMr, {25}], {nur}, {denr2}]
```

Out[5087]= {0.210961, 0.191637, 0.202713, 0.204961, 0.187072, 0.047671, 0.0508431, 0.0499038, 0.0504676, 0.0515537, 0.0511517, 0.0501128, 0.0490922, 0.0499356, 0.0507457, 0.0512225, 0.0481604, 0.0496869, 0.0482237, 0.0506625, 0.0509721, 0.0501808, 0.049151, 0.0485124, 0.0510165, 0.0496927, 0.0499998, 0.0526774, 0.0493969, 0.0508954, 0.286065, 0.312744, 0.324766, 0.295359, 0.302908, 0.000141744, 0.000149703, 0.000149885, 0.000146802, 0.000150636, 0.000169009, 0.000143187, 0.000155529, 0.000150391, 0.00013653, 0.000157891, 0.000176156, 0.000152199, 0.00013114, 0.000144931, 0.00014661, 0.000164497, 0.000158318, 0.000162804, 0.000120358, 0.000163537, 0.000156666, 0.000169636, 0.000146156, 0.000136045, 1.0111, 0.974423, 1.05224, 0.953066, 0.942104, 1.05838, 1.05305, 1.03825, 1.0177, 0.993813, 0.981184, 1.02204, 1.01679, 0.940398, 1.05995, 1.01498, 0.990605, 1.05853, 1.05516, 1.04118, 1.01876, 0.998902, 0.993171, 0.945556, 1.07514, 1499.76, 2.00096}

In[5032]:= **parR = %**

Out[5032]= {0.201845, 0.195413, 0.197672, 0.196705, 0.199603, 0.0482943, 0.0498595, 0.0509316, 0.0490876, 0.0506488, 0.0503456, 0.0501962, 0.049137, 0.0502639, 0.0499214, 0.0515947, 0.0507716, 0.0506371, 0.0490935, 0.0498561, 0.0489025, 0.0503342, 0.0475094, 0.0503801, 0.0509015, 0.0483202, 0.0496839, 0.0516042, 0.0505784, 0.0484774, 0.302547, 0.299913, 0.302278, 0.294801, 0.303087, 0.000162122, 0.000139135, 0.000143539, 0.00014988, 0.000145385, 0.000147672, 0.000165981, 0.000164963, 0.0001407, 0.000155689, 0.000162361, 0.000158192, 0.000150238, 0.000153067, 0.00015345, 0.000148583, 0.000172513, 0.000151723, 0.000129574, 0.000141109, 0.000159251, 0.000129223, 0.000147796, 0.000169063, 0.000163308, 0.999374, 0.977899, 0.997894, 0.997512, 1.00094, 1.00775, 0.993954, 0.99994, 0.989276, 1.00773, 0.987437, 0.989435, 1.00715, 1.0149, 1.0115, 0.990365, 1.00423, 1.00402, 1.00552, 0.992677, 0.982101, 1.00485, 0.999586, 1.00267, 1.00318, 1499.75, 2.00109}

In[5088]:=

fNewSaitoR[Net_, Dh_] := (

$$dB_1 = B_1[t]\left(-B_1[t]\,\varkappa_1 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{1,1} + c_{1,2} + c_{1,3} + c_{1,4} + c_{1,5} + Dh\right) B_1[t];$$

$$dB_2 = B_2[t]\left(-B_2[t]\,\varkappa_2 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{2,1} + c_{2,2} + c_{2,3} + c_{2,4} + c_{2,5} + Dh\right) B_2[t];$$

$$dB_3 = B_3[t]\left(-B_3[t]\,\varkappa_3 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{3,1} + c_{3,2} + c_{3,3} + c_{3,4} + c_{3,5} + Dh\right) B_3[t];$$

```
dB4 = B4[t] (-B4[t] κ4 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c4,1 + c4,2 + c4,3 + c4,4 + c4,5 + Dh) B4[t];
dB5 = B5[t] (-B5[t] κ5 + nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) *
      M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t])) - (c5,1 + c5,2 + c5,3 + c5,4 + c5,5 + Dh) B5[t];

dM1 = -M1[t] q1 +
   (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) * M5[t]/(denK + M5[t]))
    (-B1[t] d1,1 - B2[t] d1,2 - B3[t] d1,3 - B4[t] d1,4 - B5[t] d1,5) +
   B1[t] Ω1,1 + B2[t] Ω1,2 + B3[t] Ω1,3 + B4[t] Ω1,4 + B5[t] Ω1,5;
dM2 = -M2[t] q2 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d2,1 - B2[t] d2,2 - B3[t] d2,3 - B4[t] d2,4 - B5[t] d2,5) +
   B1[t] Ω2,1 + B2[t] Ω2,2 + B3[t] Ω2,3 + B4[t] Ω2,4 + B5[t] Ω2,5;
dM3 = -M3[t] q3 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d3,1 - B2[t] d3,2 - B3[t] d3,3 - B4[t] d3,4 - B5[t] d3,5) +
   B1[t] Ω3,1 + B2[t] Ω3,2 + B3[t] Ω3,3 + B4[t] Ω3,4 + B5[t] Ω3,5;
dM4 = -M4[t] q4 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d4,1 - B2[t] d4,2 - B3[t] d4,3 - B4[t] d4,4 - B5[t] d4,5) +
   B1[t] Ω4,1 + B2[t] Ω4,2 + B3[t] Ω4,3 + B4[t] Ω4,4 + B5[t] Ω4,5;
dM5 = -M5[t] q5 + (nuK * M1[t]/(denK + M1[t]) * M2[t]/(denK + M2[t]) * M3[t]/(denK + M3[t]) * M4[t]/(denK + M4[t]) *
      M5[t]/(denK + M5[t])) (-B1[t] d5,1 - B2[t] d5,2 - B3[t] d5,3 - B4[t] d5,4 - B5[t] d5,5) +
   B1[t] Ω5,1 + B2[t] Ω5,2 + B3[t] Ω5,3 + B4[t] Ω5,4 + B5[t] Ω5,5;


tmax = 1000;
par = {
   κ1 → parR[[1]], κ2 → parR[[2]], κ3 → parR[[3]], κ4 → parR[[4]], κ5 → parR[[5]],

   c1,1 → parR[[6]] × Net[[1]][[1]],
   c1,2 → parR[[7]] × Net[[1]][[2]], c1,3 → parR[[8]] × Net[[1]][[3]],
   c1,4 → parR[[9]] × Net[[1]][[4]], c1,5 → parR[[10]] × Net[[1]][[5]],
   c2,1 → parR[[11]] × Net[[2]][[1]], c2,2 → parR[[12]] × Net[[2]][[2]],
```

```
c2,3 → parR[[13]] × Net[[2]][[3]], c2,4 → parR[[14]] × Net[[2]][[4]],
c2,5 → parR[[15]] × Net[[2]][[5]],
c3,1 → parR[[16]] × Net[[3]][[1]], c3,2 → parR[[17]] × Net[[3]][[2]],
c3,3 → parR[[18]] × Net[[3]][[3]], c3,4 → parR[[19]] × Net[[3]][[4]],
c3,5 → parR[[20]] × Net[[3]][[5]],
c4,1 → parR[[21]] × Net[[4]][[1]], c4,2 → parR[[22]] × Net[[4]][[2]],
c4,3 → parR[[23]] × Net[[4]][[3]], c4,4 → parR[[24]] × Net[[4]][[4]],
c4,5 → parR[[25]] × Net[[4]][[5]],
c5,1 → parR[[26]] × Net[[5]][[1]], c5,2 → parR[[27]] × Net[[5]][[2]],
c5,3 → parR[[28]] × Net[[5]][[3]], c5,4 → parR[[29]] × Net[[5]][[4]],
c5,5 → parR[[30]] × Net[[5]][[5]],

q1 → parR[[31]], q2 → parR[[32]],
q3 → parR[[33]], q4 → parR[[34]], q5 → parR[[35]],

d1,1 → parR[[36]], d1,2 → parR[[37]],
d1,3 → parR[[38]], d1,4 → parR[[39]], d1,5 → parR[[40]],
d2,1 → parR[[41]], d2,2 → parR[[42]], d2,3 → parR[[43]],
d2,4 → parR[[44]], d2,5 → parR[[45]],
d3,1 → parR[[46]], d3,2 → parR[[47]], d3,3 → parR[[48]],
d3,4 → parR[[49]], d3,5 → parR[[50]],
d4,1 → parR[[51]], d4,2 → parR[[52]], d4,3 → parR[[53]],
d4,4 → parR[[54]], d4,5 → parR[[55]],
d5,1 → parR[[56]], d5,2 → parR[[57]], d5,3 → parR[[58]],
d5,4 → parR[[59]], d5,5 → parR[[60]],

Ω1,1 → parR[[61]] × Net[[1]][[1]],
Ω1,2 → parR[[62]] × Net[[1]][[2]], Ω1,3 → parR[[63]] × Net[[1]][[3]],
Ω1,4 → parR[[64]] × Net[[1]][[4]], Ω1,5 → parR[[65]] × Net[[1]][[5]],
Ω2,1 → parR[[66]] × Net[[2]][[1]], Ω2,2 → parR[[67]] × Net[[2]][[2]],
Ω2,3 → parR[[68]] × Net[[2]][[3]], Ω2,4 → parR[[69]] × Net[[2]][[4]],
Ω2,5 → parR[[70]] × Net[[2]][[5]],
Ω3,1 → parR[[71]] × Net[[3]][[1]], Ω3,2 → parR[[72]] × Net[[3]][[2]],
Ω3,3 → parR[[73]] × Net[[3]][[3]], Ω3,4 → parR[[74]] × Net[[3]][[4]],
Ω3,5 → parR[[75]] × Net[[3]][[5]],
Ω4,1 → parR[[76]] × Net[[4]][[1]], Ω4,2 → parR[[77]] × Net[[4]][[2]],
Ω4,3 → parR[[78]] × Net[[4]][[3]], Ω4,4 → parR[[79]] × Net[[4]][[4]],
Ω4,5 → parR[[80]] × Net[[4]][[5]],
Ω5,1 → parR[[81]] × Net[[5]][[1]], Ω5,2 → parR[[82]] × Net[[5]][[2]],
Ω5,3 → parR[[83]] × Net[[5]][[3]], Ω5,4 → parR[[84]] × Net[[5]][[4]],
Ω5,5 → parR[[85]] × Net[[5]][[5]],
nuK → parR[[86]],
denK → parR[[87]]
```

```
 };



B10 = 1500;
B20 = 1500;
B30 = 1500;
B40 = 1500;
B50 = 1500;
M10 = 10;
M20 = 10;
M30 = 10;
M40 = 10;
M50 = 10;


sol =
 NDSolve[
  {
    B1'[t] == dB1,
    B2'[t] == dB2,
    B3'[t] == dB3,
    B4'[t] == dB4,
    B5'[t] == dB5,

    M1'[t] == dM1,
    M2'[t] == dM2,
    M3'[t] == dM3,
    M4'[t] == dM4,
    M5'[t] == dM5,


    B1[0] == B10,
    B2[0] == B20,
    B3[0] == B30,
    B4[0] == B40,
    B5[0] == B50,
    M1[0] == M10,
    M2[0] == M20,
    M3[0] == M30,
    M4[0] == M40,
```

```
      M5[0] == M50

    } /. par,
   {B1, B2, B3, B4, B5, M1, M2, M3, M4, M5},
   {t, 0, tmax}];

  {B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax],
     M1[tmax], M2[tmax], M3[tmax], M4[tmax], M5[tmax]} /. sol /. par;

  Min[{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax]} /. sol /. par]

 )
```

In[5089]:=

```
robustnessNewSaitoR[NetTop_] := (
  n1 = 1;
  n2 = 5000;
  mid = (n1 + n2) / 2;

  While[(n1 ≠ mid && n2 ≠ mid ),
   (If[fNewSaitoR[NetTop, mid] < 1, n2 = mid, n1 = mid];
    mid = Floor[N[(n1 + n2) / 2]];); {n1, n2, mid}]; mid

 )
```

As an example let’s take the following Network

In[5090]:=

```
NetK = {
   {0, 1, 0, 1, 0},
   {1, 0, 1, 1, 0},
   {1, 0, 1, 0, 1},
   {0, 1, 0, 1, 0},
   {0, 0, 0, 0, 1}
  };
```

Using the function fNewSaito we can calculate the smallest value of a bacterial population in the community for a given disturbance vale. For example, let’s take Disturbance value 1 and 500:

In[5091]:= `fNewSaito[NetK, 0]`

Out[5091]= `6661.43`

In[5092]:= `fNewSaito[NetK, 500]`

Out[5092]= `4158.75`

In[5093]:=
```
fNewSaitoR[NetK, 0]
```
Out[5093]=
```
7004.98
```
In[5094]:=
```
fNewSaitoR[NetK, 500]
```
Out[5094]=
```
4622.56
```

Using the function fNewSaito we can calculate Robustness of the Network:

In[5040]:=
```
robustnessNewSaito[NetK]
```
Out[5040]=
```
924
```
In[5095]:=
```
robustnessNewSaitoR[NetK]
```
Out[5095]=
```
929
```

We can calculate the (Relative) Entropy and the Assortativity:

In[∘]:=
```
RelatEntrop5[NetK]
```
Out[∘]=
```
0.960956
```
In[∘]:=
```
assortativity[NetK]
```
Out[∘]=
```
-0.113228
```

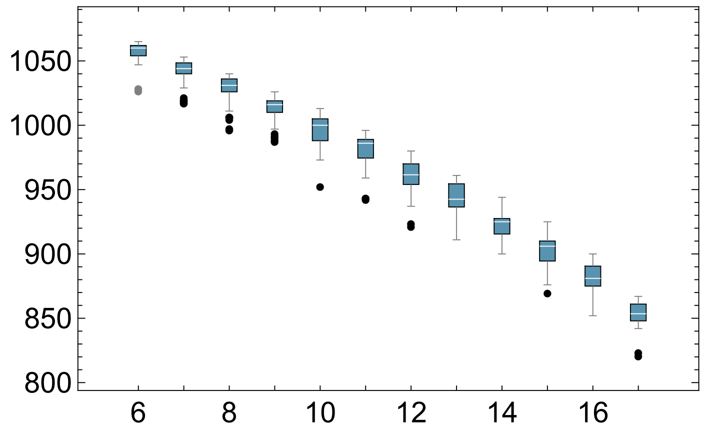
We can calculate the robustness of the previously generated random networks with different number of auxotrophies:

```
AuxoComm6R = robustnessNewSaitoR /@ hk6;
AuxoComm7R = robustnessNewSaitoR /@ hk7;
AuxoComm8R = robustnessNewSaitoR /@ hk8;
AuxoComm9R = robustnessNewSaitoR /@ hk9;
AuxoComm10R = robustnessNewSaitoR /@ hk10;
AuxoComm11R = robustnessNewSaitoR /@ hk11;
AuxoComm12R = robustnessNewSaitoR /@ hk12;
AuxoComm13R = robustnessNewSaitoR /@ hk13;
AuxoComm14R = robustnessNewSaitoR /@ hk14;
AuxoComm15R = robustnessNewSaitoR /@ hk15;
AuxoComm16R = robustnessNewSaitoR /@ hk16;
AuxoComm17R = robustnessNewSaitoR /@ hk17;
```

```
Out[5096]= {292.762, Null}

In[5097]:= 300./60

Out[5097]= 5.

In[5042]:= AuxoComm6R = robustnessNewSaitoR /@ hk6;
AuxoComm17R = robustnessNewSaitoR /@ hk17;

In[5098]:= LikR = {AuxoComm6R, AuxoComm7R, AuxoComm8R, AuxoComm9R, AuxoComm10R, AuxoComm11R,
    AuxoComm12R, AuxoComm13R, AuxoComm14R, AuxoComm15R, AuxoComm16R, AuxoComm17R};

In[•]:= coco = RGBColor[0.34509803921568627, 0.5803921568627451, 0.6901960784313725]

Out[•]=
```

In[5099]:=
```
BoxWhiskerChart[LikR, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{coco}}, Frame → True,
 ChartLabels → {"6", "", "8", "", "10", "", "12", "", "14", "", "16", ""},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[5099]=



In[5100]:=
```
AuxoComm7R
```

Out[5100]=
```
{1039, 1042, 1053, 1040, 1042, 1038, 1018, 1052, 1020, 1020, 1050, 1048,
 1042, 1035, 1046, 1041, 1050, 1050, 1046, 1043, 1037, 1049, 1045, 1034,
 1049, 1038, 1045, 1031, 1037, 1029, 1040, 1050, 1046, 1019, 1052, 1049,
 1050, 1046, 1042, 1048, 1048, 1047, 1037, 1048, 1048, 1049, 1038, 1050,
 1046, 1046, 1048, 1051, 1042, 1049, 1041, 1050, 1044, 1039, 1046, 1036, 1037,
 1044, 1041, 1040, 1052, 1040, 1046, 1049, 1051, 1049, 1043, 1051, 1042, 1050,
 1033, 1040, 1044, 1047, 1041, 1042, 1039, 1040, 1018, 1047, 1052, 1040, 1042,
 1040, 1044, 1046, 1045, 1029, 1051, 1030, 1040, 1049, 1048, 1022, 1043, 1042}
```

We can study the correlation between Relative entropy and assortativity with Robustness for Networks with 7 auxotrophies.

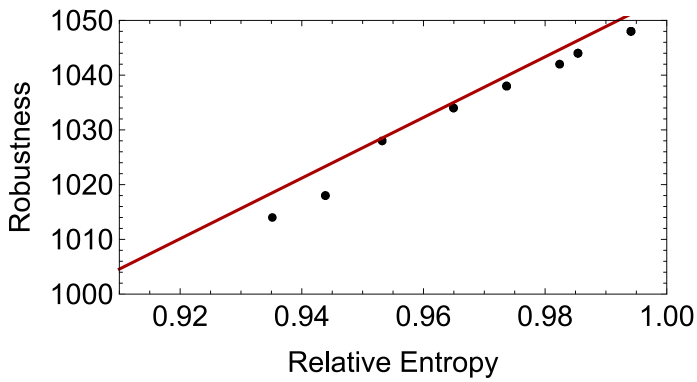
In[ ]:=
```
Entropy7 = RelatEntrop5 /@ hk7;
```

In[ ]:=
```
Assort7 = assortativity /@ hk7;
```

```
In[5101]:= RobustNewSaito7bR = AuxoComm7R;

In[5102]:= Length[Entropy7]
           Length[Assort7]
           Length[RobustNewSaito7bR]
Out[5102]= 100
Out[5103]= 100
Out[5104]= 100

In[5004]:= {Min[Entropy7], Max[Entropy7]}
           {Min[Assort7], Max[Assort7]}
Out[5004]= {0.935154, 0.994118}
Out[5005]= {-0.416667, 0.25}

In[5006]:= Position[Entropy7, Min[Entropy7]]
Out[5006]= {{7}}

In[5007]:= RobustNewSaito7bR[[#]] & /@ {1, 2, 24}
Out[5007]= {1038, 1038, 1034}

In[5105]:= {Min[RobustNewSaito7bR], {Max[RobustNewSaito7bR]}}
Out[5105]= {1018, {1053}}
```

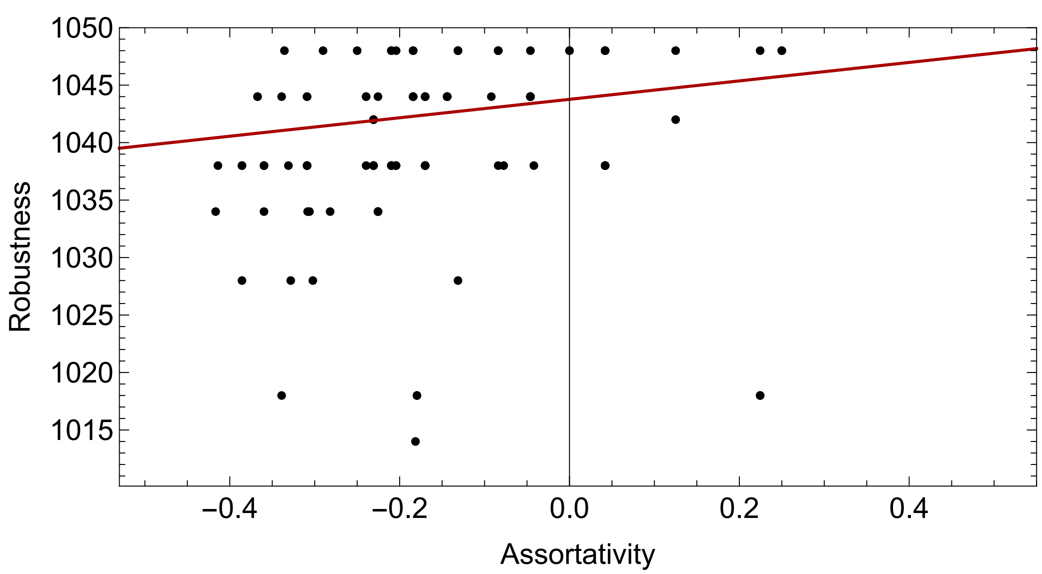
In[5106]:=
```
linerobustnessNewSaito25R =
  Fit[Partition[Riffle[Entropy7, RobustNewSaito7bR], {2}], {1, x}, x];
Show[ListPlot[Partition[Riffle[Entropy7, RobustNewSaito7b], {2}],
  Frame → True, FrameLabel → {"Relative Entropy", "Robustness"},
  FrameStyle → Directive[Black, FontSize → 15],
  PlotStyle → {Black, PointSize[Medium]}, PlotRange → {{0.91, 1}, {1000, 1050}},
  AspectRatio → 0.5], Plot[linerobustnessNewSaito25R,
  {x, 0.91, 1}, AspectRatio → 0.5, PlotStyle → Darker[Red]]]
```

Out[5107]=



In[5108]:=
```
lineAssoRobrobustnessNewSaito25R =
  Fit[Partition[Riffle[Assort7, RobustNewSaito7bR], {2}], {1, x}, x];
Show[ListPlot[Partition[Riffle[Assort7, RobustNewSaito7b], {2}],
  Frame → True, FrameLabel → {"Assortativity", "Robustness"},
  FrameStyle → Directive[Black, FontSize → 15],
  PlotStyle → {Black, PointSize[Medium]}, PlotRange → {{-0.53, 0.55}, Automatic},
  AspectRatio → 0.5], Plot[lineAssoRobrobustnessNewSaito25R,
  {x, -0.53, 0.55}, AspectRatio → 0.5, PlotStyle → Darker[Red]]]
```

Out[5109]=



In[5110]:=
```
SpearmanRankTest[Entropy7, RobustNewSaito7bR, "TestDataTable"]
```

Out[5110]=

	Statistic	P-Value
Spearman Rank	0.965653	3.63312×10^{-59}

In[5111]:=
```
SpearmanRankTest[Assort7, RobustNewSaito7bR, "TestDataTable"]
```

Out[5111]=

	Statistic	P-Value
Spearman Rank	0.343978	0.000458145

In[5112]:=
```
parR // Length
```

Out[5112]= 87

Solving the system of ODE with Overproduction Random parametrization

In[5114]:=
```
fNewSaitoOVR[Net_, Dh_, coop_] := (
```

$$dB_1 = B_1[t]\left(-B_1[t]\,\kappa_1 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{1,1} + c_{1,2} + c_{1,3} + c_{1,4} + c_{1,5} + Dh\right) B_1[t];$$

$$dB_2 = B_2[t]\left(-B_2[t]\,\kappa_2 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{2,1} + c_{2,2} + c_{2,3} + c_{2,4} + c_{2,5} + Dh\right) B_2[t];$$

$$dB_3 = B_3[t]\left(-B_3[t]\,\kappa_3 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{3,1} + c_{3,2} + c_{3,3} + c_{3,4} + c_{3,5} + Dh\right) B_3[t];$$

$$dB_4 = B_4[t]\left(-B_4[t]\,\kappa_4 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{4,1} + c_{4,2} + c_{4,3} + c_{4,4} + c_{4,5} + Dh\right) B_4[t];$$

$$dB_5 = B_5[t]\left(-B_5[t]\,\kappa_5 + nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) - \left(c_{5,1} + c_{5,2} + c_{5,3} + c_{5,4} + c_{5,5} + Dh\right) B_5[t];$$

$$dM_1 = -M_1[t]\, q_1 + \left(nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) \left(-B_1[t]\, d_{1,1} - B_2[t]\, d_{1,2} - B_3[t]\, d_{1,3} - B_4[t]\, d_{1,4} - B_5[t]\, d_{1,5}\right) + B_1[t]\, \Omega_{1,1} + B_2[t]\, \Omega_{1,2} + B_3[t]\, \Omega_{1,3} + B_4[t]\, \Omega_{1,4} + B_5[t]\, \Omega_{1,5};$$

$$dM_2 = -M_2[t]\, q_2 + \left(nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) \left(-B_1[t]\, d_{2,1} - B_2[t]\, d_{2,2} - B_3[t]\, d_{2,3} - B_4[t]\, d_{2,4} - B_5[t]\, d_{2,5}\right) + B_1[t]\, \Omega_{2,1} + B_2[t]\, \Omega_{2,2} + B_3[t]\, \Omega_{2,3} + B_4[t]\, \Omega_{2,4} + B_5[t]\, \Omega_{2,5};$$

$$dM_3 = -M_3[t]\, q_3 + \left(nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) \left(-B_1[t]\, d_{3,1} - B_2[t]\, d_{3,2} - B_3[t]\, d_{3,3} - B_4[t]\, d_{3,4} - B_5[t]\, d_{3,5}\right) + B_1[t]\, \Omega_{3,1} + B_2[t]\, \Omega_{3,2} + B_3[t]\, \Omega_{3,3} + B_4[t]\, \Omega_{3,4} + B_5[t]\, \Omega_{3,5};$$

$$dM_4 = -M_4[t]\, q_4 + \left(nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) \left(-B_1[t]\, d_{4,1} - B_2[t]\, d_{4,2} - B_3[t]\, d_{4,3} - B_4[t]\, d_{4,4} - B_5[t]\, d_{4,5}\right) + B_1[t]\, \Omega_{4,1} + B_2[t]\, \Omega_{4,2} + B_3[t]\, \Omega_{4,3} + B_4[t]\, \Omega_{4,4} + B_5[t]\, \Omega_{4,5};$$

$$dM_5 = -M_5[t]\, q_5 + \left(nuK * \frac{M_1[t]}{denK + M_1[t]} * \frac{M_2[t]}{denK + M_2[t]} * \frac{M_3[t]}{denK + M_3[t]} * \frac{M_4[t]}{denK + M_4[t]} * \frac{M_5[t]}{denK + M_5[t]}\right) \left(-B_1[t]\, d_{5,1} - B_2[t]\, d_{5,2} - B_3[t]\, d_{5,3} - B_4[t]\, d_{5,4} - B_5[t]\, d_{5,5}\right) + B_1[t]\, \Omega_{5,1} + B_2[t]\, \Omega_{5,2} + B_3[t]\, \Omega_{5,3} + B_4[t]\, \Omega_{5,4} + B_5[t]\, \Omega_{5,5};$$

```
op = coop; (*Number of links with overExpression*)
posNe = Position[Net, 1];
(*Positions in the matrix where there are links (=1)*)
RaN = RandomSample[posNe, op];
(*Random sample of op links that will be overproduced*)

costincr = 1.3; (*Term multiplying the cost link*)
overprodincr = 1.15;
(*Term multiplying the overproduction link*)


NewNetCost = Partition[Flatten[Net] × parR[[6 ;; 30]], {5}];
Table[NewNetCost[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] =
  NewNetCost[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] * costincr, {i, Length[RaN]}];

NewNetOvProd = Partition[Flatten[Net] × parR[[61 ;; 85]], {5}];
Table[NewNetOvProd[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] =
```

```
   NewNetOvProd[[RaN[[i]][[1]]]][[RaN[[i]][[2]]]] * overprodincr, {i,
   Length[RaN]}];


 tmax = 1000;
 par = {
   κ1 → parR[[1]], κ2 → parR[[2]], κ3 → parR[[3]], κ4 → parR[[4]], κ5 → parR[[5]],

   c1,1 → NewNetCost[[1]][[1]],
   c1,2 → NewNetCost[[1]][[2]], c1,3 → NewNetCost[[1]][[3]],
   c1,4 → NewNetCost[[1]][[4]], c1,5 → NewNetCost[[1]][[5]],
   c2,1 → NewNetCost[[2]][[1]], c2,2 → NewNetCost[[2]][[2]],
   c2,3 → NewNetCost[[2]][[3]], c2,4 → NewNetCost[[2]][[4]],
   c2,5 → NewNetCost[[2]][[5]],
   c3,1 → NewNetCost[[3]][[1]], c3,2 → NewNetCost[[3]][[2]],
   c3,3 → NewNetCost[[3]][[3]], c3,4 → NewNetCost[[3]][[4]],
   c3,5 → NewNetCost[[3]][[5]],
   c4,1 → NewNetCost[[4]][[1]], c4,2 → NewNetCost[[4]][[2]],
   c4,3 → NewNetCost[[4]][[3]], c4,4 → NewNetCost[[4]][[4]],
   c4,5 → NewNetCost[[4]][[5]],
   c5,1 → NewNetCost[[5]][[1]], c5,2 → NewNetCost[[5]][[2]],
   c5,3 → NewNetCost[[5]][[3]], c5,4 → NewNetCost[[5]][[4]],
   c5,5 → NewNetCost[[5]][[5]],

   r1,1 → parR[[31]], r1,2 → parR[[32]],
   r1,3 → parR[[33]], r1,4 → parR[[34]], r1,5 → parR[[35]],
   r2,1 → parR[[36]], r2,2 → parR[[37]], r2,3 → parR[[38]],
   r2,4 → parR[[39]], r2,5 → parR[[40]],
   r3,1 → parR[[41]], r3,2 → parR[[42]], r3,3 → parR[[43]],
   r3,4 → parR[[44]], r3,5 → parR[[45]],
   r4,1 → parR[[46]], r4,2 → parR[[47]], r4,3 → parR[[48]],
   r4,4 → parR[[49]], r4,5 → parR[[50]],
   r5,1 → parR[[51]], r5,2 → parR[[52]], r5,3 → parR[[53]],
   r5,4 → parR[[54]], r5,5 → parR[[55]],

   q1 → parR[[31]], q2 → parR[[32]],
   q3 → parR[[33]], q4 → parR[[34]], q5 → parR[[35]],

   d1,1 → parR[[36]], d1,2 → parR[[37]],
   d1,3 → parR[[38]], d1,4 → parR[[39]], d1,5 → parR[[40]],
   d2,1 → parR[[41]], d2,2 → parR[[42]], d2,3 → parR[[43]],
   d2,4 → parR[[44]], d2,5 → parR[[45]],
   d3,1 → parR[[46]], d3,2 → parR[[47]], d3,3 → parR[[48]],
   d3,4 → parR[[49]], d3,5 → parR[[50]],
```

```
  d4,1 → parR[[51]], d4,2 → parR[[52]], d4,3 → parR[[53]],
  d4,4 → parR[[54]], d4,5 → parR[[55]],
  d5,1 → parR[[56]], d5,2 → parR[[57]], d5,3 → parR[[58]],
  d5,4 → parR[[59]], d5,5 → parR[[60]],

  Ω1,1 → NewNetOvProd[[1]][[1]],
  Ω1,2 → NewNetOvProd[[1]][[2]], Ω1,3 → NewNetOvProd[[1]][[3]],
  Ω1,4 → NewNetOvProd[[1]][[4]], Ω1,5 → NewNetOvProd[[1]][[5]],
  Ω2,1 → NewNetOvProd[[2]][[1]], Ω2,2 → NewNetOvProd[[2]][[2]],
  Ω2,3 → NewNetOvProd[[2]][[3]], Ω2,4 → NewNetOvProd[[2]][[4]],
  Ω2,5 → NewNetOvProd[[2]][[5]],
  Ω3,1 → NewNetOvProd[[3]][[1]], Ω3,2 → NewNetOvProd[[3]][[2]],
  Ω3,3 → NewNetOvProd[[3]][[3]], Ω3,4 → NewNetOvProd[[3]][[4]],
  Ω3,5 → NewNetOvProd[[3]][[5]],
  Ω4,1 → NewNetOvProd[[4]][[1]], Ω4,2 → NewNetOvProd[[4]][[2]],
  Ω4,3 → NewNetOvProd[[4]][[3]], Ω4,4 → NewNetOvProd[[4]][[4]],
  Ω4,5 → NewNetOvProd[[4]][[5]],
  Ω5,1 → NewNetOvProd[[5]][[1]], Ω5,2 → NewNetOvProd[[5]][[2]],
  Ω5,3 → NewNetOvProd[[5]][[3]], Ω5,4 → NewNetOvProd[[5]][[4]],
  Ω5,5 → NewNetOvProd[[5]][[5]],
  nuK → parR[[86]],
  denK → parR[[87]]


 };


B10 = 1500;
B20 = 1500;
B30 = 1500;
B40 = 1500;
B50 = 1500;
M10 = 10;
M20 = 10;
M30 = 10;
M40 = 10;
M50 = 10;


sol =
 NDSolve[
  {
    B1'[t] == dB1,
    B2'[t] == dB2,
```

```
      B3'[t] == dB3,
      B4'[t] == dB4,
      B5'[t] == dB5,

      M1'[t] == dM1,
      M2'[t] == dM2,
      M3'[t] == dM3,
      M4'[t] == dM4,
      M5'[t] == dM5,


      B1[0] == B10,
      B2[0] == B20,
      B3[0] == B30,
      B4[0] == B40,
      B5[0] == B50,
      M1[0] == M10,
      M2[0] == M20,
      M3[0] == M30,
      M4[0] == M40,
      M5[0] == M50

     } /. par,
    {B1, B2, B3, B4, B5, M1, M2, M3, M4, M5},
    {t, 0, tmax}];

  {B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax],
     M1[tmax], M2[tmax], M3[tmax], M4[tmax], M5[tmax]} /. sol /. par;

  Min[{B1[tmax], B2[tmax], B3[tmax], B4[tmax], B5[tmax]} /. sol /. par]

 )
```

In[5115]:=
```
robustnessNewSaitoOVR[NetTop_, coop_] := (
  n1 = 1;
  n2 = 5000;
  mid = (n1 + n2) / 2;

  While[(n1 ≠ mid && n2 ≠ mid ),
   (If[fNewSaitoOVR[NetTop, mid, coop] < 1, n2 = mid, n1 = mid];
    mid = Floor[N[(n1 + n2) / 2]];); {n1, n2, mid}]; mid

 )
```

In[5116]:=
```
robustnessNewSaitoOVR[NetK, 5]
```

Out[5116]=
```
940
```

In[5117]:=
```
coop5to15R =
  {Table[robustnessNewSaitoOVR[#, 5], {20}], Table[robustnessNewSaitoOVR[#, 10],
     {20}], Table[robustnessNewSaitoOVR[#, 15], {20}]} &;
```

In[5118]:=
```
wf6R = coop5to15R /@ hk6;

wf8R = coop5to15R /@ hk8;
```

In[5120]:=
```
wf6NormalizedR = N[wf6R[[#]] / AuxoComm6R[[#]]] & /@ Range[100]
```

In[5121]:=
```
wf8NormalizedR = N[wf8R[[#]] / AuxoComm8R[[#]]] & /@ Range[100]
```

In[5122]:=
```
wf6NormalizedWith5CoopR = wf6NormalizedR[[#]][[1]] & /@ Range[100]
```

In[5126]:=
```
wf6NormalizedWith10CoopR = wf6NormalizedR[[#]][[2]] & /@ Range[100]
```

In[5123]:=
```
wf6NormalizedWith15CoopR = wf6NormalizedR[[#]][[3]] & /@ Range[100]
```

In[5124]:=
```
wf8NormalizedWith5CoopR = wf8NormalizedR[[#]][[1]] & /@ Range[100]
```

In[5127]:=
```
wf8NormalizedWith10CoopR = wf8NormalizedR[[#]][[2]] & /@ Range[100]
```

In[5125]:=
```
wf8NormalizedWith15CoopR = wf8NormalizedR[[#]][[3]] & /@ Range[100]
```

In[5128]:=
```
allcoopWith6AuxoR = {Flatten[wf6NormalizedWith5CoopR],
  Flatten[wf6NormalizedWith10CoopR], Flatten[wf6NormalizedWith15CoopR]}
```

In[5129]:=
```
allcoopWith6AuxoPlusAuxoR = Join[{ConstantArray[1, {2000}]}, allcoopWith6AuxoR]
```

In[∘]:=
```
RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178, 0.6]
```

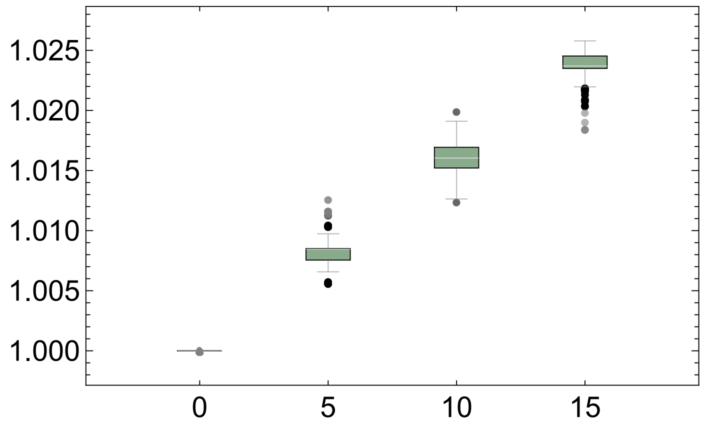
Out[∘]=

In[∘]:=
```
RGBColor[0.23921568627450981, 0.45098039215686275, 0.24705882352941178]
```

Out[∘]=

In[5130]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxoR, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[5130]=



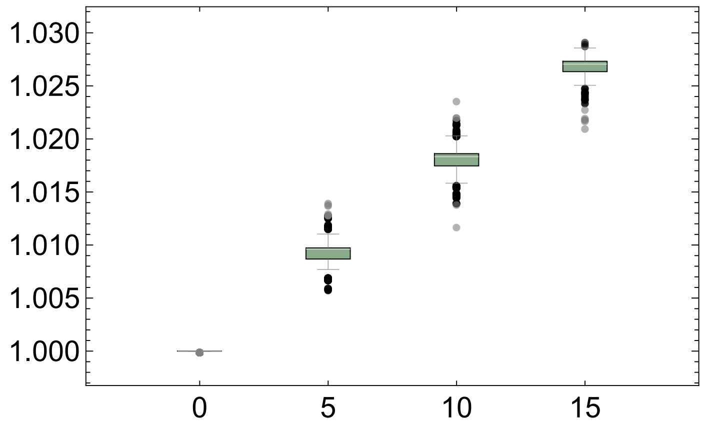

```
(*For 8 auxotrophies networks*)
```

In[5131]:=
```
allcoopWith8AuxoR = {Flatten[wf8NormalizedWith5CoopR],
  Flatten[wf8NormalizedWith10CoopR], Flatten[wf8NormalizedWith15CoopR]}
```

In[5132]:=
```
allcoopWith8AuxoPlusAuxoR = Join[{ConstantArray[1, {2000}]}, allcoopWith8AuxoR]
```

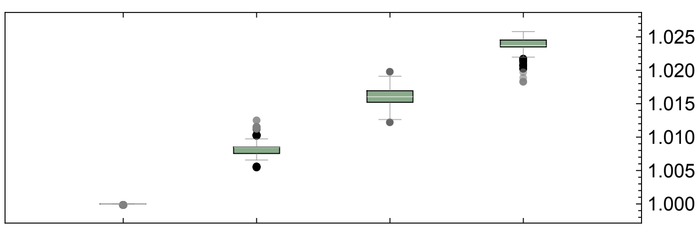
In[5133]:=
```
BoxWhiskerChart[allcoopWith8AuxoPlusAuxoR, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, ChartLabels → {"0", "5", "10", "15"},
 BarSpacing → 1.9, FrameStyle → Directive[Black, FontSize → 15]]
```

Out[5133]=



In[5134]:=
```
BoxWhiskerChart[allcoopWith6AuxoPlusAuxoR, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, FrameTicks → {{None, All}, {None, All}}, BarSpacing → 1.9,
 FrameStyle → Directive[Black, FontSize → 10], AspectRatio → 0.33]
```

Out[5134]=



In[5135]:=
```
BoxWhiskerChart[allcoopWith8AuxoPlusAuxoR, "Outliers",
 ChartBaseStyle → EdgeForm[Dashing[0.99]], ChartStyle → {{gree1}},
 Frame → True, FrameTicks → {{None, All}, {None, All}}, BarSpacing → 1.9,
 FrameStyle → Directive[Black, FontSize → 10], AspectRatio → 0.33]
```

Out[5135]=