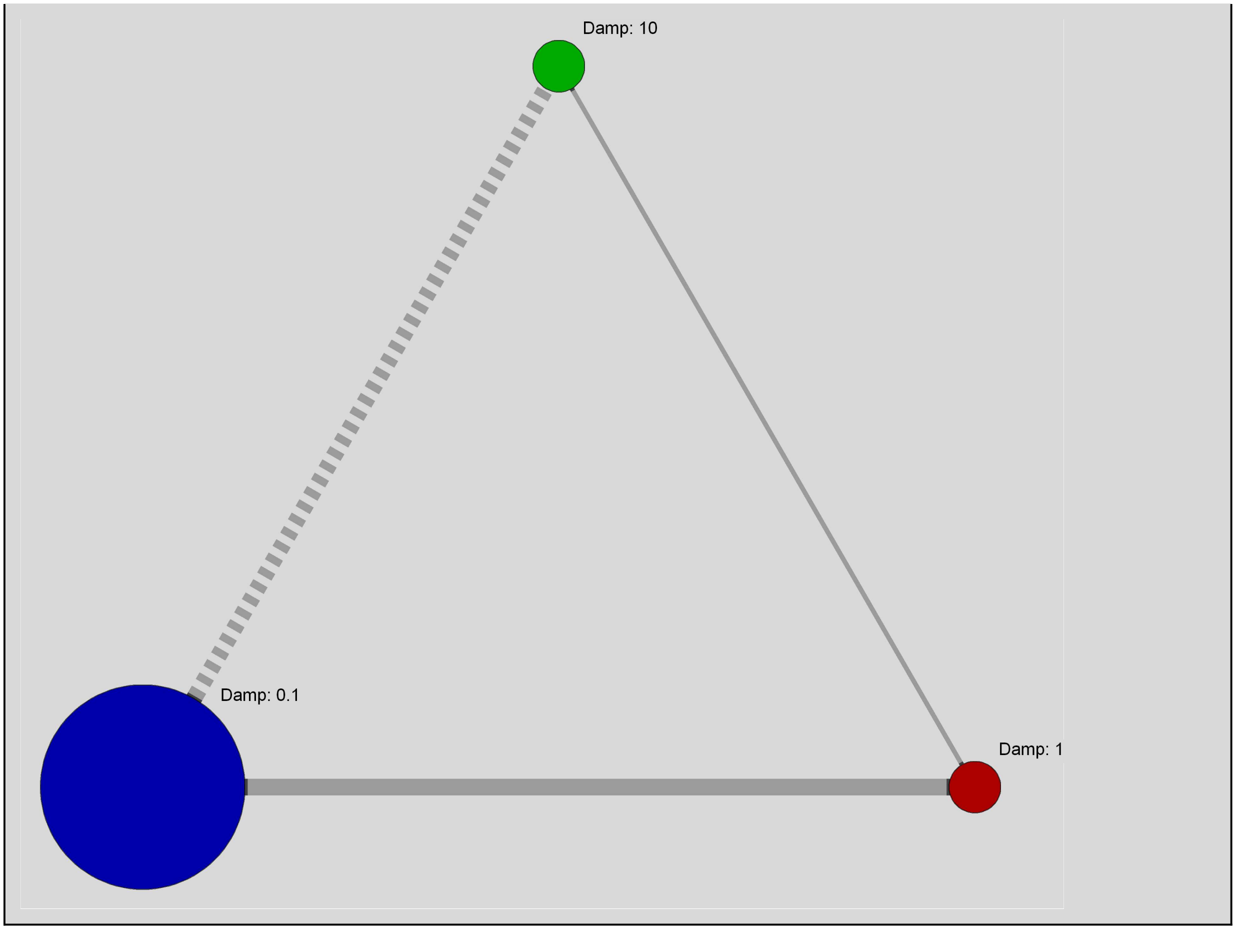
Synthetic Test Case Overview

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      plus fo··· bleu  style     plus fo··· rouge style    plus fo··· vert
 {Style[1 ⟷ 2, Thickness[40/17 edgsca1], Gray], Style[2 ⟷ 3, Thickness[4/17 edgsca2], Gray],
  style           épaisseur                gris   style           épaisseur               gris
  Style[3 ⟷ 1, Thickness[40/17 edgsca1], Dashed, Gray]}, VertexShapeFunction → "Circle",
  style           épaisseur                rayé    gris     fonction de forme de noeud   cercle
 VertexSize → {1 -> 417/17 versca1, 2 -> 21/17 versca2, 3 -> 21/17 versca2},
 taille de noeud
 VertexLabels → {1 → "Damp: 0.1", 2 → "Damp: 1", 3 → "Damp: 10"}]
 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]:=

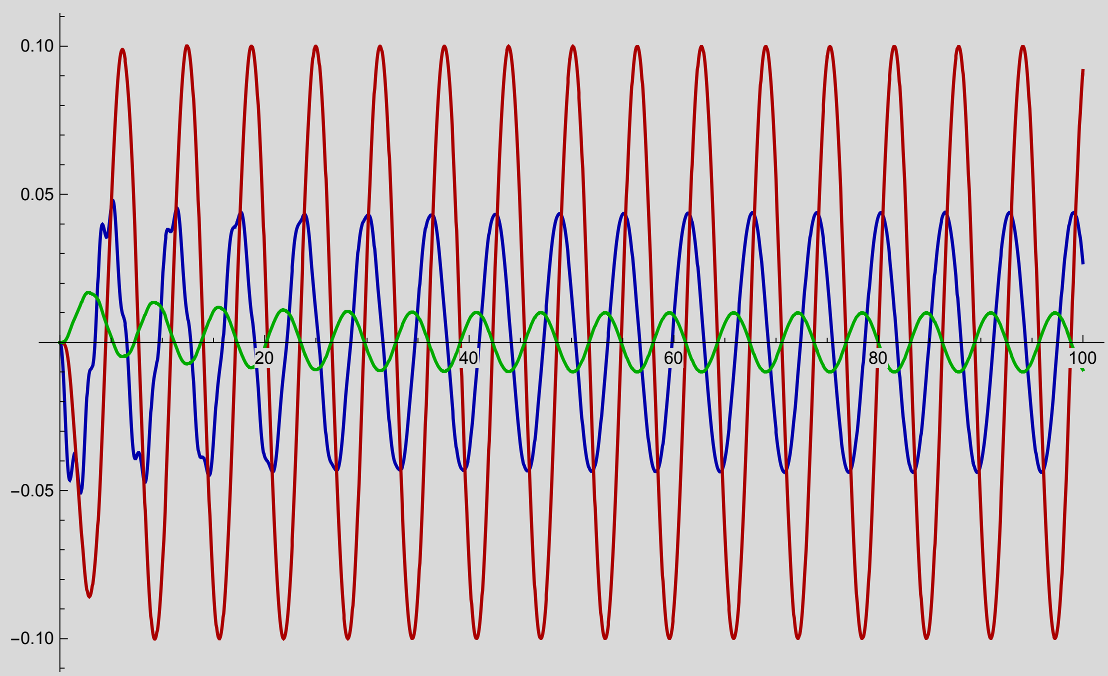
$$\text{eqnham} = \text{Flatten}\left[\text{FullSimplify}\left[\begin{pmatrix} x_1'[t] \\ x_2'[t] \\ x_3'[t] \\ p_1'[t] \\ p_2'[t] \\ p_3'[t] \end{pmatrix} + H.\begin{pmatrix} x_1[t] \\ x_2[t] \\ x_3[t] \\ p_1[t] \\ p_2[t] \\ p_3[t] \end{pmatrix}\right]\right];$$

aplatis simplifie complètement

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]=



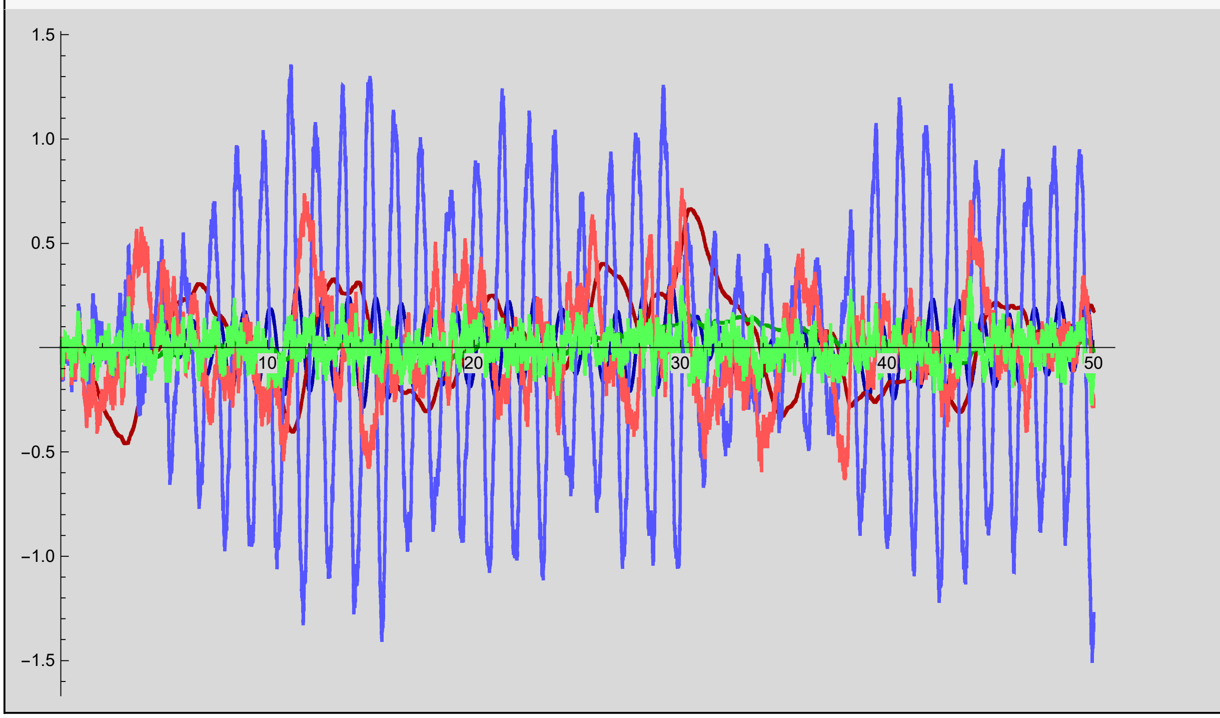
Stochastic ODE & Data Generation

In[65]:=

```
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[65]=



Discrete Frequencies

In[8]:=

```
N@ time/(2 π) {1, 5}
```

valeur numérique

Out[8]=

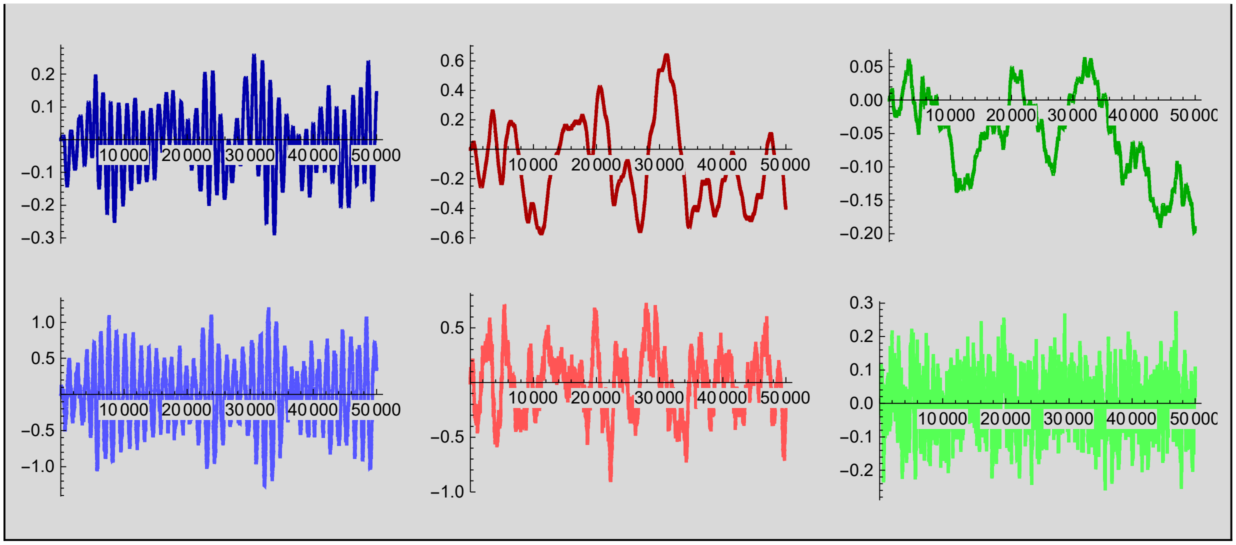
```
{7.95775, 39.7887}
```

In[66]:=

```
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
```

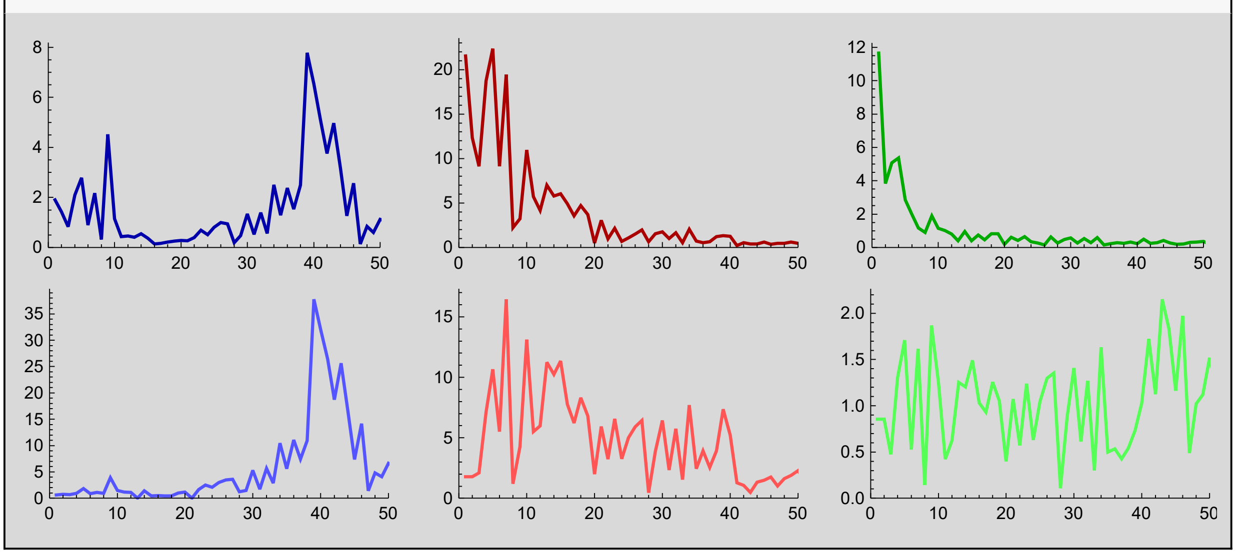
Out[66]=



In[67]:=

```
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
```

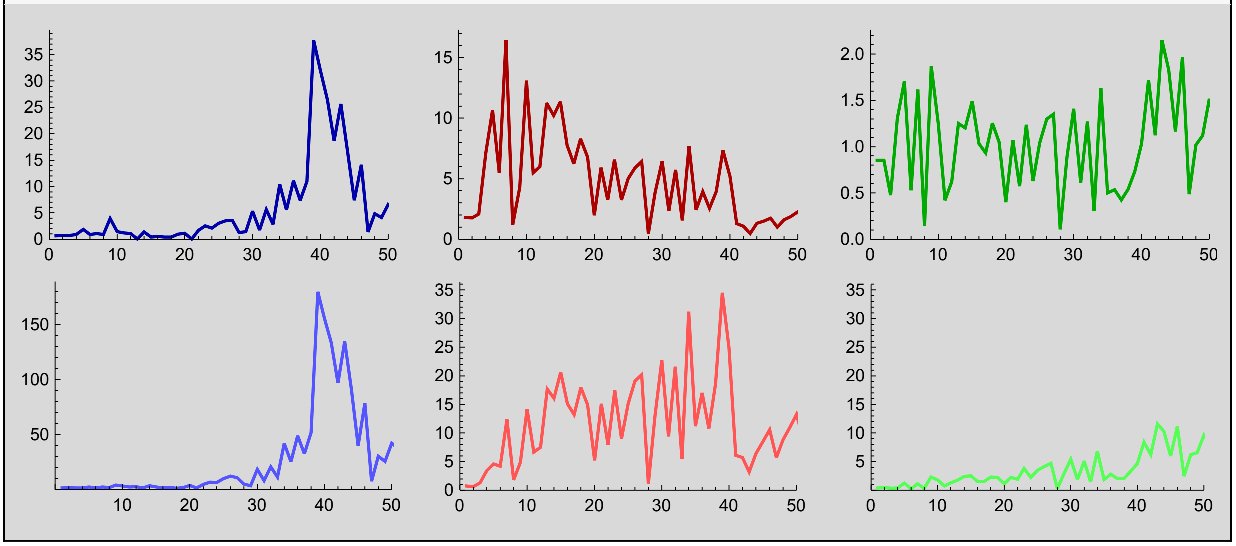
Out[67]=



In[68]:=

```
GraphicsGrid[Partition[
grille de graphi··· partitionne
  Table[ListPlot[{Abs[fourierderivative[[i]]]}, PlotRange → {{0, 50}, {0, All}},
        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
```

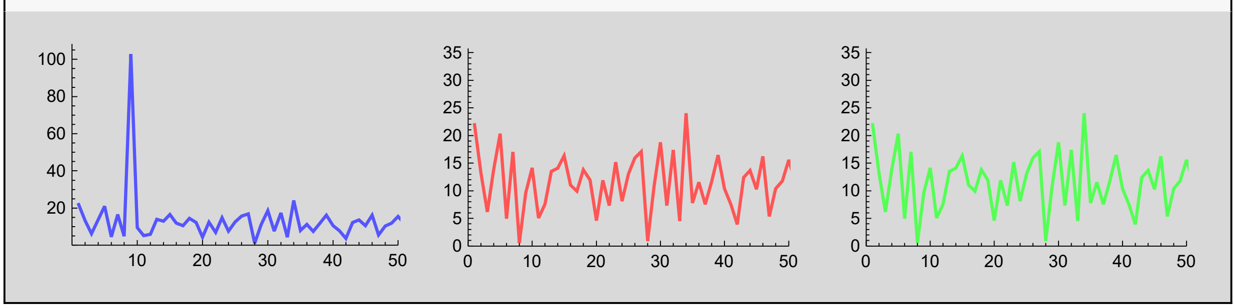
Out[68]=



In[69]:=

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

Out[69]=



Log-Likelihood Regression

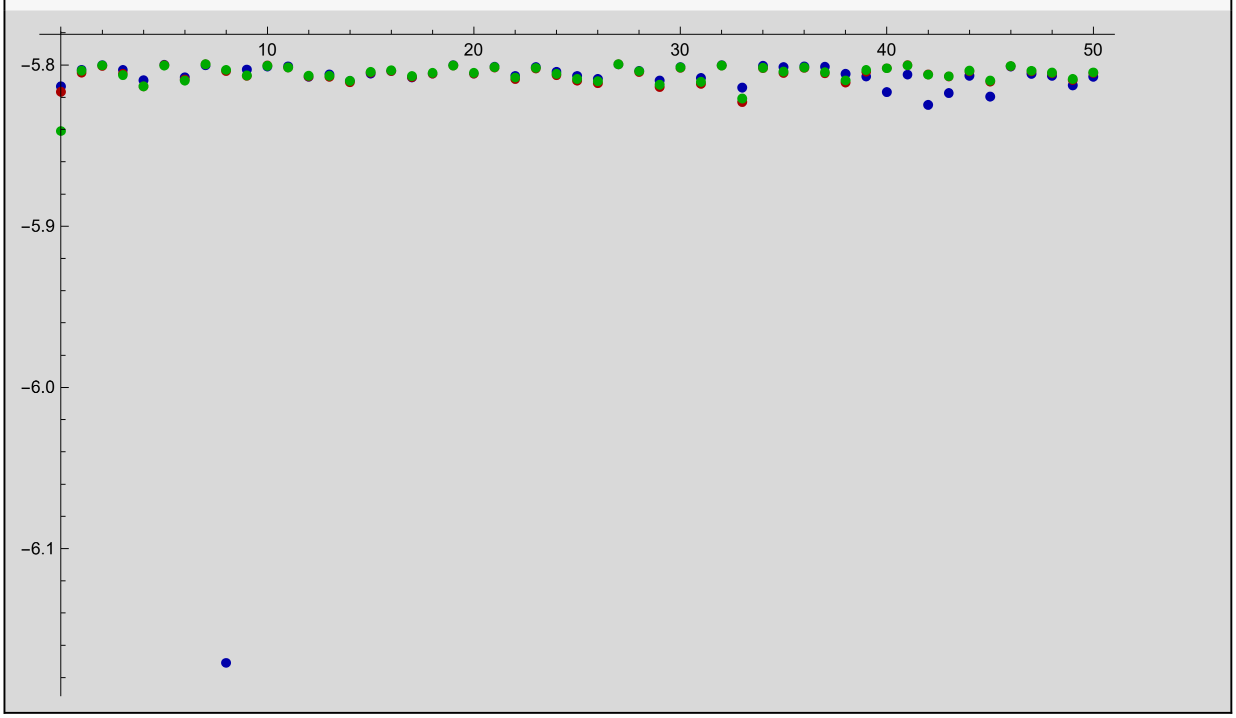
In[73]:=

```
localizationreg = Table[varmat = {{a1, b1, b2, c1, 0, 0}, {b1, a2, b3, 0, c2, 0}, {b2, b3, a3, 0, 0, c3}};
   NMinimize[{Tr[varmat^T.varmat.matrixsignal] + 2 Tr[varmat.matrixderivativesignal] +
      1/2 γ^2 - (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[74]:=

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

Out[74]=



Estimated Matrix & Amplitude

In[75]:=
```
siteest = 1;
frequest = 8;
Grid[{{MatrixForm[varmat /. localizationreg[[siteest, frequest + 1, 2]]],
   "γ = " ~~ ToString[γ /. localizationreg[[siteest, frequest + 1, 2]]]}}]
```
grille / apparence matricielle / convertis en chaîne de caractères

Out[75]=

$$\begin{pmatrix} 23.9492 & -2.41168 & 1.90936 & 0.172478 & 0 & 0 \\ -2.41168 & 1.09525 & -0.287662 & 0 & 0.881968 & 0 \\ 1.90936 & -0.287662 & 0.724023 & 0 & 0 & 9.27058 \end{pmatrix} \quad \gamma = 0.897702$$

Ground Truth

In[76]:=
```
Grid[{{MatrixForm[N@H[[4 ;; 6, ;;]]], "γ = 1"}}]
```
grille / apparence m·· / valeur numérique

Out[76]=

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

Log-Likelihood Spatial Relaxation

In[77]:=

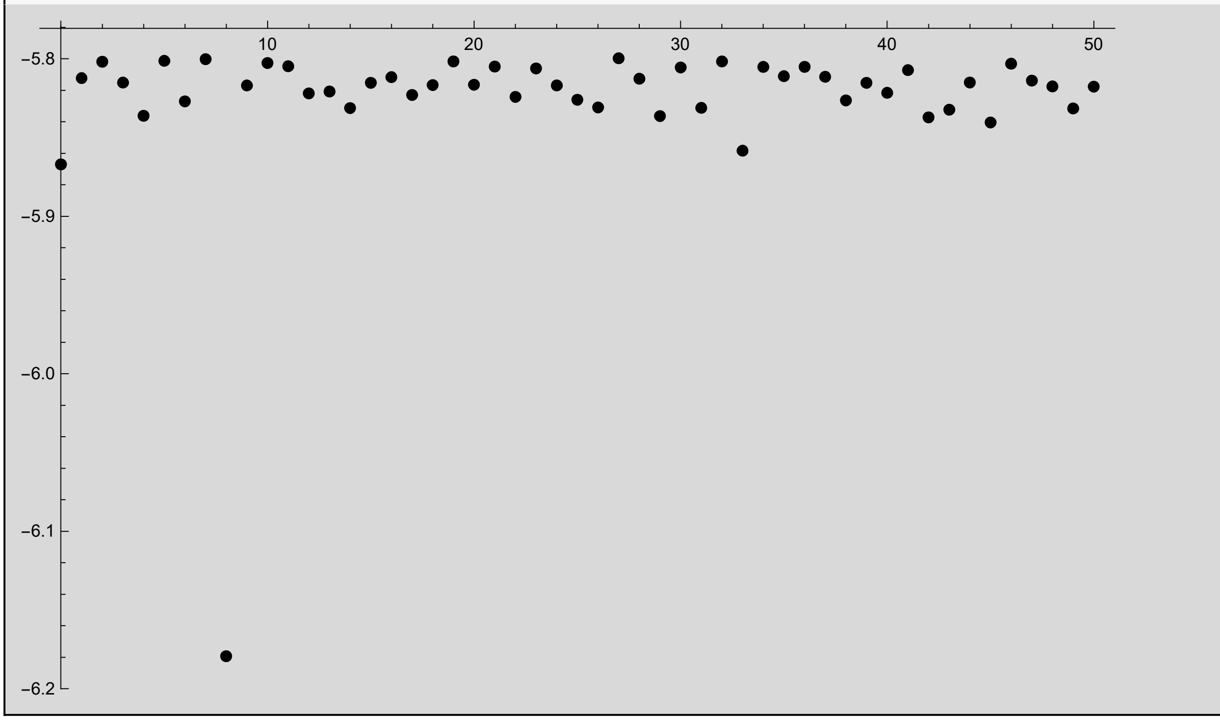
```
relocalizationreg = Table[varmat = {{a1, b1, b2, c1, 0, 0}, {b1, a2, b3, 0, c2, 0}, {b2, b3, a3, 0, 0, c3}};
  NMinimize[{Tr[varmat^T.varmat.matrixsignal] + 2 Tr[varmat.matrixderivativesignal] +
      1/2 (γ1^2 + γ2^2 + γ3^2) - (2 γ1/Sqrt[samples] (Tr[{varmat[[1, ;;]]}^T.{varmat[[1, ;;]]}.
              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[78]:=

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

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

Out[78]=



Estimated Matrix & Amplitude

In[79]:=

```
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[79]=

$$\begin{pmatrix} 23.9669 & -2.41341 & 1.80327 & 0.172466 & 0 & 0 \\ -2.41341 & 1.08784 & -0.30075 & 0 & 0.872284 & 0 \\ 1.80327 & -0.30075 & 0.832256 & 0 & 0 & 9.37652 \end{pmatrix}$$

(γ1,γ2,γ3) = {0.898
0.0926842, 0.094

Ground Truth

In[80]:=

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

grille | apparence m·· | valeur numérique

Out[80]=

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