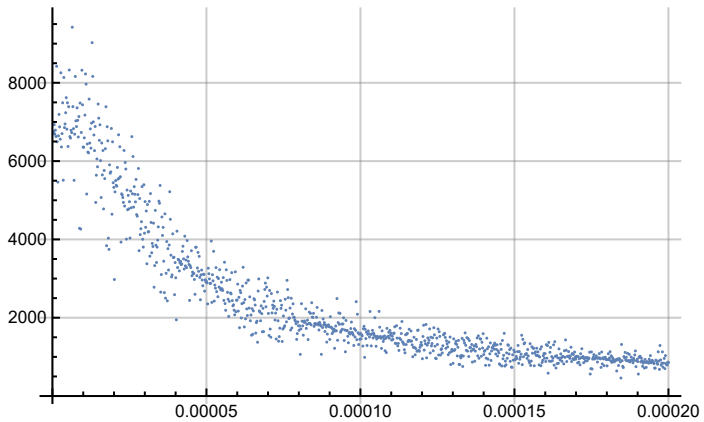
Kurzschluß-Schaltvorgang der geladenen, realen Koax-Leitung RG213/U nach dem numerischen Talbot-Verfahren
jeweils der Graph mit den Lösungspunkten und der Graph als kubischer Spline

In[16]:=
```
ta = UnixTime[];
     Unixzeit
ClearAll[U, Z0, Z1, Z2, l, Cs, Rs, lap, M, Zeit, Liste, te, td, Gs, Ls];
lösche alle
U = 100; Z0 = 50; Z1 = 0.01; Z2 = 1*^12; l = 1*^2; Cs = 101.049872*^-12; Rs = 6.56167979*^-3; x = 0;
Ls = Z0^2 * Cs;
Gs = Rs * Cs / Ls;
td = 2 * l * Sqrt[Ls * Cs];
             Quadratwurzel
Talbot[Fs_, t_, N1_] := Module[{h, shift, ans, theta, k, z, dz},
                        Modul
  h = 2 * Pi / N1;
          Kreiszahl π
  shift = 0;
  ans = 0;
  For[k = 0, k ≤ N1, k++,
  For–Schleife
   theta = -Pi + (k + 1 / 2) * h;
             Kreiszahl π
   z = shift + N1 / t * (0.5017 * theta * Cot[0.6407 * theta] - 0.6122 + 0.2645 * I * theta);
                                          Kotangens                                imaginäre Einheit I
   dz = N1 / t * (-0.5017 * 0.6407 * theta / Sin[0.6407 * theta]^2 + 0.5017 * Cot[0.6407 * theta] + 0.2645 * I);
                                             Sinus                          Kotangens                   imaginäre Einheit I
   ans = ans + Exp[z * t] * Fs[z] * dz;];
               Exponentialfunktion
  Re[h / (2 * I * Pi) * ans]]
  Realteil     ⋯ Kreiszahl π

lap[p_] := -U / p * (Cosh[Sqrt[(Rs + p * Ls) * (Gs + p * Cs)] * (l - x)] + Z2 / Z0 * Sinh[Sqrt[(Rs + p * Ls) * (Gs + p * Cs)] * (l - x)]) /
                     Kos⋯ Quadratwurzel                                    Sinu⋯ Quadratwurzel
    ((Z1 + Z2) * Cosh[Sqrt[(Rs + p * Ls) * (Gs + p * Cs)] * l] + (Z0 + Z1 * Z2 / Z0) * Sinh[Sqrt[(Rs + p * Ls) * (Gs + p * Cs)] * l]);
                 Kos⋯ Quadratwurzel                                    Sinu⋯ Quadratwurzel
```

```
M = 1024; Talits = 256; Zeit = 1 / 5000;
Liste = Table[{Zeit / M * i, Talbot[lap, Zeit / M * i, Talits]}, {i, 1, M}];
        Tabelle
ListPlot[Liste, PlotRange → All, GridLines → Automatic]
listenbezogene Gr··· Koordinatenb··· alle Gitternetzlinien automatisch
```

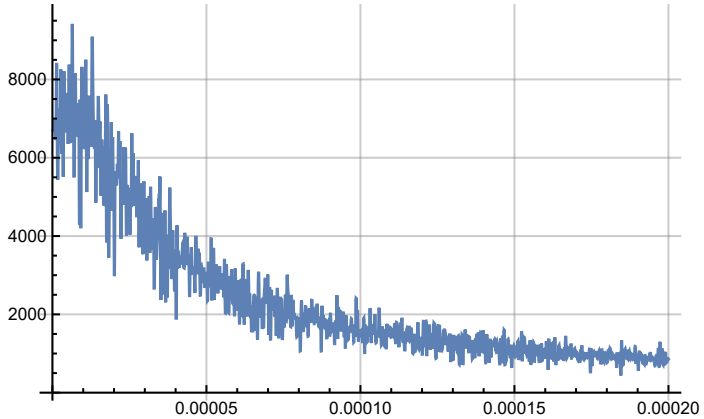
Out[26]=



In[27]:=

```
ListLinePlot[Liste, InterpolationOrder → 3, PlotRange → All, GridLines → Automatic]
listenbezogene Liniengra··· Ordnung der Interpolation Koordinatenb··· alle Gitternetzlinien automatisch
```

Out[27]=



In[28]:=
```
te = UnixTime[] - ta
```
Unixzeit

```
N[te / 60]
```
numerischer Wert

```
Max[Liste]
```
größtes Element

```
td
```

Out[28]= 12

Out[29]= 0.2

Out[30]= 9419.48

Out[31]= 1.0105×10^{-6}