

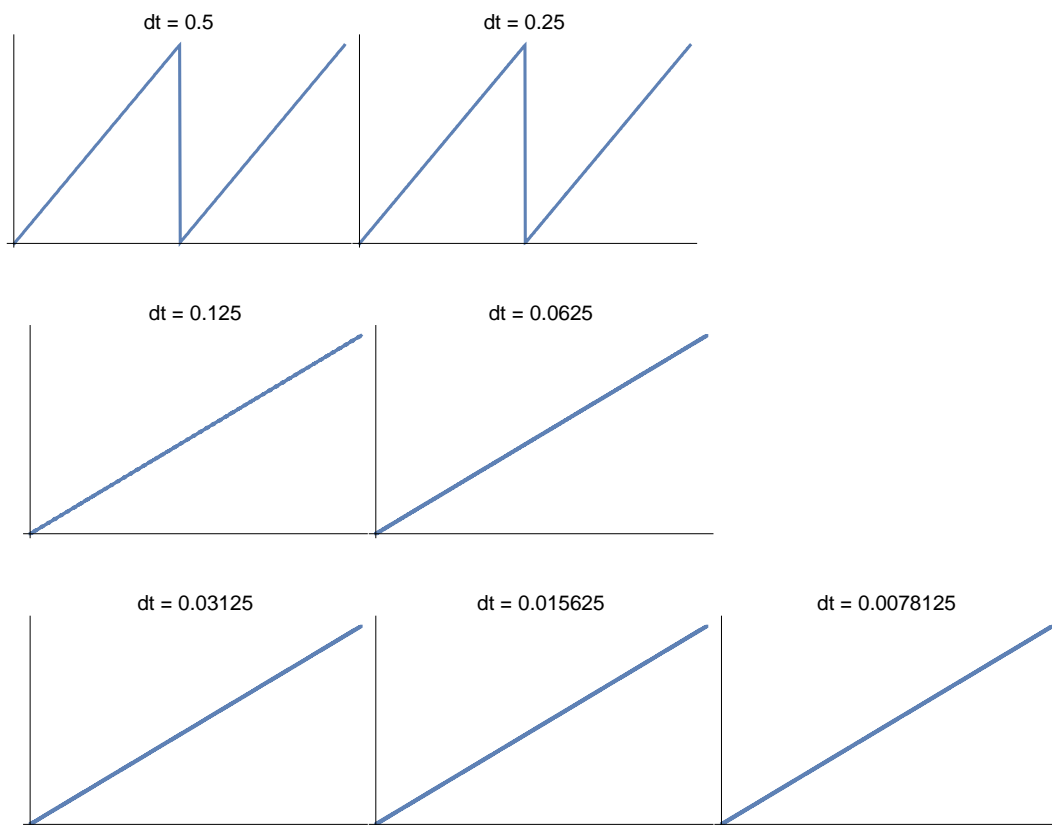
# Rounding Issues in Simple Euler Integration

---

## Using Machine Precision

### Basic Formulation

```
Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1. / 2i;
  inflow[_] := 1. / 100;
  outflow[t_] := If[stock[t] == 1.,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0.];
  stock[0.] := 0.;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0., 200., dt}
    ],
    Ticks → None,
    PlotLabel → "dt = " <> ToString[dt],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row
```



## Using Approximate Equality-Check

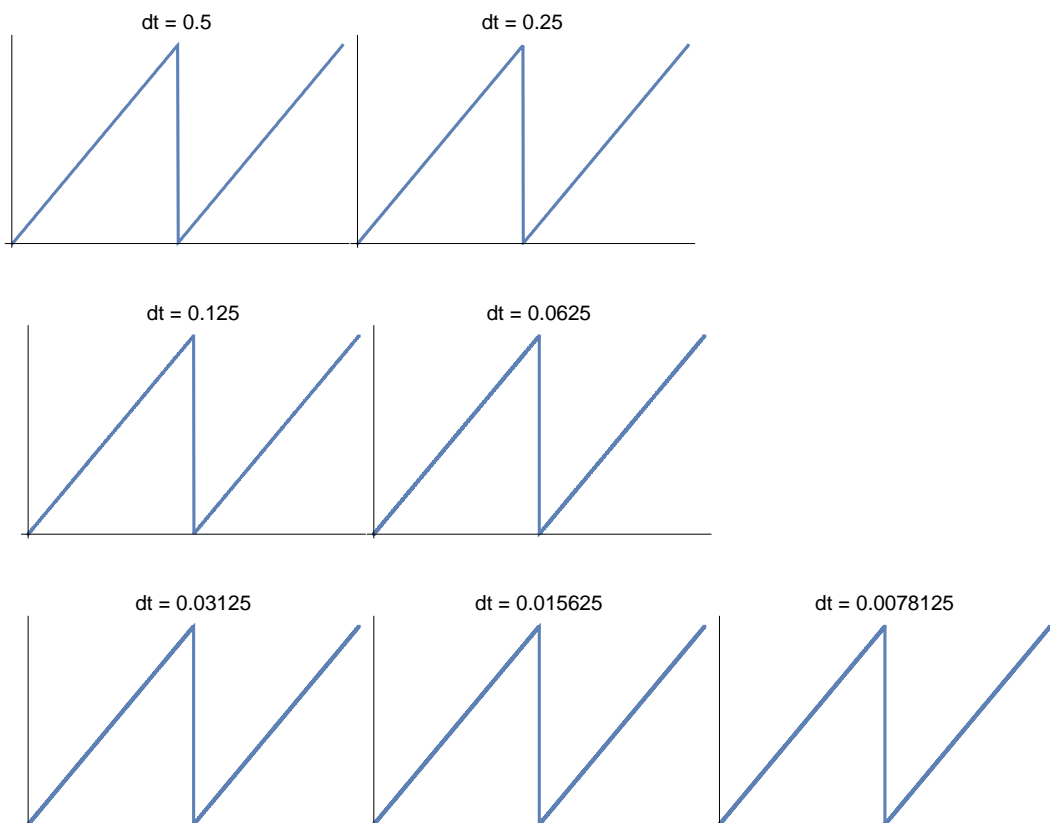
### ? Chop

Chop[*expr*] replaces approximate real numbers in *expr* that are close to zero by the exact integer 0. >>

```

Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1. / 2i;
  inflow[_] := 1. / 100;
  outflow[t_] := If[Chop[stock[t] - 1.] == 0.,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0.];
  stock[0.] := 0.;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0., 200., dt}
    ],
    Ticks → None,
    PlotLabel → "dt = " <> ToString[dt],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row

```



## Using Higher Precision

```
Table[
  Clear[dt, stock, inflow, outflow];
  dt := 1 / 2i;
  inflow[_] := 1 / 100;
  outflow[t_] := If[stock[t] == 1,  $\frac{\text{stock}[t]}{\text{dt}}$ , 0];
  stock[0] := 0;
  stock[t_] := stock[t] = stock[t - dt] + (inflow[t - dt] - outflow[t - dt]) × dt;
  ListLinePlot[
    Table[
      stock[τ],
      {τ, 0, 200, dt}
    ],
    Ticks → None,
    PlotLabel → "dt = 1 / " <> ToString[2i],
    ImageSize → Small
  ],
  {i, 1, 7}
] // Row
```

