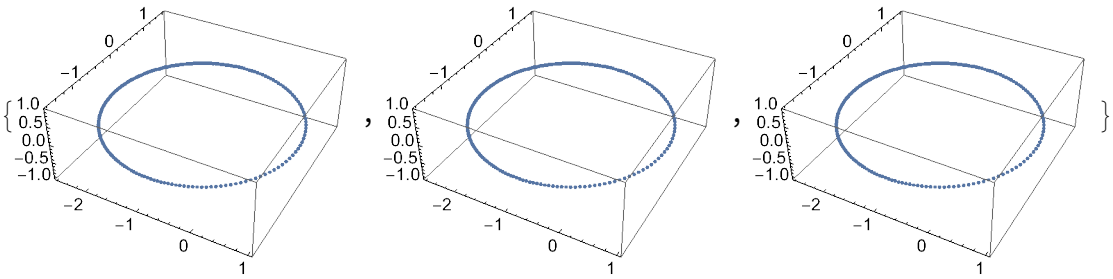
Лабораторная 4 (MPI + OMP)

In[1]:=
```
ClearAll["Global`*"];
SetDirectory[NotebookDirectory[]];
```

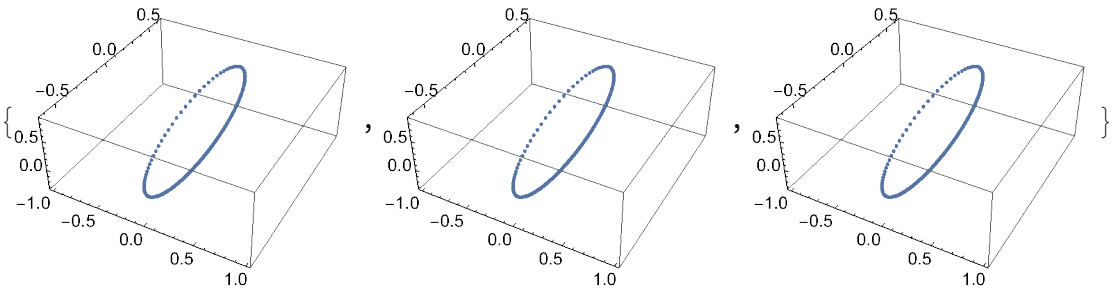
Проверка (+cuda)

In[3]:=
```
seq = Import["seq.txt", "Data"];
tSeq = seq[[ ;; , 1]];
xSeq = seq[[ ;; , 2 ;; 4]];
vSeq = seq[[ ;; , 13 ;; 15]];
mpi = Import["mpi.txt", "Data"];
tMpi = mpi[[ ;; , 1]];
xMpi = mpi[[ ;; , 2 ;; 4]];
vMpi = mpi[[ ;; , 13 ;; 15]];
cuda = Import["cuda.txt", "Data"];
tCuda = cuda[[ ;; , 1]];
xCuda = cuda[[ ;; , 2 ;; 4]];
vCuda = cuda[[ ;; , 13 ;; 15]];
On[Assert];
Assert[Norm[Flatten[seq[[ ;; , 2 ;;]]] - Flatten[mpi[[ ;; , 2 ;;]]]] == 0.0];
Assert[Norm[Flatten[seq[[ ;; , 2 ;;]]] - Flatten[cuda[[ ;; , 2 ;;]]]] == 0.0];
{ListPointPlot3D[xSeq], ListPointPlot3D[xMpi], ListPointPlot3D[xCuda]}
{ListPointPlot3D[vSeq], ListPointPlot3D[vMpi], ListPointPlot3D[vCuda]}
```

Out[18]=



Out[19]=



Порядок

In[20]:=
```
tauInit = 0.1;
orderExact = 2;
trajCnt = 4;
runsCnt = 5;
tau = tauInit / orderExact^# & /@ Range[0, runsCnt - 1];
data = Import[ToString[#] <> ".txt", "Data"] & /@ tau;
trajCalc = {data[[#, ;; , 2 ;; 4]], data[[#, ;; , 5 ;; 7]],
             data[[#, ;; , 8 ;; 10]], data[[#, ;; , 11 ;; 13]]} & /@ Range[runsCnt];
trajExact = Import["traj" <> ToString[#] <> ".txt", "Data"][[ ;; , 2 ;; 4]] & /@
        Range[trajCnt];
(*trajCalc[[4,1]]
  trajExact[[1]]*)
err = Norm[Flatten[trajExact] - Flatten[trajCalc[[#]]], 1] & /@ Range[runsCnt];
orderCalc = Log[err[[1 ;; runsCnt - 1]] / err[[2 ;; runsCnt]]] / Log[orderExact];
Grid[{{"τ", "Err = ||u - y_τ||", "Order= Log[||u - y_τ|| / ||u - y_τ/2||] / Log[2]"}} ~
     Join ~ Transpose@{tau, err, Append[orderCalc, "-"]}, Frame → All]
```

Out[30]=

τ	$\text{Err} = \|\|u - y_\tau\|\|$	$\text{Order}= \frac{\text{Log}\left[\frac{\|\|u - y_\tau\|\|}{\|\|u - y_{\tau/2}\|\|}\right]}{\text{Log}[2]}$
0.1	10.799	1.92766
0.05	2.83857	1.96654
0.025	0.726294	1.98743
0.0125	0.183162	2.00891
0.00625	0.0455086	–

Проверка для большой задачи

In[31]:=
```
seq = ToExpression[StringSplit[#]] & /@ Import["seq_10000.txt", "Data"];
xSeq = seq[[ ;; , 2 ;; 4]];
vSeq = seq[[ ;; , 23 ;; 25]];
mpi = ToExpression[StringSplit[#]] & /@ Import["mpi_10000.txt", "Data"];
xMpi = mpi[[ ;; , 2 ;; 4]];
vMpi = mpi[[ ;; , 23 ;; 25]];
On[Assert];
Assert[Norm[Flatten[seq[[ ;; , 2 ;;]]] - Flatten[mpi[[ ;; , 2 ;;]]]] == 0.0];
```

Сравнение результатов для различных пар (np, nth)

N = 10000

Time seq: 7.856 secs

(2 процесса на разных узлах)

np = 2

nth = 1
Time mpi: 4.162 secs
Acceleration: **1.88755**

(2 процесса на одном узле)
np = 2
nth = 1
Time mpi: 4.132 secs
Acceleration: **1.90126**

(3 процесса на разных узлах)
np = 3
nth = 1
Time mpi: 2.868 secs
Acceleration: **2.73919**

(3 процесса на одном узле)
np = 3
nth = 1
Time mpi: 2.749 secs
Acceleration: **2.85777**

(4 потока на 1 узле)
np = 1
nth = 4
Time mpi + omp: 2.06 secs
Acceleration: **3.81359**

(8 потоков на 2 узлах)
np = 2
nth = 4
Time mpi + omp: 1.283
Acceleration: **6.12315**

(12 потоков на 3 узлах)
np = 3
nth = 4
Time mpi + omp: 0.941
Acceleration: **8.34857**

Лабораторная 5 (CUDA)

Сравнение результатов для различных значений числа точек

(N = 5000)
Time exec: 2.78783
Time copy: 0.000324736
Time cuda: 1.525 (Tesla T4)
Time seq: 4.232
Acceleration: **2.7751**

(N = 20000)
Time exec: 10.7356
Time copy: 0.00101994
Time cuda: 5.561 (Tesla T4)
Time seq: 32.935
Acceleration: **5.9225**

(N = 50000)
Time exec: 52.5237
Time copy: 0.00201072
Time cuda: 26.536 (Tesla T4)
Time seq: 208.012
Acceleration: **7.83886**