A flux-based approach for analyzing the disguised toric locus of reaction networks

Balázs Boros[1,2], Gheorghe Craciun[3,4], Oskar Henriksson[5], Jiaxin Jin[6], Diego Rojas La Luz[3]

1 Bolyai Institute, University of Szeged, Hungary
2 National Laboratory for Health Security, University of Szeged, Hungary
3 Department of Mathematics, University of Wisconsin-Madison, United States
4 Department of Biomolecular Chemistry, University of Wisconsin-Madison, United States
5 Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany
6 Department of Mathematics, University of Louisiana at Lafayette, United States

This file contains the calculations that appear in Section 7 in the paper titled "A flux-based approach for analyzing the disguised toric locus of reaction networks."

Modules

We collect some modules that are used below.

In[14]:=
```
GetMassAction[reactions_] :=
  Module[{n, G, rates, flux, vars, srcs, tgts, monomials, f},
   G = Graph[reactions];
   n = Length[reactions[[1, 1]]];
   rates = Table[κi, {i, 1, Length[reactions]}];
   flux = Table[βi, {i, 1, Length[reactions]}];
   vars = Switch[n, 2, {x, y}, 3, {x, y, z}, 4, {x, y, z, w}, _, Table[xi, {i, 1, n}]];
   srcs = EdgeList[G][[All, 1]];
   tgts = EdgeList[G][[All, 2]];
   monomials = rates * Apply[Times, vars^(srcs^T)];
   f = Total[monomials * (tgts - srcs)];
   {f, rates, G, monomials, vars, flux}
  ];

GetEQ[reactions_] := Module[{f, vars, all1, EQUIL, i},
   {f, vars} = GetMassAction[reactions][[{1, 5}]];
   all1 = Table[vars[[i]] → 1, {i, 1, Length[vars]}];
   EQUIL = (f /. {κ → β} /. all1) == 0;
   EQUIL
  ];

GetFlux[reactions_] := Module[{},
   GetMassAction[reactions][[6]]
  ];

GetDEandVB[rxnG_, rxnH_] := Module[{f1, vars1, f2, rates2, G2, flux2, DE, VB},
   {f1, vars1} = GetMassAction[rxnG][[{1, 5}]];
   {f2, rates2, G2} = GetMassAction[rxnH][[{1, 2, 3}]];
   flux2 = rates2 /. {κ → γ};
   DE = CoefficientList[(f1 /. {κ → β}) - (f2 /. {κ → γ}), vars1] == 0;
   VB = IncidenceMatrix[G2].flux2 == 0;
   {DE, VB}
  ];

Getβ2κ[reactions_] := Module[{flux, monomials, β2κ, vars, κs, i},
   flux = GetFlux[reactions];
   {κs, monomials, vars} = GetMassAction[reactions][[{2, 4, 5}]];
   β2κ = Table[flux[[i]] → monomials[[i]], {i, 1, Length[reactions]}];
   {β2κ, κs, vars}
  ];

DrawEGraph[rxn_] := Module[{}, {
     Graph[rxn,
      VertexCoordinates → Table[i → i, {i, VertexList[rxn]}], ImageSize → Tiny]
    }];
```

Examples

Example 2.17

We verify that the Horn-Jackson function is indeed not always a global Lyapunov function for the partly reversible square.
We find that the value of the Horn-Jackson can both increase and decrease along trajectories in any neighborhood of the equilibrium, provided κ is chosen accordingly.
The equilibrium is a saddle of the function $x \mapsto \langle \text{grad } L(x), f(x) \rangle$ for those specific rate constants.

In[20]:=

```
gradL = Log[{x₁, x₂}];
f = {κ₁ (1 - x₁ x₂), κ₂ (x₁ - x₂) + κ₁ (1 - x₁ x₂)};
H = D[gradL.f, {{x₁, x₂}, 2}] /. {x₁ → 1, x₂ → 1};
Print["Hessian matrix: ", MatrixForm[H]];
Print["determinant of the Hessian matrix is negative for ",
  Reduce[Det[H] < 0 && κ₁ > 0 && κ₂ > 0]];
```

Hessian matrix: $\begin{pmatrix} -2\kappa_1 & -2\kappa_1+\kappa_2 \\ -2\kappa_1+\kappa_2 & 2(-\kappa_1-\kappa_2) \end{pmatrix}$

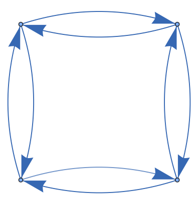
determinant of the Hessian matrix is negative for $\kappa_1 > 0$ && $\kappa_2 > 8\kappa_1$

7.1 Reversible square

In[25]:=

```
rxnG = {{0, 0} → {1, 0}, {1, 0} → {1, 1}, {1, 1} → {0, 1}, {0, 1} → {0, 0},
    {1, 0} → {0, 0}, {1, 1} → {1, 0}, {0, 1} → {1, 1}, {0, 0} → {0, 1}};
DrawEGraph[rxnG][[1]]
```

Out[26]=



Equilibrium fluxes

In[27]:=

```
EQ = GetEQ[rxnG]
```

Out[27]=

$\{\beta_1 - \beta_3 - \beta_5 + \beta_7, \beta_2 - \beta_4 - \beta_6 + \beta_8\} == 0$

Disguised toric flux cone

In[28]:=
```
rxnH = Join[rxnG, {{0, 0} → {1, 1}, {1, 1} → {0, 0}, {1, 0} → {0, 1}, {0, 1} → {1, 0}}];
(* H=G^max=G^comp *)
{DE, VB} = GetDEandVB[rxnG, rxnH];
sol = Solve[DE && VB, {β1, β8, γ1, γ2, γ3, γ4, γ5, γ6, γ7, γ8, γ9}][[1]];
MatrixForm[sol]
```

Out[31]//MatrixForm=

$$\begin{pmatrix}
\beta_1 \to \beta_3 + \beta_5 - \beta_7 \\
\beta_8 \to -\beta_2 + \beta_4 + \beta_6 \\
\gamma_1 \to \beta_2 + \beta_5 - \beta_6 + \gamma_{10} - \gamma_{11} - \gamma_{12} \\
\gamma_2 \to \beta_2 - \gamma_{11} \\
\gamma_3 \to \beta_3 - \gamma_{10} \\
\gamma_4 \to \beta_4 - \gamma_{12} \\
\gamma_5 \to \beta_5 - \gamma_{11} \\
\gamma_6 \to \beta_6 - \gamma_{10} \\
\gamma_7 \to \beta_7 - \gamma_{12} \\
\gamma_8 \to -\beta_3 + \beta_4 + \beta_7 + \gamma_{10} - \gamma_{11} - \gamma_{12} \\
\gamma_9 \to -\beta_2 + \beta_3 + \beta_6 - \beta_7 - \gamma_{10} + \gamma_{11} + \gamma_{12}
\end{pmatrix}$$

Next, we verify that the simplified formula is indeed equivalent to what we originally got.

In[32]:=
```
flux = GetFlux[rxnG];
disgβ1 = Max[β1, β8] - β4 - β5 ≤ Min[β3, β6] &&
    -Min[β2, β5] - Min[β4, β7] ≤ β1 - β5 - β2 + β6 && EQ && flux > 0;
disgβ2 = (β1 - β8) (β3 - β6) ≤ (β2 + β5) (β4 + β7) &&
    (β2 - β5) (β4 - β7) ≤ (β1 + β8) (β3 + β6) && EQ && flux > 0;
Reduce[disgβ1 && Not[disgβ2]]
Reduce[Not[disgβ1] && disgβ2]
```

Out[35]=
```
False
```

Out[36]=
```
False
```

Disguised toric locus

How much of the simplex is covered by the disguised toric locus? We get approximately **83.3%** (1 million simulations take about 25 seconds.)

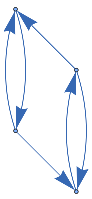
In[37]:=
```
M = 1000000;
count = 0;
disgκ = (κ1 - κ8) (κ3 - κ6) ≤ (κ2 + κ5) (κ4 + κ7) && (κ2 - κ5) (κ4 - κ7) ≤ (κ1 + κ8) (κ3 + κ6);
For[i = 1, i ≤ M, i++, {
    rnd = RandomVariate[ExponentialDistribution[1], 8];
    If[disgκ /. Table[κj → rnd[[j]], {j, 1, 8}], count++];
   }];
Print[N[100 count / M, 3], "% of the simplex is covered by K^dt (G)"];
```

83.3% of the simplex is covered by $K^{dt}(G)$

7.2 Square vs. parallelogram

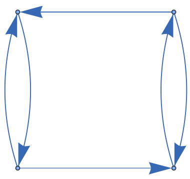
In[42]:=
```
rxnG = {{0, 1} → {1, 0}, {1, 0} → {1, 2},
    {1, 2} → {0, 3}, {0, 3} → {0, 1}, {1, 2} → {1, 0}, {0, 1} → {0, 3}};
DrawEGraph[rxnG][[1]]
```

Out[43]=



In[44]:=
```
rxnG = {{0, 0} → {1, 0}, {1, 0} → {1, 1},
    {1, 1} → {0, 1}, {0, 1} → {0, 0}, {1, 1} → {1, 0}, {0, 0} → {0, 1}};
DrawEGraph[rxnG][[1]]
```

Out[45]=



Equilibrium fluxes

In[46]:=
```
EQ = GetEQ[rxnG]
```

Out[46]=

$\{\beta_1 - \beta_3, \beta_2 - \beta_4 - \beta_5 + \beta_6\} == 0$

Disguised toric flux cone

In[47]:=
```
rxnH = Join[rxnG, {{0, 0} → {1, 1}, {1, 1} → {0, 0}}];
{DE, VB} = GetDEandVB[rxnG, rxnH];
sol = Solve[DE && VB, {β2, β3, γ1, γ2, γ3, γ4, γ5, γ6, γ7}][[1]];
MatrixForm[sol]
```

Out[50]//MatrixForm=

$$\begin{pmatrix} \beta_2 \to \beta_4 + \beta_5 - \beta_6 \\ \beta_3 \to \beta_1 \\ \gamma_1 \to \beta_4 - \beta_6 + \gamma_8 \\ \gamma_2 \to \beta_4 + \beta_5 - \beta_6 \\ \gamma_3 \to \beta_1 - \gamma_8 \\ \gamma_4 \to \beta_4 \\ \gamma_5 \to \beta_5 - \gamma_8 \\ \gamma_6 \to -\beta_1 + \beta_4 + \gamma_8 \\ \gamma_7 \to \beta_1 - \beta_4 + \beta_6 - \gamma_8 \end{pmatrix}$$

Next, we verify that the simplified formula is indeed equivalent to what we originally got.

In[51]:=
```
flux = GetFlux[rxnG];
disgβ1 = Max[β1, β6] - β4 ≤ Min[β1, β5, β1 + β6 - β4] && β1 + β6 - β4 ≥ 0 && EQ && flux > 0;
disgβ2 = Abs[β4 - β6] ≤ β1 ≤ β4 + β5 && EQ && flux > 0;
Reduce[disgβ1 && Not[disgβ2]]
Reduce[Not[disgβ1] && disgβ2]
```

Out[54]=
```
False
```

Out[55]=
```
False
```

Disguised toric locus

In[56]:=
```
{β2κ, κs, vars} = Getβ2κ[rxnG];
disgκ1 = Reduce[Resolve[Exists[{x, y}, (disgβ2 /. β2κ) && κs > 0 && vars > 0]]];
```

$$\text{disg}\kappa 2 = \left(1 - \frac{\kappa_6}{\kappa_1}\right)\left(1 - \frac{\kappa_5}{\kappa_3}\right) \le \frac{\kappa_2\,\kappa_4}{\kappa_1\,\kappa_3} \le \left(1 + \frac{\kappa_6}{\kappa_1}\right)\left(1 + \frac{\kappa_5}{\kappa_3}\right) \;\&\&\; \kappa s > 0;$$

```
Reduce[disgκ1 && Not[disgκ2]]
Reduce[Not[disgκ1] && disgκ2]
```

Out[59]=
```
False
```

Out[60]=
```
False
```

How much of the simplex is covered by the disguised toric locus? We get approximately **58.3%**
(1 million simulations take about 25 seconds.)

In[61]:=
```
M = 1000000;
count = 0;
disgκ = (1 - κ6/κ1) (1 - κ5/κ3) ≤ (κ2 κ4)/(κ1 κ3) ≤ (1 + κ6/κ1) (1 + κ5/κ3);
For[i = 1, i ≤ M, i++, {
    rnd = RandomVariate[ExponentialDistribution[1], 8];
    If[disgκ /. Table[κj → rnd[[j]], {j, 1, 8}], count++];
   }];
Print[N[100 count / M, 3], "% of the simplex is covered by K^dt (G)"];
```

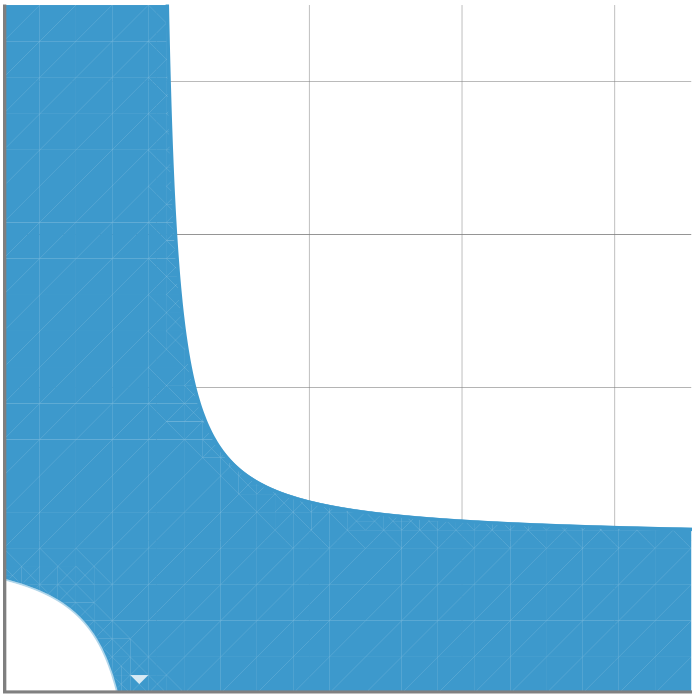
```
58.3% of the simplex is covered by K^dt (G)
```

Plots (Figure 1 in Section 1)

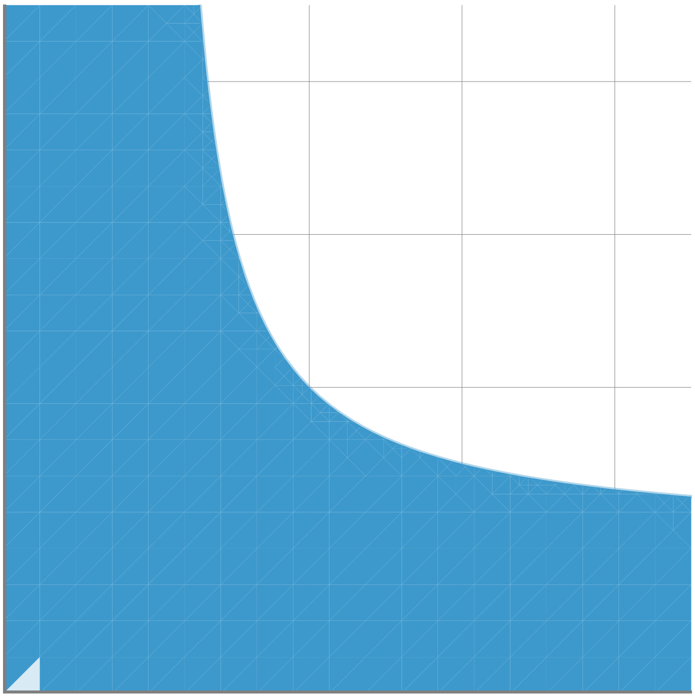
In[117]:=
```
dir = "C://bboros//Dropbox//disguised_toric//figures//";
disgκ = (1 - κ6/κ1) (1 - κ5/κ3) ≤ (κ2 κ4)/(κ1 κ3) ≤ (1 + κ6/κ1) (1 + κ5/κ3);
b = 4.5;
xaxis = ListLinePlot[{{0, 0}, {b, 0}}, PlotStyle → {Gray, Thickness[0.008]}];
yaxis = ListLinePlot[{{0, 0}, {0, b}}, PlotStyle → {Gray, Thickness[0.008]}];

(* case (κ2κ4)/(κ1κ3)<1 *)
κspec = {κ1 → 1, κ3 → 1, κ2 → 1, κ4 → 1 / 4};
rgnpl = RegionPlot[disgκ /. κspec, {κ5, 0, b},
    {κ6, 0, b}, PlotStyle → {Opacity[0.2]}, GridLines → Automatic,
    BoundaryStyle → None, Frame → False, PlotRange → {{0, b}, {0, b}}];
dt1 = Plot[1 - (κ2 κ4)/(κ1 κ3) / (1 - κ5/κ3) /. κspec, {κ5, 0, 1},
    PlotRange → {{0, 1}, {0, 1}}, PlotStyle → {Opacity[0.4], Thickness[0.005]}];
dt2 = Plot[1 - (κ2 κ4)/(κ1 κ3) / (1 - κ5/κ3) /. κspec, {κ5, 1, b},
    PlotRange → {{0, b}, {0, b}}, PlotStyle → {Opacity[0.4], Thickness[0.005]}];
shw = Show[rgnpl, dt1, dt2, xaxis, yaxis]
Export[dir <> "running_k4_small.png", shw];

(* case (κ2κ4)/(κ1κ3)=1 *)
κspec = {κ1 → 1, κ3 → 1, κ2 → 1, κ4 → 1};
rgnpl = RegionPlot[disgκ /. κspec, {κ5, 0, b},
    {κ6, 0, b}, PlotStyle → {Opacity[0.2]}, GridLines → Automatic,
    BoundaryStyle → None, Frame → False, PlotRange → {{0, b}, {0, b}}];
dt = Plot[1 - (κ2 κ4)/(κ1 κ3) / (1 - κ5/κ3) /. κspec, {κ5, 1, b},
    PlotRange → {{0, b}, {0, b}}, PlotStyle → {Opacity[0.4], Thickness[0.005]}];
```

```
shw = Show[rgnpl, dt, xaxis, yaxis]
Export[dir <> "running_k4_equal.png", shw];



(* case κ2κ4/(κ1κ3) >1 *)
κspec = {κ1 → 1, κ3 → 1, κ2 → 1, κ4 → 3};
rgnpl = RegionPlot[disgκ /. κspec, {κ5, 0, b},
    {κ6, 0, b}, PlotStyle → {Opacity[0.2]}, GridLines → Automatic,
    BoundaryStyle → None, Frame → False, PlotRange → {{0, b}, {0, b}}];
dt1 = Plot[1 - (κ2 κ4)/(κ1 κ3) / (1 - κ5/κ3) /. κspec, {κ5, 1, b},
    PlotRange → {{0, b}, {0, b}}, PlotStyle → {Opacity[0.4], Thickness[0.005]}];
dt2 = Plot[(κ2 κ4)/(κ1 κ3) / (1 + κ5/κ3) - 1 /. κspec, {κ5, 0, b},
    PlotRange → {{0, b}, {0, b}}, PlotStyle → {Opacity[0.4], Thickness[0.005]}];
toric = Plot[(κ2 κ4)/(κ1 κ3) / (1 + κ5/κ3) - 1 /. κspec, {κ5, 0, b},
    PlotRange → {{0, b}, {0, b}}, PlotStyle → {Red, Dashed, Thickness[0.006]}];
shw = Show[rgnpl, dt1, dt2, toric, xaxis, yaxis]
Export[dir <> "running_k4_big.png", shw];

(* flux cone *)
b = 6.5;
xaxis = ListLinePlot[{{0, 0}, {b, 0}}, PlotStyle → {Gray, Thickness[0.008]}];
yaxis = ListLinePlot[{{0, 0}, {0, b}}, PlotStyle → {Gray, Thickness[0.008]}];
rgnpl = RegionPlot[Abs[β4 - β1] ≤ β1 ≤ β4 + (β1 + β4) / 2, {β1, 0, b},
    {β4, 0, b}, PlotStyle → {Opacity[0.2]}, GridLines → Automatic,
    BoundaryStyle → None, Frame → False, PlotRange → {{0, b}, {0, b}}];
dt1 = Plot[2 β1, {β1, 0, b}, PlotRange → {{0, b}, {0, b}},
    PlotStyle → {Opacity[0.4], Thickness[0.005]}];
dt2 = Plot[1/3 β1, {β1, 0, b}, PlotRange → {{0, b}, {0, b}},
    PlotStyle → {Opacity[0.4], Thickness[0.005]}];
toric = Plot[2 β1, {β1, 0, b}, PlotRange → {{0, b}, {0, b}},
    PlotStyle → {Red, Dashed, Thickness[0.006]}];
shw = Show[rgnpl, xaxis, yaxis, dt1, dt2, toric]
Export[dir <> "running_flux.png", shw];
```

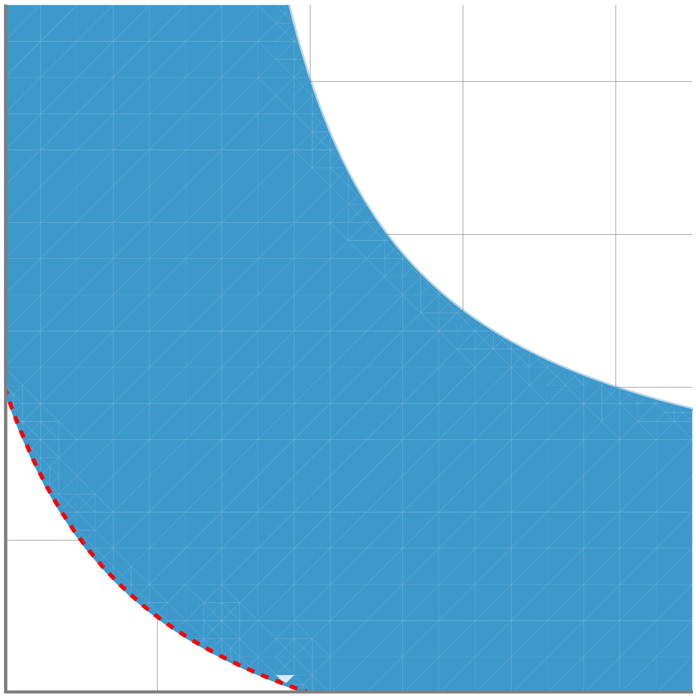
Out[126]=



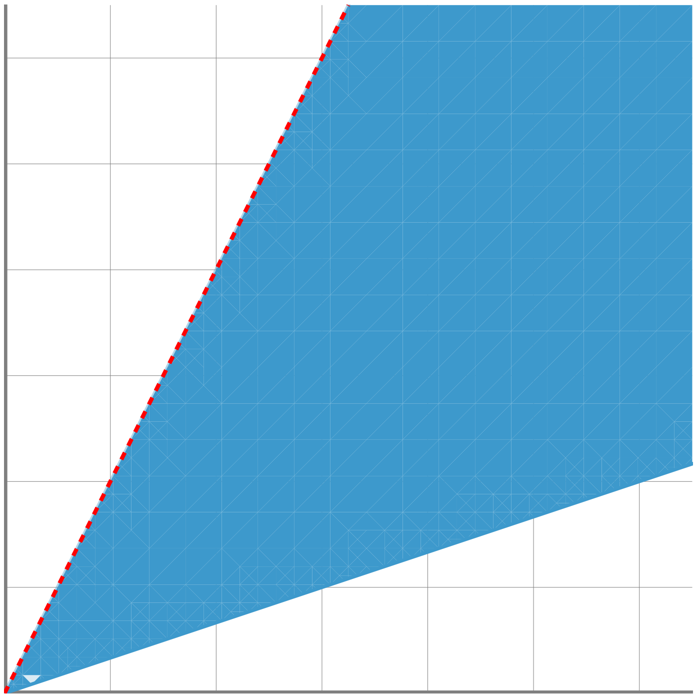
Out[131]=



Out[138]=



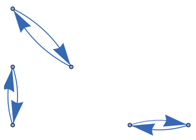
Out[147]=



7.3 Reversible LVA

In[97]:=
```
rxnG = {{2, 0} → {3, 0}, {3, 0} → {2, 0},
    {1, 1} → {0, 2}, {0, 2} → {1, 1}, {0, 1} → {0, 0}, {0, 0} → {0, 1}};
DrawEGraph[rxnG]〚1〛
```

Out[98]=



Five positive equilibria

In[99]:=
```
f = GetMassAction[rxnG]〚1〛;
κsubst = {κ1 → 2, κ2 → 1/10, κ3 → 3, κ4 → 1, κ5 → 3, κ6 → 1};
eqs = N[Solve[(f /. κsubst) == 0 && {x, y} > 0, {x, y}]]
J = D[f, {{x, y}}];
Eigenvalues /@ (J /. κsubst /. eqs)
```

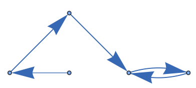
Out[101]=
```
{{x → 0.175311, y → 0.353643}, {x → 0.601603, y → 0.56736},
 {x → 0.781398, y → 0.724485}, {x → 4.28929, y → 9.96818}, {x → 14.1262, y → 39.4038}}
```

Out[103]=
```
{{-3.24818, -0.302082}, {-1.86674, 0.132589},
 {-1.01267, -0.323142}, {-29.2734, 0.938091}, {-157.916, -3.08462}}
```

7.4 A network with a subcritical Bogdanov-Takens bifurcation

In[104]:=
```
rxnG =
  {{1, 0} → {0, 0}, {0, 0} → {1, 1}, {1, 1} → {2, 0}, {2, 0} → {3, 0}, {3, 0} → {2, 0}};
DrawEGraph[rxnG]〚1〛
```

Out[105]=



Three positive equilibria

In[106]:=

```
f = GetMassAction[rxnG][[1]];
κsubst = {κ1 → 7, κ2 → 1, κ3 → 2, κ4 → 7, κ5 → 2};
eqs = N[Solve[(f /. κsubst) == 0 && {x, y} > 0, {x, y}]]
J = D[f, {{x, y}}];
Eigenvalues /@ (J /. κsubst /. eqs)
```

Out[108]=

```
{{x → 0.5, y → 1.}, {x → 1., y → 0.5}, {x → 2., y → 0.25}}
```

Out[110]=

```
{{-0.25 + 1.19896 i, -0.25 - 1.19896 i},
 {-1.41421, 1.41421}, {-3.25 + 1.19896 i, -3.25 - 1.19896 i}}
```

Disguised toric locus

In[111]:=

$$\text{Resolve}\left[\text{Exists}\left[\{x, y\}, f == 0 \,\&\&\, x \le \frac{\kappa_2}{\kappa_1} \,\&\&\, \{x, y\} > 0 \,\&\&\, \{\kappa_1, \kappa_2, \kappa_3, \kappa_4, \kappa_5\} > 0\right]\right]$$

Out[111]=

$$\kappa_1 > 0 \,\&\&\, \kappa_2 > 0 \,\&\&\, \kappa_4 > 0 \,\&\&\, \kappa_5 \ge \frac{\kappa_1^3 + \kappa_1 \kappa_2 \kappa_4}{\kappa_2^2} \,\&\&\, \kappa_3 > 0$$

How much of the simplex is covered by the disguised toric locus? We get approximately **35.4%** (5 million simulations take about 1 minute.)

In[112]:=

```
M = 5 000 000;
count = 0;
disgκ = κ5/κ1 ≥ (κ1/κ2)^2 + κ4/κ2;
For[i = 1, i ≤ M, i++, {
    rnd = RandomVariate[ExponentialDistribution[1], 5];
    If[disgκ /. Table[κj → rnd[[j]], {j, 1, 5}], count++];
   }];
Print[N[100 count / M, 3], "% of the simplex is covered by K^dt (G)"];
```

```
35.4% of the simplex is covered by K^dt (G)
```

7.6 Tetrahedron

In[149]:=

```
rxnG = {{0, 0, 0} → {1, 0, 0},
   {1, 0, 0} → {0, 0, 0}, {1, 0, 0} → {2, 0, 0}, {2, 0, 0} → {1, 0, 0},
   {1, 0, 0} → {0, 1, 1}, {0, 1, 1} → {1, 0, 0}, {0, 2, 0} → {0, 1, 1},
   {0, 1, 1} → {0, 2, 0}, {0, 1, 1} → {0, 0, 2}, {0, 0, 2} → {0, 1, 1}};
```

Disguised toric flux cone

In[150]:=

```
{DE, VB} = GetDEandVB[rxnG, rxnG];
sol = Solve[DE && VB, {β2, β5, β8, γ1, γ2, γ3, γ4, γ5, γ6, γ7, γ8, γ9, γ10}][[1]];
MatrixForm[sol]
```

Out[152]//MatrixForm=

$$\begin{pmatrix} \beta_2 \to \beta_1 + \beta_3 - \beta_4 \\ \beta_5 \to \beta_6 \\ \beta_8 \to \beta_7 + \beta_9 - \beta_{10} \\ \gamma_1 \to \beta_1 \\ \gamma_2 \to \beta_1 \\ \gamma_3 \to \beta_4 \\ \gamma_4 \to \beta_4 \\ \gamma_5 \to \beta_6 \\ \gamma_6 \to \beta_6 \\ \gamma_7 \to \beta_7 \\ \gamma_8 \to \beta_7 \\ \gamma_9 \to \beta_{10} \\ \gamma_{10} \to \beta_{10} \end{pmatrix}$$

7.7 A four-dimensional example

In[153]:=

```
rxnG = {{0, 0, 0, 0} → {1, 0, 0, 0}, {1, 0, 0, 0} → {0, 0, 0, 0},
   {0, 0, 0, 0} → {0, 1, 0, 0}, {0, 1, 0, 0} → {0, 0, 0, 0}, {0, 0, 0, 0} → {0, 0, 1, 0},
   {0, 0, 1, 0} → {0, 0, 0, 0}, {0, 0, 0, 0} → {0, 0, 0, 1}, {0, 0, 0, 1} → {0, 0, 0, 0},
   {0, 0, 0, 1} → {2, 0, 0, 0}, {2, 0, 0, 0} → {0, 0, 0, 1}, {0, 0, 0, 1} → {0, 2, 0, 0},
   {0, 2, 0, 0} → {0, 0, 0, 1}, {0, 0, 0, 1} → {0, 0, 2, 0}, {0, 0, 2, 0} → {0, 0, 0, 1}};
```

Equilibrium fluxes

In[154]:=

```
EQ = GetEQ[rxnG]
```

Out[154]=

$$\{\beta_1 - \beta_2 + 2\,\beta_9 - 2\,\beta_{10},\ \beta_3 - \beta_4 + 2\,\beta_{11} - 2\,\beta_{12},\ \beta_5 - \beta_6 + 2\,\beta_{13} - 2\,\beta_{14},\ \beta_7 - \beta_8 - \beta_9 + \beta_{10} - \beta_{11} + \beta_{12} - \beta_{13} + \beta_{14}\} == 0$$

Disguised toric flux cone

In[155]:=

```
rxnH = Join[rxnG,
   {{1, 0, 0, 0} → {2, 0, 0, 0}, {0, 1, 0, 0} → {0, 2, 0, 0}, {0, 0, 1, 0} → {0, 0, 2, 0},
    {0, 0, 0, 1} → {1, 0, 0, 0}, {0, 0, 0, 1} → {0, 1, 0, 0}, {0, 0, 0, 1} → {0, 0, 1, 0}}];
{DE, VB} = GetDEandVB[rxnG, rxnH];
sol = Solve[DE && VB, {β8, β9, β11, β13, γ1, γ3, γ5,
     γ7, γ8, γ9, γ10, γ11, γ12, γ13, γ14, γ15, γ16, γ17, γ18, γ19, γ20}][[1]];
MatrixForm[sol]
```

Out[158]//MatrixForm=

$$\begin{pmatrix}
\beta_8 \to \frac{1}{2}(\beta_1 - \beta_2 + \beta_3 - \beta_4 + \beta_5 - \beta_6 + 2\beta_7) \\
\beta_9 \to \frac{1}{2}(-\beta_1 + \beta_2 + 2\beta_{10}) \\
\beta_{11} \to \frac{1}{2}(-\beta_3 + \beta_4 + 2\beta_{12}) \\
\beta_{13} \to \frac{1}{2}(-\beta_5 + \beta_6 + 2\beta_{14}) \\
\gamma_1 \to \beta_1 \\
\gamma_3 \to \beta_3 \\
\gamma_5 \to \beta_5 \\
\gamma_7 \to \beta_7 \\
\gamma_8 \to \beta_1 + \beta_3 + \beta_5 + \beta_7 - \gamma_2 - \gamma_4 - \gamma_6 \\
\gamma_9 \to \beta_2 + \beta_{10} - \gamma_2 \\
\gamma_{10} \to \beta_{10} \\
\gamma_{11} \to \beta_4 + \beta_{12} - \gamma_4 \\
\gamma_{12} \to \beta_{12} \\
\gamma_{13} \to \beta_6 + \beta_{14} - \gamma_6 \\
\gamma_{14} \to \beta_{14} \\
\gamma_{15} \to -\beta_2 + \gamma_2 \\
\gamma_{16} \to -\beta_4 + \gamma_4 \\
\gamma_{17} \to -\beta_6 + \gamma_6 \\
\gamma_{18} \to -\beta_1 - \beta_2 + 2\gamma_2 \\
\gamma_{19} \to -\beta_3 - \beta_4 + 2\gamma_4 \\
\gamma_{20} \to -\beta_5 - \beta_6 + 2\gamma_6
\end{pmatrix}$$

Hence, $\gamma_2 \geq \max\left(\beta_2, \frac{1}{2}(\beta_1 + \beta_2)\right)$, $\gamma_4 \geq \max\left(\beta_4, \frac{1}{2}(\beta_3 + \beta_4)\right)$, $\gamma_6 \geq \max\left(\beta_5, \frac{1}{2}(\beta_5 + \beta_6)\right)$ in order to guarantee $\gamma_{15}, \gamma_{16}, \gamma_{17}, \gamma_{18}, \gamma_{19}, \gamma_{20} \geq 0$.
With $\gamma_2 = \max\left(\beta_2, \frac{1}{2}(\beta_1 + \beta_2)\right)$, $\gamma_4 = \max\left(\beta_4, \frac{1}{2}(\beta_3 + \beta_4)\right)$, $\gamma_6 = \max\left(\beta_5, \frac{1}{2}(\beta_5 + \beta_6)\right)$, we have $\gamma_9 = \min(\beta_9, \beta_{10})$, $\gamma_{11} = \min(\beta_{11}, \beta_{12})$, $\gamma_{13} = \min(\beta_{13}, \beta_{14})$, all positive.
Hence, the only thing we still need is that $\gamma_8 \geq 0$, which leads to the condition
$\beta_1 - \max\left(\beta_2, \frac{1}{2}(\beta_1 + \beta_2)\right) + \beta_3 - \max\left(\beta_4, \frac{1}{2}(\beta_3 + \beta_4)\right) + \beta_5 - \max\left(\beta_6, \frac{1}{2}(\beta_5 + \beta_6)\right) + \beta_7 \geq 0.$

How much of the simplex is covered by the disguised toric locus? We get approximately **62.6%**
We have an explicit formula for the disguised toric flux cone, but not for the disguised toric locus.
Hence, for a given rate constant, we solve numerically for the (unique) positive equilibrium, and then compute the corresponding flux.
(100k simulations take about 25 minutes, because solving for the positive equilibrium is slow.)
We got 62.70%, 62.30%, 62.77%, 62.49%, 62.43%, 62.98%, 62.40%, 62.38%, 62.58%, 62.77% for

different runs of 100k simulations, a total of 1 million runs, the average is 62.6%.

In[159]:=

```
{f, monomials} = GetMassAction[rxnG][[{1, 4}]];
M = 10 000;
count = 0;
disgβ = β1 - Max[β2, (β1 + β2)/2] + β3 - Max[β4, (β3 + β4)/2] + β5 - Max[β6, (β5 + β6)/2] + β7 ≥ 0;
For[i = 1, i ≤ M, i++, {
    rnd = RandomVariate[ExponentialDistribution[1], 14];
    κsubst = Table[κj → rnd[[j]], {j, 1, 14}];
    ff = f /. κsubst;
    equil = NSolve[ff == 0 && x > 0 && y > 0 && z > 0 && w > 0, {x, y, z, w}][[1]];
    monomspec = monomials[[1 ;; 7]] /. κsubst /. equil;
    βsubst = Table[βj → monomspec[[j]], {j, 1, 7}];
    If[disgβ /. βsubst, count++];
   }];
Print[N[100 count / M, 3],
  "% of the simplex is covered by K^dt(G) (based on ", M, " simulations)"];
```

```
62.3% of the simplex is covered by K^dt(G) (based on 10 000 simulations)
```

Appendix

We briefly explain how to sample uniformly from the simplex $\left\{\kappa \in \mathbb{R}^m_{\geq 0} \mid \sum_{i=1}^{m} \kappa_i = 1\right\}$.

Lemma Let $X_1, \ldots, X_m$ ~Exponential(λ) be independent random variables. Let $Y_i = \dfrac{X_i}{X_1 + \cdots + X_m}$ ($i = 1, \ldots, m$). Then $(Y_1, \ldots, Y_m)$ is uniformly distributed on the simplex.

Proof Let $S = X_1 + \cdots + X_m$ and $g(x_1, \ldots, x_m) = \left(\dfrac{x_1}{s}, \ldots, \dfrac{x_{m-1}}{s}, s\right)$. Then the Jacobian determinant of g^{-1} equals s^{m-1}. Hence, the joint density function of $(Y_1, \ldots, Y_{m-1}, S)$ is $\lambda^m e^{-\lambda s} s^{m-1}$ for $y_1, \ldots, y_{m-1} > 0$, $y_1 + \cdots + y_{m-1} < 1$, $s > 0$. Note that this function is independent of $y_1, \ldots, y_{m-1}$. By integration w.r.t. s, find that the joint density function of $(Y_1, \ldots, Y_{m-1})$ is constant $\Gamma(m) = (m-1)!$ over the simplex. This concludes the proof.

Remark Since the equilibrium of a mass-action doesn't depend on scaling of the rate constants, in practice we don't need to divide by the sum. Also, notice that the parameter λ of the exponential distribution has no effect on the joint distribution of $(Y_1, \ldots, Y_m)$ (which is also called Dirichlet(1,...,1) distribution).