Fourier series coefficients of a rectangular pulse signal

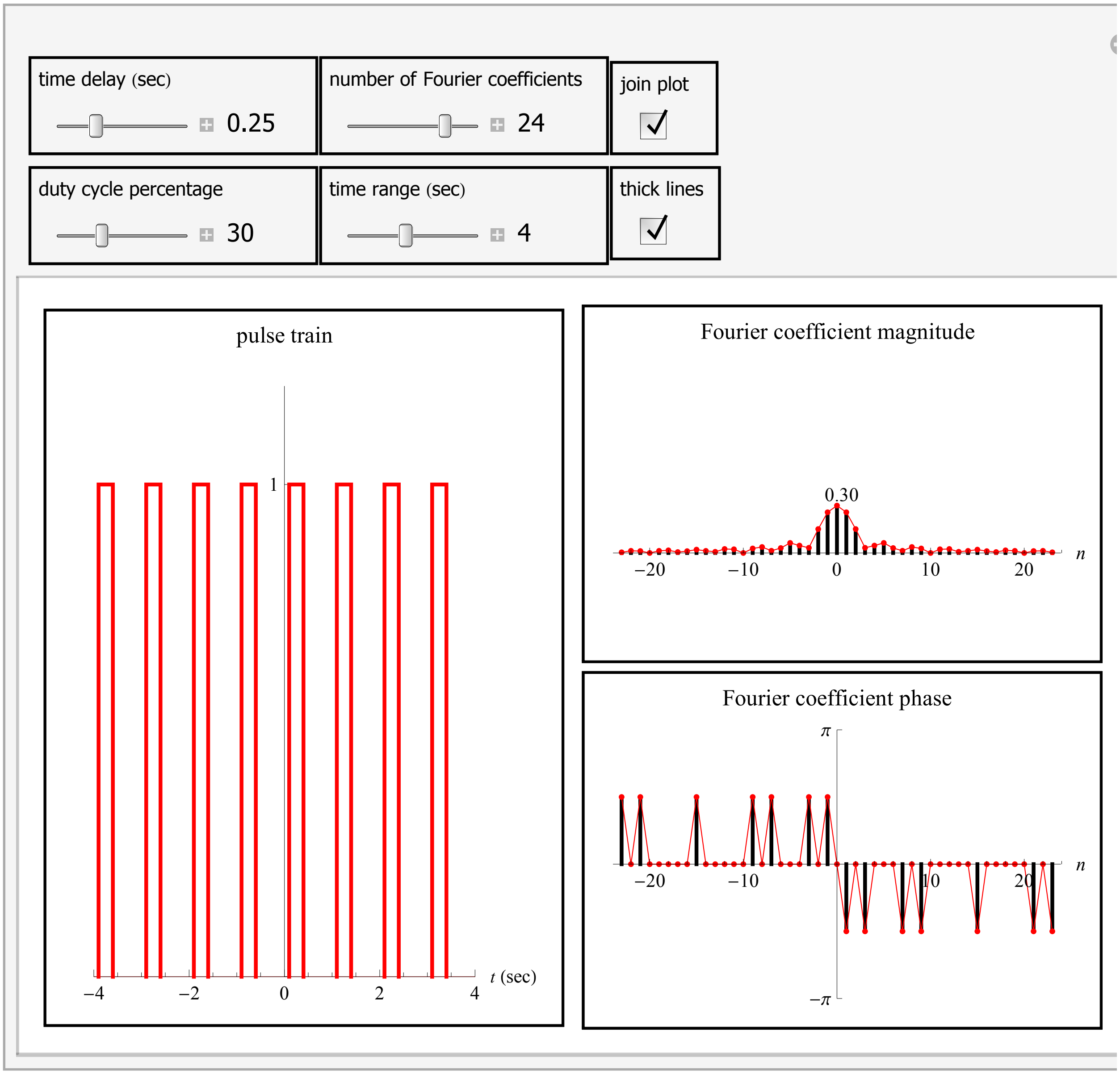
Initialization Code (optional)

Manipulate

```
Manipulate[
 (
  currentPulses = makePulseTrainDutyCycle[t0, 1, range, dutyCycle / 100, 1];
  fourierCoeffLines = getMag[t0, 1, range, dutyCycle / 100., nCoeff];

  currentMag = ListPlot[fourierCoeffLines,
    Evaluate[plotOptions],
    FillingStyle → If[thickLines, Thick, Thin],
    PlotLabel → Style["Fourier coefficient magnitude", 12],
    PlotRange → {Automatic, {-.5, 1.2}},
    Ticks → {Automatic, {-.5, 0, .5, 1}},
    Axes → {True, False}, Epilog →
     Text[ numIt[fourierCoeffLines[[nCoeff, 2]], 4, 2, 10], {0, fourierCoeffLines[[nCoeff, 2]]}, {0, -1}]];

  If[joinPlot && nCoeff > 1,
   currentMag = Show[currentMag, Plot[Interpolation[fourierCoeffLines, InterpolationOrder → 1][x],
      {x, -(nCoeff - 1), (nCoeff - 1)}, PlotRange → {Automatic, {-.5, 1}}, PlotStyle -> Red]]];

  fourierCoeffPhaseLines = getPhase[t0, 1, range, dutyCycle / 100., nCoeff];

  currentPhase = ListPlot[fourierCoeffPhaseLines,
    Evaluate[plotOptions],
    FillingStyle → If[thickLines, Thick, Thin],
    PlotLabel → Style["Fourier coefficient phase", 12],
    PlotRange → {Automatic, {-Pi, Pi}},
    Ticks → {Automatic, {-Pi, Pi}}];

  If[joinPlot && nCoeff > 1,
   currentPhase = Show[currentPhase, Plot[Interpolation[fourierCoeffPhaseLines, InterpolationOrder → 1][x],
      {x, -(nCoeff - 1), (nCoeff - 1)}, PlotRange → {Automatic, {-Pi, Pi}}, PlotStyle -> Red]]];

  Grid[{
    {Framed@Plot[0, {x, -range, range},
       Evaluate[pulsePlotOptions],
       PlotLabel → Style["pulse train", 12],
       PlotRange → {{-range, range}, {0, 1.2}},
       AxesLabel → {Row[{Style["t", Italic], Style[" (sec)", 10]}], None},
       Ticks → {Automatic, {0, 1}}, Epilog → {If[thickLines, Thick, Thin], Red, currentPulses}],
     Column[{Framed@currentMag, Framed@currentPhase}]}
   }, Spacings → 1, Alignment → Center]
 ),

 Item[Row[{

    Column[{
      Framed@Column[{Style["time delay (sec)", 10],
         Control[{ {t0, .25, ""}, 0, 1., .05, ImageSize → Tiny, Appearance → "Labeled"}]}],
      Framed@Column[{Style["duty cycle percentage", 10],
         Control[{ {dutyCycle, 30, ""}, 0, 100, .1, ImageSize → Tiny, Appearance → "Labeled"}]}]
     }],

    Column[{
      Framed@Column[{Style["number of Fourier coefficients", 10],
         Control[{ {nCoeff, 8, ""}, 1, 30, 1, ImageSize → Tiny, Appearance → "Labeled"}]}],

      Framed@Column[{Style["time range (sec)", 10],
         Control[{ {range, 4, ""}, 1, 8, .1, ImageSize → Tiny, Appearance → "Labeled"}]}]
     }],

    Column[{
      Framed@Column[{Style["join plot   ", 10],
         Control[{ {joinPlot, True, ""}, {True, False}, ControlType → Checkbox}]}],

      Framed@Column[{Style["thick lines ", 10],
         Control[{ {thickLines, True, ""}, {True, False}, ControlType → Checkbox}]}]
     }]

   }, Alignment → Top], ControlPlacement → Top],

 Control[{ {currentPulses, 0}, ControlType → None}],
 Control[{ {x, 0}, ControlType → None}],
 Control[{ {currentCoeff, 0}, ControlType → None}],
 Control[{ {currentMag, 0}, ControlType → None}],
 Control[{ {currentPhase, 0}, ControlType → None}],
 Control[{ {fourierCoeffLines, 0}, ControlType → None}],
 Control[{ {fourierCoeffPhaseLines, 0}, ControlType → None}],
 Control[{ {pulsePlotOptions, 0}, ControlType → None}],
 Control[{ {imageSize, 260}, ControlType → None}],
 Control[{ {pulseAspectRatio, 1.55}, ControlType → None}],
 Control[{ {aspectRatio, .6}, ControlType → None}],

 TrackedSymbols :→ {t0, dutyCycle, range, nCoeff, joinPlot, thickLines},
 AutorunSequencing → {1, 2, 3},

 Initialization :→
  {
   pulsePlotOptions = {ImageMargins → 1, ImagePadding → {{20, 40}, {20, 20}},
     AspectRatio → pulseAspectRatio,
     PlotStyle → {Thin, Red}, TicksStyle → Directive[10], ImageSize → imageSize};

   plotOptions = {Filling → Axis, ImageMargins → 1, ImagePadding → {{10, 15}, {10, 10}},
     AspectRatio → aspectRatio, PlotStyle → {Thin, Red},
     TicksStyle → Directive[10], ImageSize → imageSize, AxesLabel → {Style["n", Italic, 10], None}};

   (*------------------------------*)
   (*        formats a number     *)
   (*------------------------------*)
   numIt[v_, s1_, s2_, fontSize_] := Module[{},
     Style[ToString[
       AccountingForm[Chop[v], {s1, s2}, NumberPadding → {" ", "0"}, NumberSigns → {"-", ""}]], fontSize]
    ];
   (*------------------------------*)
   (*  decide on phase signa      *)
   (*------------------------------*)
   getPhase[n_, cn_, t0_, period_] := Module[{phase},
     If[n == 0, Return[0]];
     If[cn == 0, Return[0]];
     If[t0 == 0, Return[If[cn < 0, Sign[n] * Pi, 0]]];
```

```
                 2 Pi
  phase = -n * ───────── t0;
               period
  If[cn ≥  0 && Abs[phase] > 0, Return[Mod[phase, Sign[phase] * Pi]], Return[0]];
                 2 Pi
  phase = -n * ───────── t0 + Sign[n] * Pi;
               period
  If[Abs[phase] > 0, Return[Mod[phase, Sign[phase] * Pi]], Return[Sign[n] * Pi]];

 ];
(*-----------------------------*)
(* find the magnitude of cn    *)
(*-----------------------------*)
getMag[t0_?NumericQ, period_?NumericQ,
  range_?NumericQ, dutyCycle_?NumericQ, nCoeff_Integer] := Module[{data, n, h = 1},

  data = {{0, h * dutyCycle}};
  data =
   Append[List@Reverse@Table[{n, Abs[h * dutyCycle * Sinc[Pi * n * dutyCycle] * Exp[-I * 2 * Pi / period * n * t0]]},
        {n, -1, -(nCoeff - 1), -1}], data];

  data = Append[data, Table[
     {n, Abs[h * dutyCycle * Sinc[Pi * n * dutyCycle] * Exp[-I * 2 * Pi / period * n * t0]]}, {n, 1, nCoeff - 1, 1}]];
  Chop@Flatten[data, 1]
 ];
(*-----------------------------*)
(* find the phase of cn        *)
(*-----------------------------*)
getPhase[delay_?NumericQ, period_?NumericQ,
  range_?NumericQ, dutyCycle_?NumericQ, nCoeff_Integer] := Module[{data, n, t0},

  t0 = Mod[delay, period];
  data = {{0, 0}};

  data = Append[List@
     Reverse@Table[{n, getPhase[n, Sinc[Pi n dutyCycle], t0, period]}, {n, -1, -(nCoeff - 1), -1}], data];

  data = Append[data, Table[{n, getPhase[n,  Sinc[Pi n dutyCycle], t0, period]}, {n, 1, nCoeff - 1, 1}]];

  Chop@Flatten[data, 1]
 ];
(*-----------------------------*)
(* build the pulse train      *)
(*-----------------------------*)
makePulseTrainDutyCycle[delay_?NumericQ, period_?NumericQ, range_?NumericQ,
  dutyCycle_?NumericQ, h_?NumericQ] := Module[{allPulses, tao = period * dutyCycle},

  getForwardPulses[] := Module[{n, z, ok = True, pulse, forwardPulses = {}, more, t0},
    t0 = Mod[delay, period];

    n = 0;
    more = True;
    While[more,
     {
      z = t0 + n * period;

      If[z > range, more = False,
       {
        pulse = Line[{ {z - tao / 2, 0}, {z - tao / 2, h}, {z + tao / 2, h}, {z + tao / 2, 0}}];
        forwardPulses = Append[forwardPulses, pulse];
        n = n + 1;
       }]
     }
```

```
       ];

       forwardPulses
      ];

     getBackPulses[] := Module[{n, z, ok = True, pulse, backPulses = {}, more, t0},
       t0 = Mod[delay, period];
       n = 0;
       more = True;
       While[more,
        {
         z = -(period - t0) - n*period;

         If[Abs[z] > range, more = False,
          {
           pulse = Line[{ {z + tao / 2, 0}, {z + tao / 2, h}, {z - tao / 2, h}, {z - tao / 2, 0}}];
           backPulses = Append[backPulses, pulse];
           n = n + 1;
          }]
        }
       ];

       backPulses
      ];

     allPulses = getForwardPulses[];
     allPulses = Append[allPulses, getBackPulses[]]
    ];
  }
]
```



time delay (sec) 0.25

number of Fourier coefficients 24

join plot ☑

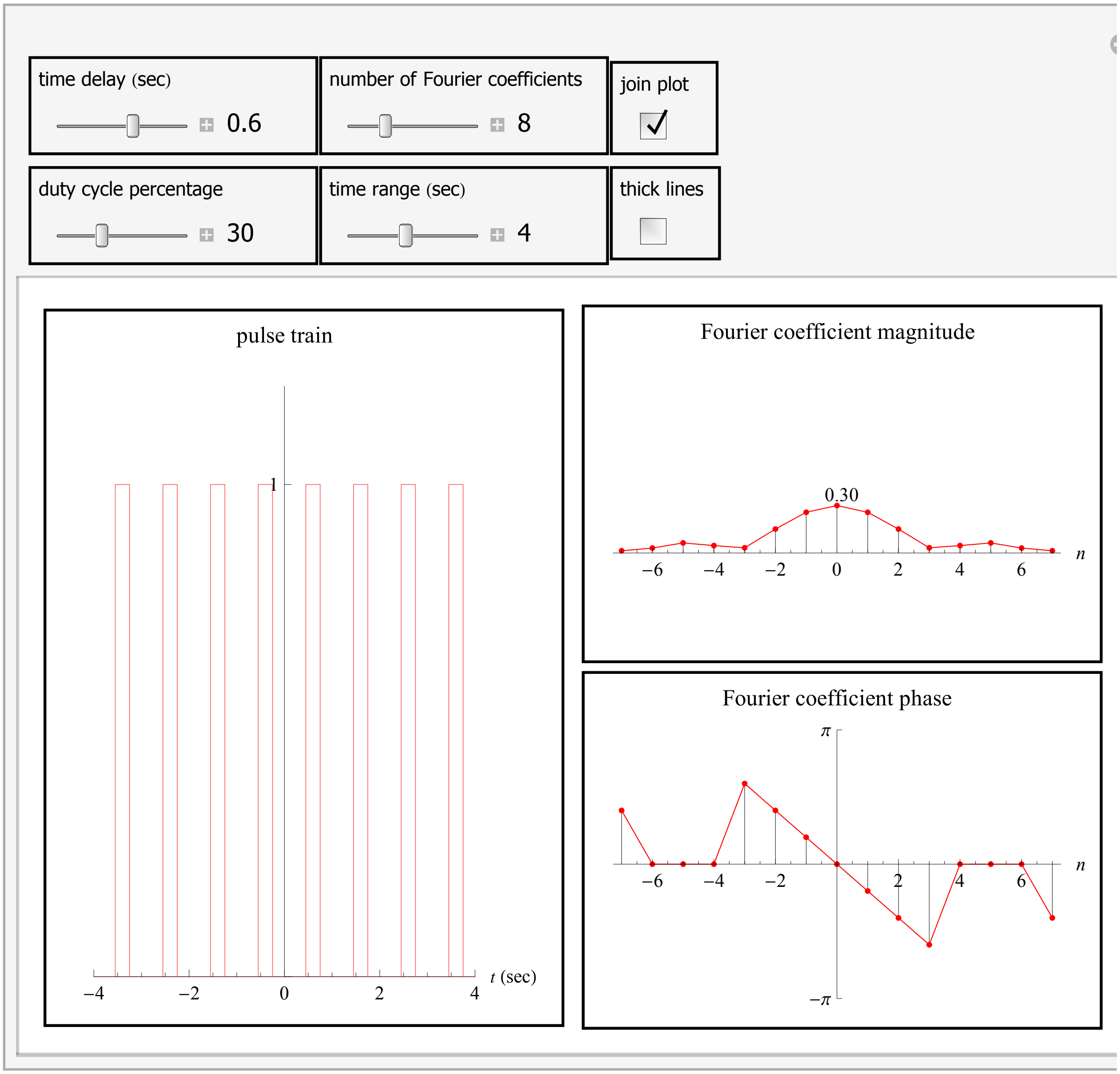
duty cycle percentage 30

time range (sec) 4

thick lines ☑

Caption

This demonstration calculates and plots the magnitude and phase of the Fourier coefficients for a rectangular pulse train signal. A rectangular pulse is defined by its duty cycle (the ratio of the width of the rectangle to its period), and by the delay of the pulse. In this demonstration, the pulse period is fixed at one second, and the height is fixed at unity. The delay and the duty cycle can be adjusted as well as the number of fourier coefficients. We notice that, since the signal is a real signal, the magnitude plot is an even function and the phase plot is an odd function.



time delay (sec) 0.6

number of Fourier coefficients 8

join plot ☑

duty cycle percentage 30

time range (sec) 4

thick lines □

Details (optional)

The n^{th} Fourier coefficient of a rectangular pulse train is given by $c_n = h\, d\, \mathrm{Sinc}[n\, d]\, \mathrm{Exp}\left[-I\, \frac{2\pi}{T_o}\, n\, t_0\right]$ where h is the pulse height, d is the duty cycle, T_0 is the period of the pulse train, t_0 is the delay of the pulse in seconds. $\mathrm{Sinc}[x]$ is defined as $\frac{\mathrm{Sin}[\pi\, x]}{\pi\, x}$. This demonstration displays the magnitude and phase of c_n.

Control Suggestions (optional)

□ Slider Zoom

□ Drag Locators

□ Rotate and Zoom in 3D

□ Automatic Animation

□ Gamepad Controls

□ Resize Images

□ Bookmark Animation

Search Terms (optional)

rectangular pulse
Fourier series coefficient
phase
Sinc

Related Links (optional)

rectangle pulse

Authoring Information

Contributed by: Nasser M. Abbasi