Synthetic Test Case Overview

In[1]:=

```
versca1 = 0.01;
versca2 = 0.05;
edgsca1 = 0.007;
edgsca2 = 0.02;
Graph[{Style[1, Darker@Blue], Style[2, Darker@Red], Style[3, Darker@Green]},
  {Style[1 <-> 2, Thickness[40/17 edgsca1], Gray], Style[2 <-> 3, Thickness[4/17 edgsca2], Gray],
   Style[3 <-> 1, Thickness[40/17 edgsca1], Dashed, Gray]}, VertexShapeFunction → "Circle",
  VertexSize → {1 -> 417/17 versca1, 2 -> 21/17 versca2, 3 -> 21/17 versca2},
  VertexLabels → {1 → "Damp: 0.1", 2 → "Damp: 1", 3 → "Damp: 10"}]
```

style | plus fo··· | bleu | style | plus fo··· | rouge | style | plus fo··· | vert

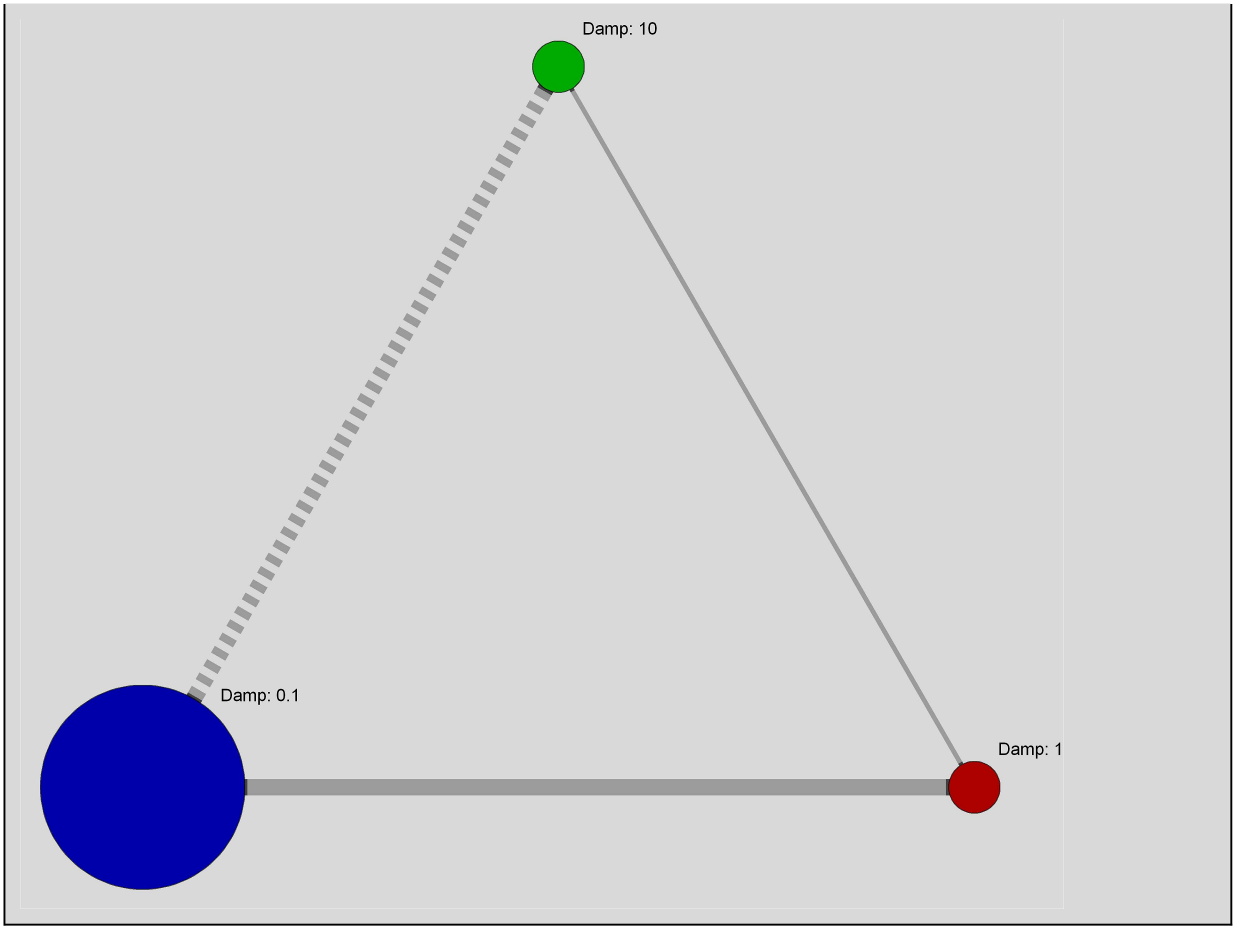
style | épaisseur | gris | style | épaisseur | gris

style | épaisseur | rayé | gris | fonction de forme de noeud | cercle

taille de noeud

étiquettes des sommets

Out[1]=



In[2]:=

$$H = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ \frac{417}{17} & -\frac{40}{17} & \frac{40}{17} & \frac{1}{10} & 0 & 0 \\ -\frac{40}{17} & \frac{21}{17} & -\frac{4}{17} & 0 & 1 & 0 \\ \frac{40}{17} & -\frac{4}{17} & \frac{21}{17} & 0 & 0 & 10 \end{pmatrix};$$

In[3]:=

$$\text{eqnham} = \underbrace{\text{Flatten}}_{\text{aplatis}}\left[\underbrace{\text{FullSimplify}}_{\text{simplifie complètement}}\left[\begin{pmatrix} x_1'[t] \\ x_2'[t] \\ x_3'[t] \\ p_1'[t] \\ p_2'[t] \\ p_3'[t] \end{pmatrix} + H.\begin{pmatrix} x_1[t] \\ x_2[t] \\ x_3[t] \\ p_1[t] \\ p_2[t] \\ p_3[t] \end{pmatrix}\right]\right];$$

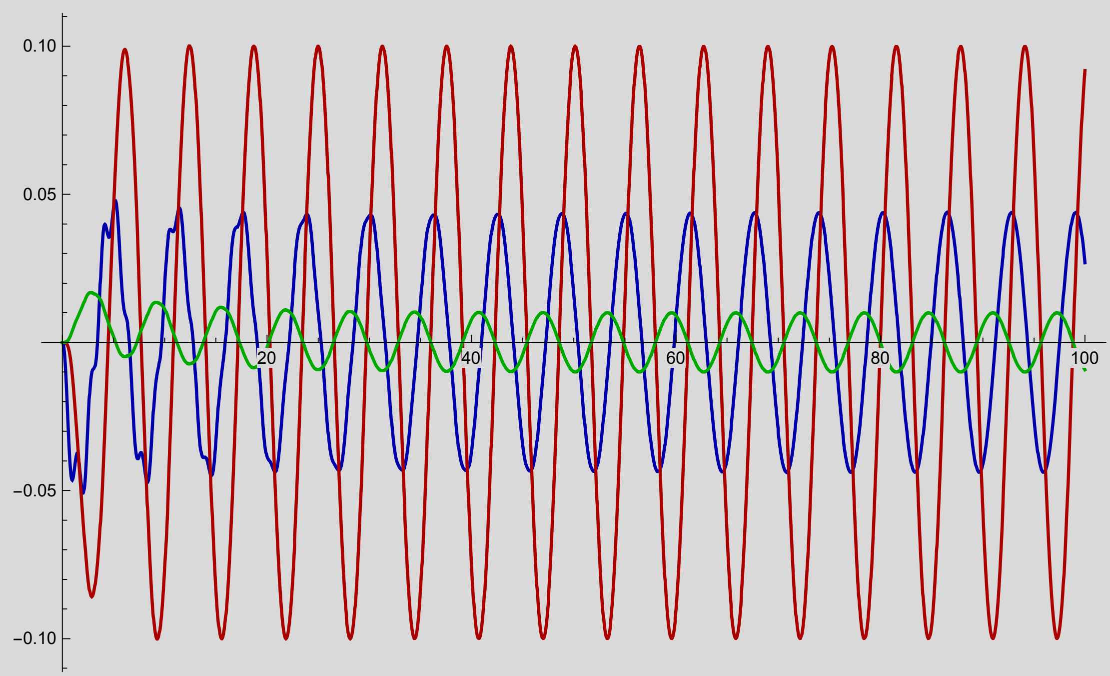
In[5]:=

```
solham = NDSolve[
           résolveur numérique d'équations différentielles
  {eqnham⟦1⟧ == 0, eqnham⟦2⟧ == 0, eqnham⟦3⟧ == 0, eqnham⟦4⟧ == Cos[t + 1.7], eqnham⟦5⟧ == 0,
                                                                    cosinus
   eqnham⟦6⟧ == 0, x1[0] == 0, x2[0] == 0, x3[0] == 0, p1[0] == 0, p2[0] == 0, p3[0] == 0},
  {x1, x2, x3, p1, p2, p3}, {t, 0, 100}, Method → "ImplicitRungeKutta"];
                                         méthode
```

In[6]:=

```
Plot[{x1[t] /. solham, x2[t] /. solham, x3[t] /. solham}, {t, 0, 100}, PlotRange → All,
tracé de courbes                                                        zone de tracé  tout
 PlotStyle → {Darker@Blue, Darker@Red, Darker@Green}, ImageSize → 570]
 style de tracé plus fo··· bleu plus fo··· ro··· plus fo··· vert  taille d'image
```

Out[6]=



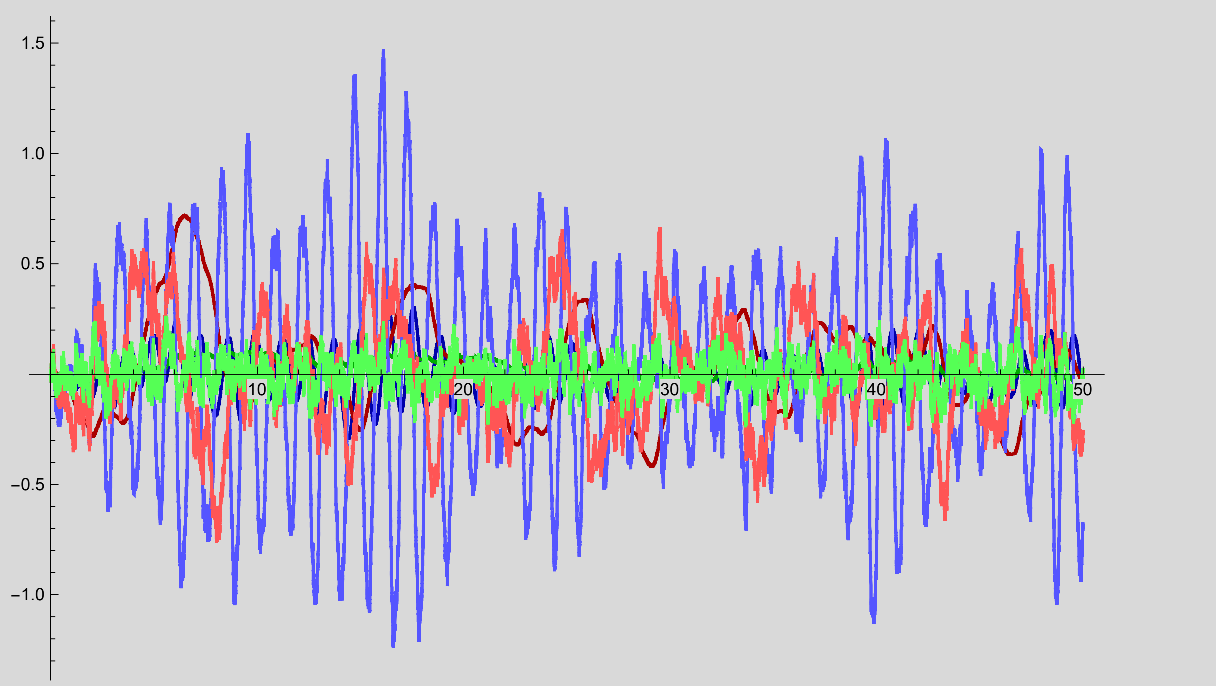
Stochastic ODE & Data Generation

In[32]:=

```
proc = ItoProcess[{{a[t], b[t], c[t], -a[t]/10 - 417 x[t]/17 + 40 y[t]/17 - 40 z[t]/17 + Cos[t + 1.7],
    -b[t] + 40 x[t]/17 - 21 y[t]/17 + 4 z[t]/17, -10 c[t] - 40 x[t]/17 + 4 y[t]/17 - 21 z[t]/17},
   {{0}, {0}, {0}, {1/3}, {1/3}, {1/3}}, {x[t], y[t], z[t], a[t], b[t], c[t]}},
  {{x, y, z, a, b, c}, {0, 0, 0, 0, 0, 0}}, {t, 0}];
time = 50;
resolution = 10^-3;
path = RandomFunction[proc, {0., time, resolution}, Method → "StochasticRungeKutta"];
ListLinePlot[path, PlotStyle → {Darker@Blue, Darker@Red, Darker@Green,
   Lighter@Blue, Lighter@Red, Lighter@Green}, PlotRange → All, ImageSize → 570]
```

processus Ito · cosinus · fonction aléatoire · méthode · tracé de liste de ligne · style de tracé · plus fo··· · bleu · plus fo··· · ro··· · plus fo··· · vert · plus clair · bleu · plus clair · ro··· · plus clair · vert · zone de tracé · tout · taille d'image

Out[32]=



Discrete Frequencies

In[8]:=

```
N@ time/(2 π) {1, 5}
```

valeur numérique

Out[8]=

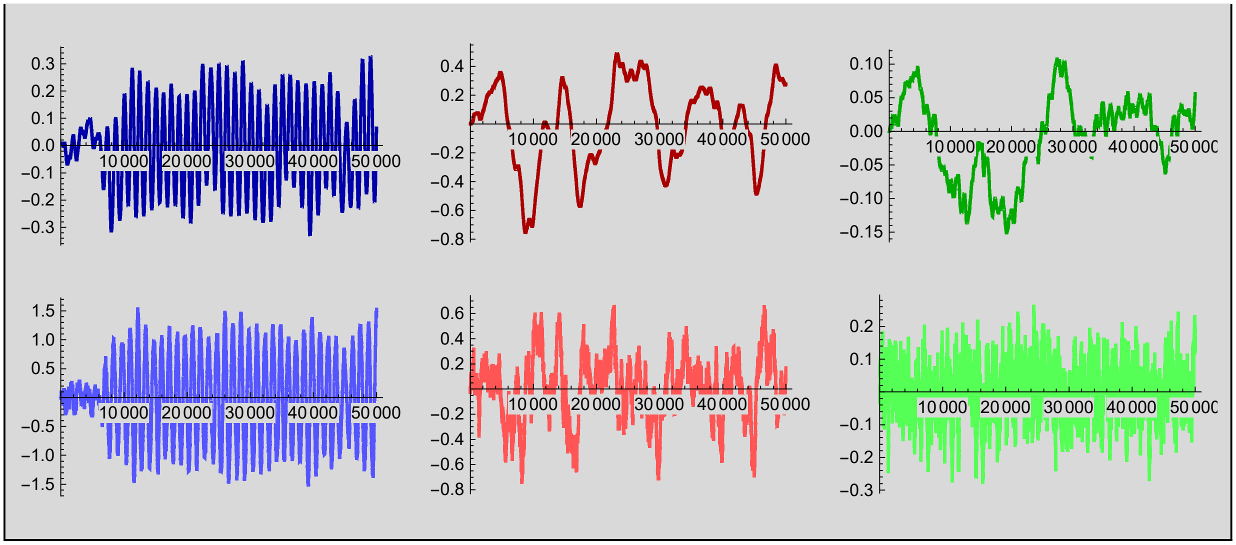
```
{7.95775, 39.7887}
```

In[40]:=

```
tsfulldata = Flatten[Normal@
                   aplatis    forme normale
   RandomFunction[proc, {0., time, resolution}, Method → "StochasticRungeKutta"], 1];
   fonction aléatoire                              méthode
samples = Length[tsfulldata] - 1;
          longueur
tssignal = Table[tsfulldata[[k, 2, i]], {i, 1, 6}, {k, 1, samples}];
           table
tsderivative =
  Table[(tsfulldata[[k + 1, 2, i]] - tsfulldata[[k, 2, i]]) / (tsfulldata[[k + 1, 1]] - tsfulldata[[k, 1]]), {i, 1, 6}, {k, 1, samples}];
  table

matrixsignal = Sum[{tssignal[[ ;; , k]]}^T.{tssignal[[ ;; , k]]}, {k, 1, samples}] / samples;
               somme
matrixderivativesignal =
  Sum[{tssignal[[ ;; , k]]}^T.{tsderivative[[4 ;; 6, k]]}, {k, 1, samples}] / samples;
  somme
colorlist = {Darker@Blue, Darker@Red, Darker@Green,
             plus fo··· bleu  plus fo··· ro··· plus fo··· vert
   Lighter@Blue, Lighter@Red, Lighter@Green};
   plus clair bleu plus clair ro··· plus clair vert
fouriersignal = Table[Fourier[tssignal[[i]]], {i, 1, 6}];
                table  transformée de Fourier discrète
fourierderivative = Table[Fourier[tsderivative[[i]]], {i, 1, 6}];
                    table  transformée de Fourier discrète
fouriermatrixsignal = Table[
                      table
   Re[Conjugate[{fouriersignal[[ ;; , k]]}^T].{fouriersignal[[ ;; , k]]}], {k, 1, samples}];
      conjugué
fouriervectorderivativesignal = Table[Re[Conjugate[fourierderivative[[i, k]]]
                                table p·· conjugué
     fouriersignal[[ ;; , k]]], {i, 1, 6}, {k, 1, samples}];
fourierdynamic = Table[Abs[fourierderivative[[i + 3, k]] +
                 table valeur absolue
     H[[4 ;; 6,  ;;]][[i,  ;;]].fouriersignal[[ ;; , k]]], {i, 1, 3}, {k, 1, samples}];
GraphicsGrid[Partition[Table[ListPlot[tssignal[[i]], PlotRange → {{0, All}, All},
             partitionne table  tracé de liste        zone de tracé   tout   tout
     Joined → True, PlotStyle → colorlist[[i]], ImageSize → 190], {i, 1, 6}], 3]]
     vrai     style de tracé                  taille d'image
```

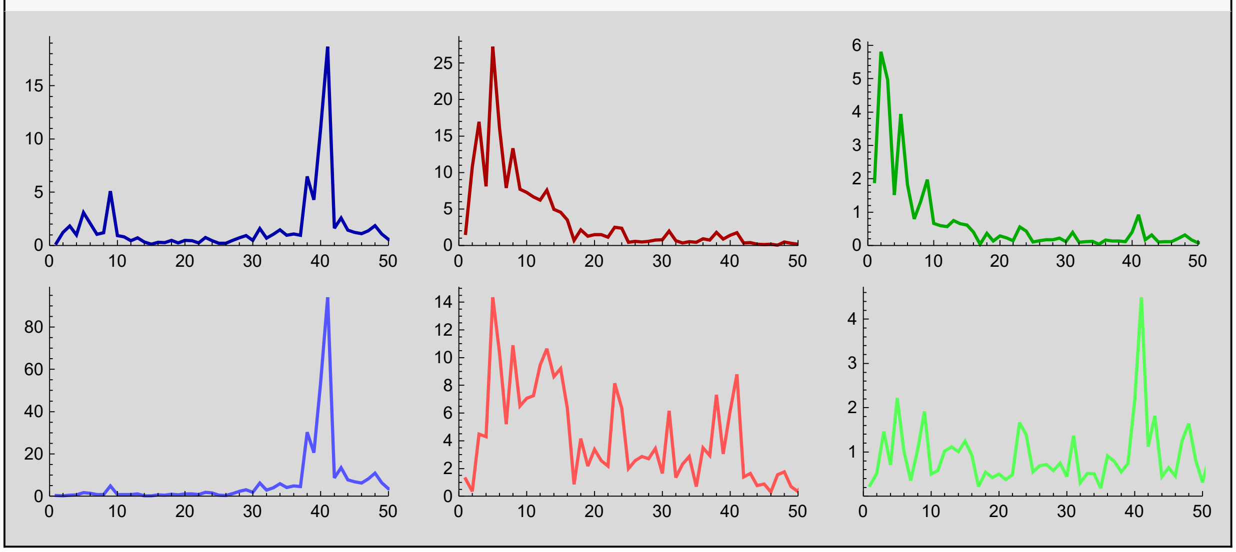
Out[40]=



In[41]:=

```
GraphicsGrid[
grille de graphiques
  Partition[Table[ListPlot[{Abs[fouriersignal⟦i⟧]}, PlotRange → {{0, 50}, {0, All}},
  partitionne  table  tracé de liste  valeur absolue     zone de tracé        tout
     Joined → True, PlotStyle → colorlist⟦i⟧, ImageSize → 190], {i, 1, 6}], 3]]
             vrai     style de tracé                  taille d'image
```

Out[41]=

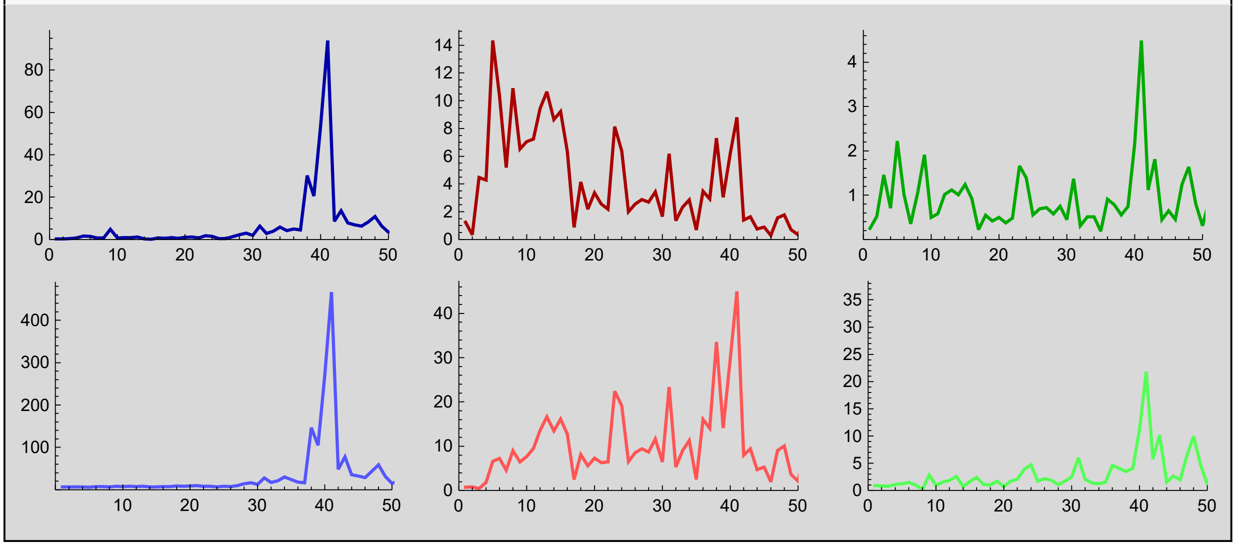


In[42]:=

```
GraphicsGrid[Partition[
  Table[ListPlot[{Abs[fourierderivative〚i〛]}, PlotRange → {{0, 50}, {0, All}},
    Joined → True, PlotStyle → colorlist〚i〛, ImageSize → 190], {i, 1, 6}], 3]]
```

grille de graphi··· | partitionne | tracé de liste | valeur absolue | zone de tracé | tout | vrai | style de tracé | taille d'image

Out[42]=

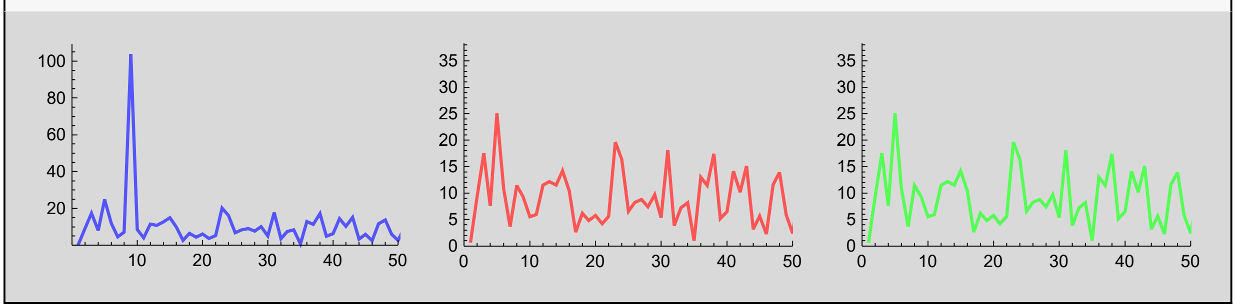


In[43]:=

```
GraphicsGrid[
 {Table[ListPlot[{Abs[fourierdynamic〚i〛]}, PlotRange → {{0, 50}, {0, All}},
    Joined → True, PlotStyle → colorlist〚i + 3〛, ImageSize → 190], {i, 1, 3}]}]
```

grille de graphiques | table | tracé de liste | valeur absolue | zone de tracé | tout | joint | vrai | style de tracé | taille d'image

Out[43]=



Log-Likelihood Regression

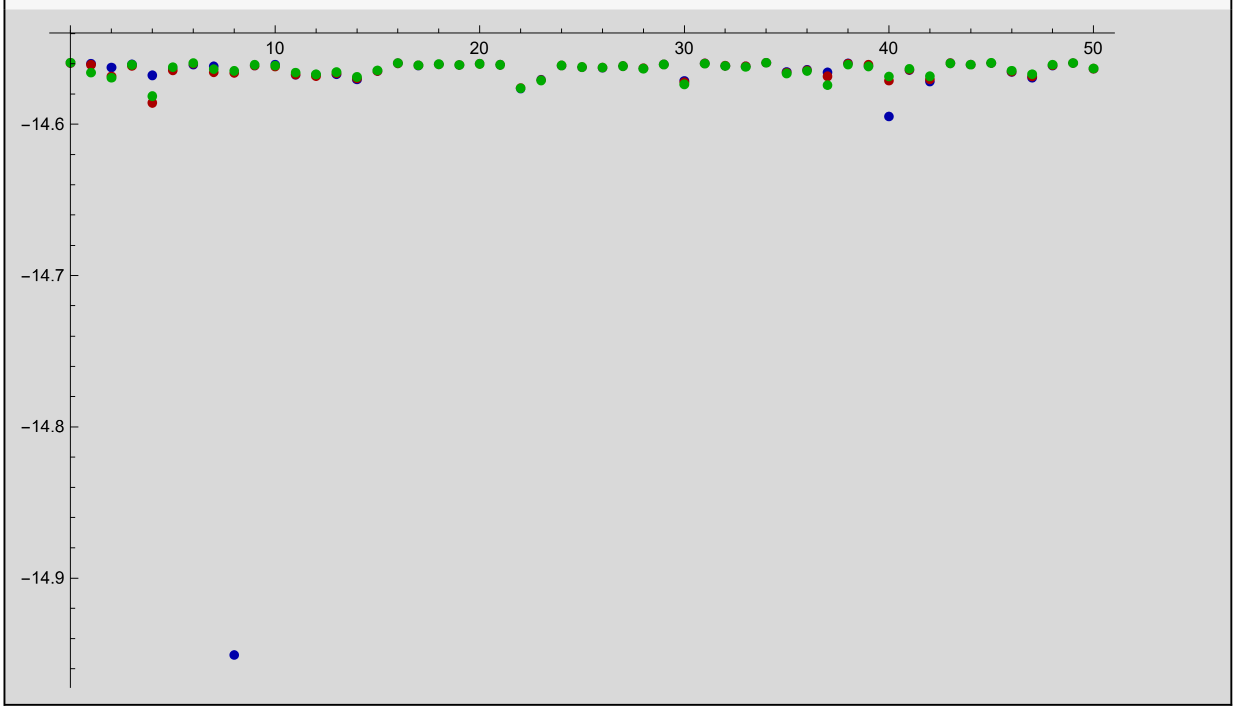
In[44]:=

```
localizationreg = Table[varmat = (a₁ b₁ b₂ c₁ 0 0
                                  b₁ a₂ b₃ 0 c₂ 0
                                  b₂ b₃ a₃ 0 0 c₃);
  NMinimize[{Tr[varmatᵀ.varmat.matrixsignal] + 2 Tr[varmat.matrixderivativesignal] +
      1/2 γ² - (2 γ)/Sqrt[samples] (Tr[{varmat[[l, ;;]]}ᵀ.{varmat[[l, ;;]]}.fouriermatrixsignal[[
            k + 1]]] + 2 fouriervectorderivativesignal[[3 + l, k + 1]].varmat[[l, ;;]] +
          Abs[fourierderivative[[3 + l, k + 1]]]²)^(1/2), γ ≥ 0 && c₁ ≥ 0 && c₂ ≥ 0 && c₃ ≥ 0 && a₁ ≥
      0 && a₂ ≥ 0 && a₃ ≥ 0}, {a₁, a₂, a₃, b₁, b₂, b₃, c₁, c₂, c₃, γ}], {l, 1, 3}, {k, 0, 50}];
freqreg = Table[{k, localizationreg[[l, k + 1, 1]]}, {l, 1, 3}, {k, 0, 50}];
```

In[45]:=

```
ListPlot[freqreg, PlotRange → All, PlotStyle → colorlist[[1 ;; 3]], ImageSize → 570]
```

Out[45]=



Estimated Matrix & Amplitude

In[46]:=

```
siteest = 1;
frequest = 8;
Grid[{{MatrixForm[varmat /. localizationreg⟦siteest, frequest + 1, 2⟧],
 grille      apparence matricielle
    "γ = " ~~ ToString[γ /. localizationreg⟦siteest, frequest + 1, 2⟧]}}]
                convertis en chaîne de caractères
```

Out[46]=

$$\begin{pmatrix} 24.4667 & -2.58193 & 2.43902 & 0.0289876 & 0 & 0 \\ -2.58193 & 1.05439 & -0.526089 & 0 & 1.147 & 0 \\ 2.43902 & -0.526089 & 1.98908 & 0 & 0 & 9.98179 \end{pmatrix} \quad \gamma = 0.910979$$

Ground Truth

In[47]:=

```
Grid[{{MatrixForm[N@H⟦4 ;; 6, ;;⟧], "γ = 1"}}]
 grille      apparence m·· valeur numérique
```

Out[47]=

$$\begin{pmatrix} 24.5294 & -2.35294 & 2.35294 & 0.1 & 0. & 0. \\ -2.35294 & 1.23529 & -0.235294 & 0. & 1. & 0. \\ 2.35294 & -0.235294 & 1.23529 & 0. & 0. & 10. \end{pmatrix} \quad \gamma = 1$$

Log-Likelihood Spatial Relaxation

In[48]:=

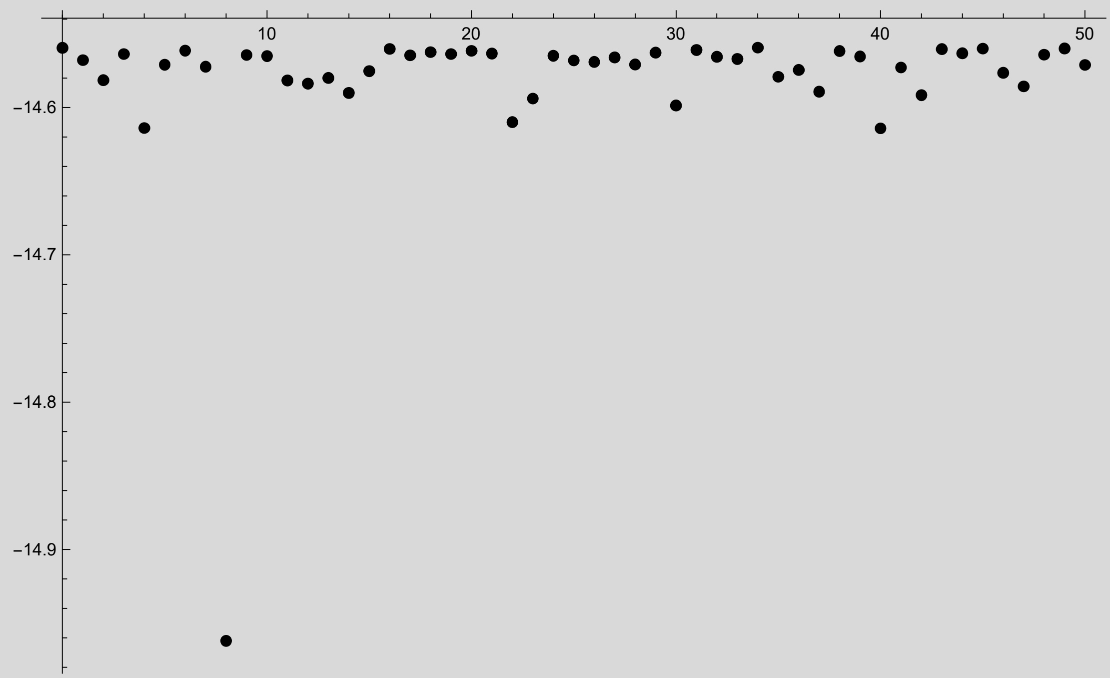
```
relocalizationreg = Table[varmat = ( a1 b1 b2 c1 0 0
                                     b1 a2 b3 0 c2 0
                                     b2 b3 a3 0 0 c3 );
   NMinimize[{Tr[varmatᵀ.varmat.matrixsignal] + 2 Tr[varmat.matrixderivativesignal] +
       1/2 (γ1^2 + γ2^2 + γ3^2) - (2 γ1/Sqrt[samples] (Tr[{varmat[[1, ;;]]}ᵀ.{varmat[[1, ;;]]}.
              fouriermatrixsignal[[k + 1]]] + 2 fouriervectorderivativesignal[[3 +
                1, k + 1]].varmat[[1, ;;]] + Abs[fourierderivative[[3 + 1, k + 1]]]^2)^(1/2) +
          2 γ2/Sqrt[samples] (Tr[{varmat[[2, ;;]]}ᵀ.{varmat[[2, ;;]]}.fouriermatrixsignal[[k + 1]]] +
              2 fouriervectorderivativesignal[[3 + 2, k + 1]].varmat[[2, ;;]] +
              Abs[fourierderivative[[3 + 2, k + 1]]]^2)^(1/2) + 2 γ3/Sqrt[samples]
          (Tr[{varmat[[3, ;;]]}ᵀ.{varmat[[3, ;;]]}.fouriermatrixsignal[[k + 1]]] +
              2 fouriervectorderivativesignal[[3 + 3, k + 1]].varmat[[3, ;;]] +
              Abs[fourierderivative[[3 + 3, k + 1]]]^2)^(1/2)),
     γ1 ≥ 0 && γ2 ≥ 0 && γ3 ≥ 0 && c1 ≥ 0 && c2 ≥ 0 && c3 ≥ 0 && a1 ≥ 0 && a2 ≥ 0 && a3 ≥ 0},
    {a1, a2, a3, b1, b2, b3, c1, c2, c3, γ1, γ2, γ3}], {k, 0, 50}];
refreqreg = Table[{k, relocalizationreg[[k + 1, 1]]}, {k, 0, 50}];
```

In[49]:=

```
ListPlot[refreqreg, PlotRange → All, PlotStyle → Black, ImageSize → 570]
```

tracé de liste · zone de tracé · tout · style de tracé · noir · taille d'image

Out[49]=



Estimated Matrix & Amplitude

In[50]:=

```
frequest = 8;
Grid[{{MatrixForm[varmat /. relocalizationreg[[frequest + 1, 2]]],
   "(γ1,γ2,γ3) = " ~~ ToString[{γ1, γ2, γ3} /. relocalizationreg[[frequest + 1, 2]]]}}]
```

grille · apparence matricielle · convertis en chaîne de caractères

Out[50]=

$$\begin{pmatrix} 24.473 & -2.59643 & 2.3902 & 0.0290794 & 0 & 0 \\ -2.59643 & 1.01942 & -0.585697 & 0 & 1.16075 & 0 \\ 2.3902 & -0.585697 & 2.23086 & 0 & 0 & 10.1016 \end{pmatrix}$$

$(\gamma_1,\gamma_2,\gamma_3)$ = {0.909947, 0.10854

Ground Truth

In[51]:=

```
Grid[{{MatrixForm[N@H[[4 ;; 6, ;;]]], "γ = 1"}}]
```

grille · apparence m·· · valeur numérique

Out[51]=

$$\begin{pmatrix} 24.5294 & -2.35294 & 2.35294 & 0.1 & 0. & 0. \\ -2.35294 & 1.23529 & -0.235294 & 0. & 1. & 0. \\ 2.35294 & -0.235294 & 1.23529 & 0. & 0. & 10. \end{pmatrix}$$

γ = 1