Exercise_3

May 19, 2025

1 Advanced Deep Learning for Physics (IN2298)

2 Exercise 3: Sphere Packing

[ ]:
```
# Install PhyFlow and Import libraries
%%capture
!pip install --quiet phiflow;
!pip install nbconvert;
!apt-get install texlive texlive-xetex texlive-latex-extra pandoc;
from google.colab import drive
drive.mount("/content/drive");
from phi.torch.flow import *
```

2.1 [1] Setup

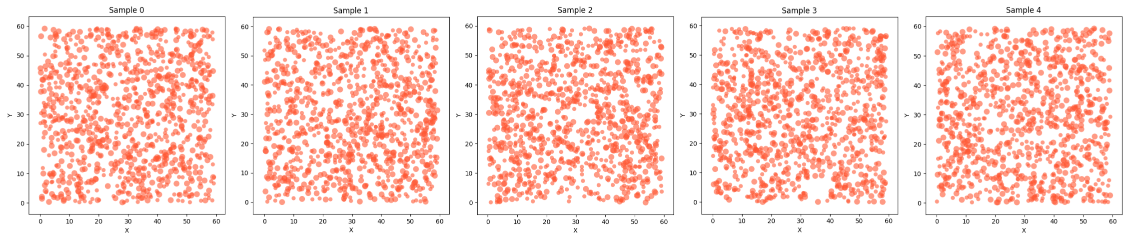
[ ]:
```
# We have 500 spheres of radius 1 and 500 spheres of radius 0.7
# let's calculate the total volume they occucpy:
n_sphere = 1000
r_sphere_big = 1.
r_sphere_small = .7
Area = 0.5*n_sphere*(r_sphere_big**2*math.pi)+0.
 ↪5*n_sphere*(r_sphere_small**2*math.pi)
min_square_lenght = math.sqrt(Area)
print(f'The minimum length for the square\'s side must be: {min_square_lenght:.
 ↪3}')
```

```
The minimum length for the square's side must be: 48.4
```

[ ]:
```
# Define the size of the 2D domain
length = (Area*1.5)**0.5
print(length)
'''bounds = Box(x=55,y=55)'''
# Seeds for reproducibility across batches
seeds = [10, 34, 45, 76, 98]

# List to collect batches of spheres
all_batches = []
```

```
# Loop through each seed to create a batch of 1000 spheres
for seed in seeds:
    torch.manual_seed(seed)  # Ensure reproducibility for each batch

    # Generate 500 random 2D centers and assign radius 1.0
    centers_1 = math.random_uniform(instance(pos=0.5*n_sphere),␣
 ↪channel(vector='x,y'),high=length)
    sphere_1 = Sphere(center=centers_1, radius=1.0)

    # Generate 500 random 2D centers and assign radius 0.7
    centers_07 = math.random_uniform(instance(pos=0.5*n_sphere),␣
 ↪channel(vector='x,y'),high=length)
    sphere_07 = Sphere(center=centers_07, radius=0.7)

    # Concatenate both sphere groups into one tensor of 1000 positions
    balls = concat([sphere_1, sphere_07], dim=instance('pos'))  # Shape:␣
 ↪(pos =1000, vector ='x,y')

    # Add a batch dimension to allow stacking
    balls_batched = expand(balls, batch(sample=1))  # Shape: (b =1, pos =1000,␣
 ↪vector ='x,y')

    # Append this batch to the list
    all_batches.append(balls_batched)

# Combine all batches along the batch dimension → (b =5, pos =1000,␣
 ↪vector ='x,y')
sample = concat(all_batches, dim=batch('sample'))

# Print one batch and the full stacked sample
print(balls.radius)   # Last batch created
print(sample)  # All batches concatenated

# Visualize the last created batch using matplotlib, with semi-transparent␣
 ↪red-orange color
show(sample, lib='matplotlib', show_color_bar=False, color='#ff5733', alpha=0.
 ↪6,size=(25,15))
```

```
59.25141171640211
(pos =1000) 0.850 ± 0.150 (7e-01…1e+00)
Sphere(sample =5, pos =1000, vector =x,y)
```



2.2 [2] Energy function

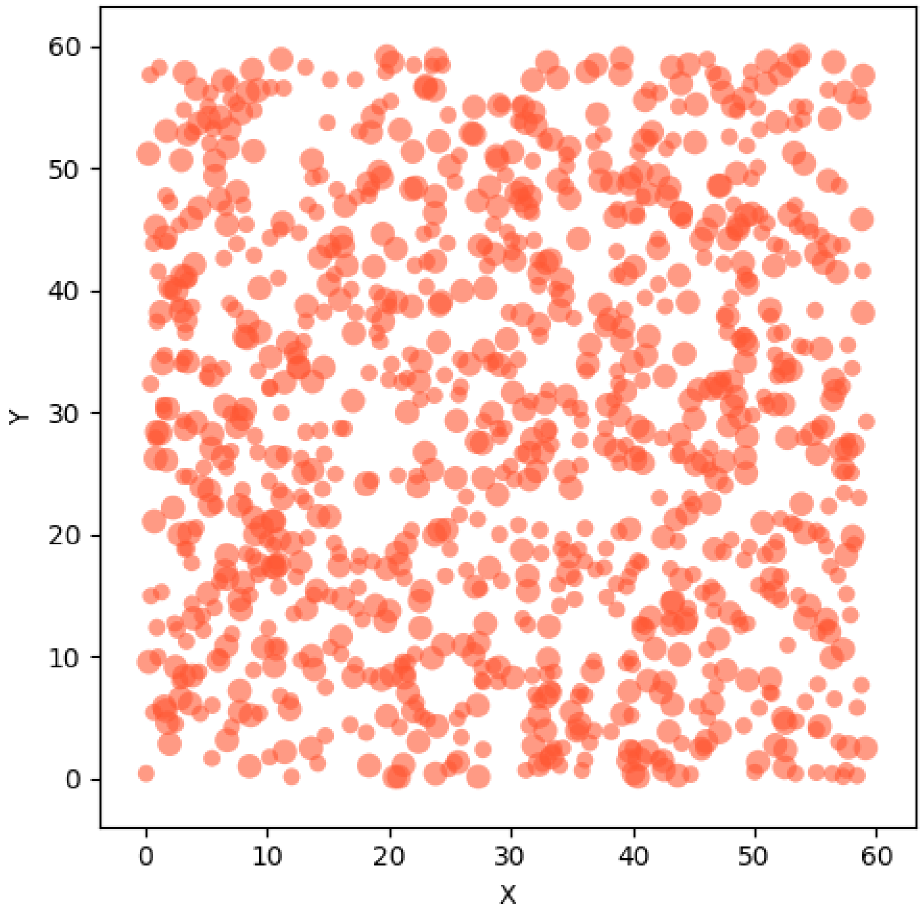
[141]:
```
def loss(x: tensor, boundary=PERIODIC):
  half_sphere=int(n_sphere*0.5)
  r = wrap([1.0]*half_sphere + [0.7]*half_sphere, instance('pos'))
  difx = boundary.shortest_distance(x,math.rename_dims(x,'pos','other'),length)
  # distances between balls
  vec_norm = (math.sum(difx**2, dim='vector'))
  # Radii distance
  dr2 = (r + math.rename_dims(r,'pos','other'))**2
  #dr2.print()
  # difference between distances
  diff = vec_norm-dr2
  #print(diff)
  # set to zero diagonal elements
  # Create identity matrix using PyTorch
  identity_torch = torch.eye(n_sphere) #identity
  identity = tensor(identity_torch, instance(diff))
  diff = diff * (1 - identity)
  #print(diff)
  #diff.print()
  diff = math.where(diff>0.,0.,diff**2)
  #diff_no_diag.print()
  return math.l2_loss(diff)
```

2.3 Gradient Descent Optimization

[ ]:
```
def GradDesc(function: loss, x: tensor, learning_rate: float,desired_loss: float,max_iteration: int):
  iteration = 0
  while function(x).native() >= desired_loss:
    loss_value,dx= math.gradient(loss,wrt='x')(x)
    #print(loss_value)
    #print(dx)
    #print(x)
    x= x-learning_rate*dx
    #print(x)
```

```
    iteration += 1
    if iteration % 2000 == 0:
      print(loss_value)
    if(iteration == max_iteration):
      print(f'max iteration reached: {max_iteration}, break')
      break
  return x
```

[ ]:
```
half_sphere = int(0.5*n_sphere)
r = wrap([1.0]*half_sphere + [0.7]*half_sphere, instance('pos'))
x0 = balls.center
show(Sphere(x0,r),lib='matplotlib', show_color_bar=False, color='#ff5733',␣
 ↪alpha=0.6)
```

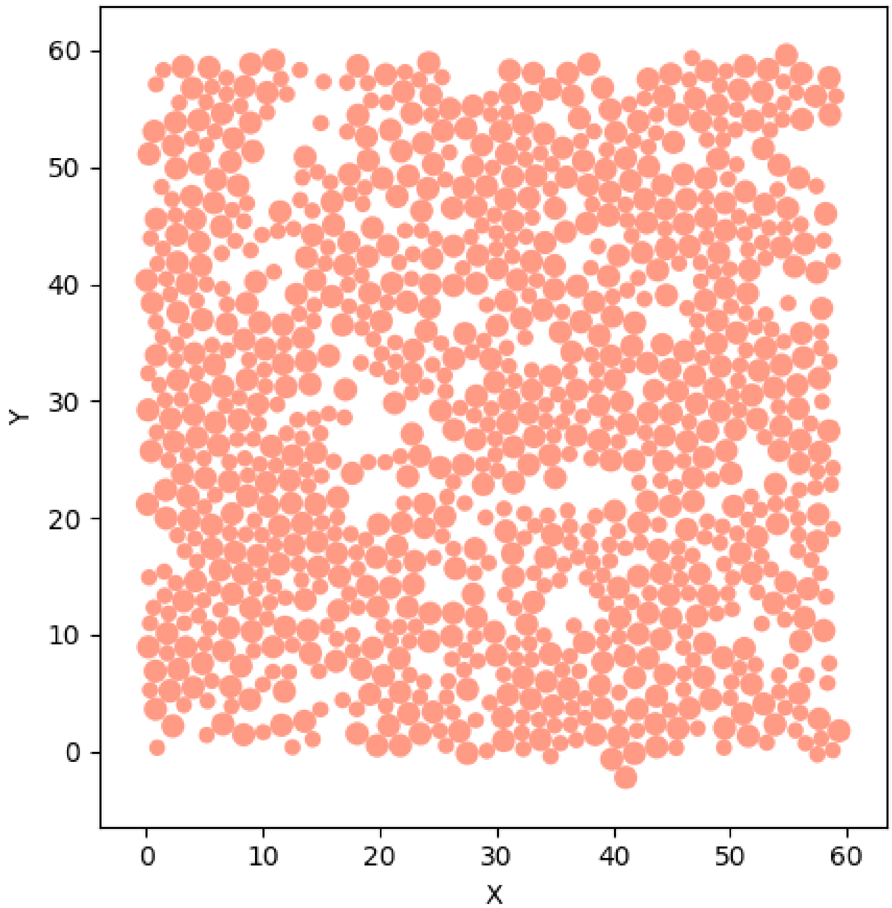


[ ]:
```
x = GradDesc(loss, x0,0.001,0.005,50000)
show(Sphere(x,r), lib='matplotlib', show_color_bar=False, color='#ff5733',␣
 ↪alpha=0.6)
```

```
13.21507
4.7110887
2.4412084
1.5322897
1.0765898
0.8084995
0.6349414
0.51393807
0.42555806
0.359125
0.30769163
0.26725745
0.23463409
0.20802633
0.1858761
0.16725765
0.15146387
0.1379646
0.12631184
0.11615908
0.107323945
0.099509686
0.092584975
0.08639629
0.08083646
0.07583183
0.07133089
0.067325056
0.063691355
0.060378183
0.057346836
0.05460525
0.052086305
0.04976377
0.047607254
0.045616686
0.043791253
0.042100772
0.040557418
0.03914373
0.037837125
0.03661363
0.035470728
0.03440219
0.03341573
0.032498725
0.031649902
0.03085509
```

0.03010904
0.02942053
0.028774472
0.028173225
0.027607284
0.027073786
0.026577454
0.026112381
0.025671406
0.025255036
0.02486293
0.024491047
0.02413587
0.023796808
0.023468655
0.02315524
0.022853905
0.022560205
0.02227874
0.022008104
0.021752471
0.021521494
0.021303138
0.021091811
0.020887174
0.020688979
0.020497879
0.02031399
0.020135542
0.019969039
0.01980928
0.019655107
0.019509831
0.019370202
0.0192351
0.019103529
0.01897845
0.018861124
0.018746046
0.018633874
0.018525286
0.018423114
0.01832328
0.018227445
0.018139921
0.018058676
0.017979871
0.017903334

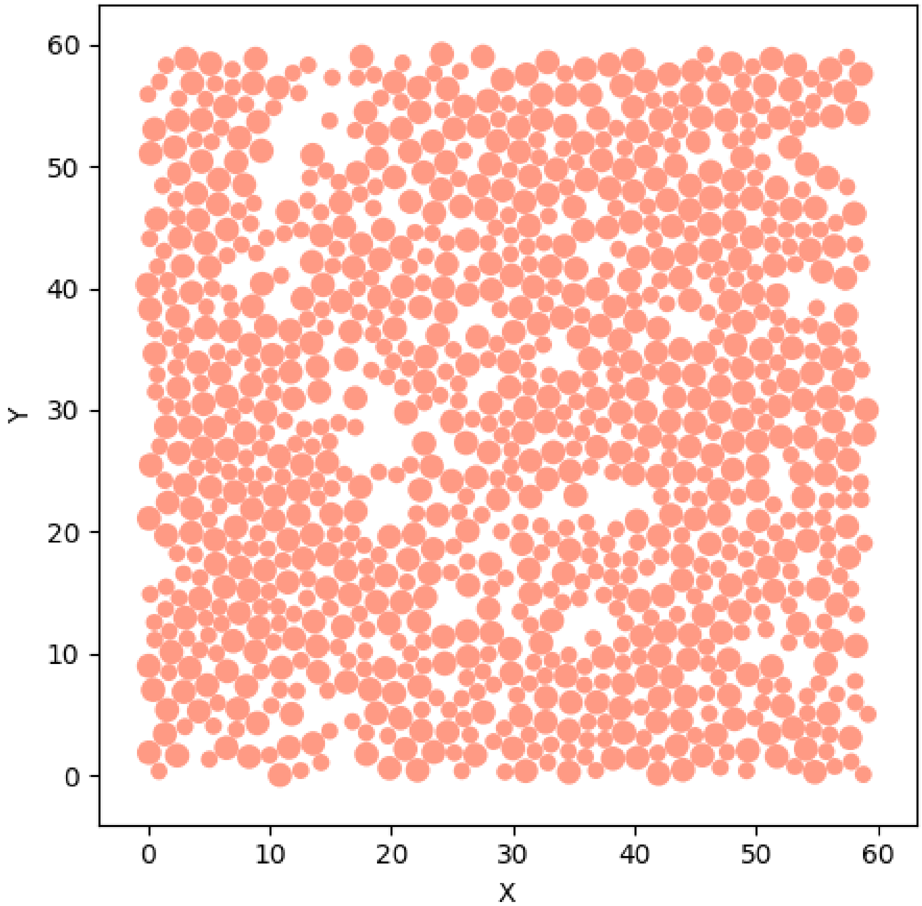
```
0.017828949
0.017755922
0.017684825
0.01761711
max iteration reached: 50000, break
```



2.4 Higher-order Optimization

[ ]:
```
solve = math.Solve(method='L-BFGS-B', x0=x0,rel_tol=0, abs_tol=1e-5,␣
  ↪max_iterations=60)
x =math. minimize(loss, solve) % length
print(loss(x))
show(Sphere(x,r), lib='matplotlib', show_color_bar=False, color='#ff5733',␣
  ↪alpha=0.6)
```

```
7.1621444e-06
```



2.5 Smallest domain

```
[ ]: # Initial scaling factor
scaling_factor = 1.20
min_scaling = 1  # Stop when the scaling factor gets too small
tolerance = 0.01
shrink_step = 0.01  # Reduce scaling factor by 0.05 each loop
max_iters = 50

# Loop to minimize for progressively smaller scaling_factor
for i in range(max_iters):
    length = (Area * scaling_factor) ** 0.5
    solve = math.Solve(method='L-BFGS-B', x0=x0,rel_tol=0, abs_tol=1e-5,␣
 ↪max_iterations=2000)
    result = math.minimize(loss, solve) % length

    print(f"Scale: {scaling_factor:.2f}, Length: {length:.4f}, "
          f"Loss: {loss(result):.5f}")
```

```
    loss_value = loss(result)
    if loss_value.native() > tolerance:
        print("  Stopping: could not reach desired loss.")
        break

    scaling_factor -= shrink_step

    if scaling_factor < min_scaling:
        print("  Reached minimum scaling limit.")
        break
```

```
Scale: 1.20, Length: 52.9961, Loss: 0.000676811
Scale: 1.19, Length: 52.7748, Loss: 0.0054065837
Scale: 1.18, Length: 52.5526, Loss: 0.008407186
Scale: 1.17, Length: 52.3294, Loss: 0.0075260205
Scale: 1.16, Length: 52.1053, Loss: 0.055541698
  Stopping: could not reach desired loss.
```

2.6 Extensive Sphere Packing (optional)

[132]:
```
radii = np.linspace(0.1,1,10)
scaling_factor = 1.75
data = np.zeros((10, 5))  # shape: (radius, Area)
for j,rad in zip(range(10),radii):
  r_values = [1]*500 + [rad]*500  # (1000,)
  r = wrap(r_values, instance(pos=1000))  # shape: (pos =1000,)
  r = expand(r, batch(batch=5))
  Area = 0.5*n_sphere*(r_values[0]**2*math.pi)+0.
↪5*n_sphere*(r_values[500]**2*math.pi)

  for b in range(5):
      # Initial scaling factor
    min_scaling = 1  # Stop when the scaling factor gets too small
    tolerance = 0.01
    shrink_step = 0.005  # Reduce scaling factor by 0.05 each loop
    max_iters = 50

    # Loop to minimize for progressively smaller scaling_factor
    for i in range(max_iters):
        length = (Area * scaling_factor) ** 0.5
        x0 = math.random_uniform(instance(r), channel(vector='x,y'),␣
↪high=length)
        solve = math.Solve(method='L-BFGS-B', x0=x0.batch[b],rel_tol=0.,␣
↪abs_tol=1e-5, max_iterations=200)
        result = math.minimize(loss, solve) % length

        print(f"Scale: {scaling_factor:.4f}, Length: {length:.4f}, "
```

```
                f"Loss: {loss(result):.5f}")
        loss_value = loss(result)
        if loss_value.native() > tolerance:
            data[j,b] = length**2
            scaling_factor += 2*shrink_step
            print("  Stopping: could not reach desired loss.")
            break

        scaling_factor -= shrink_step
```

```
Scale: 1.7500, Length: 52.6914, Loss: 0.0027227253
Scale: 1.7450, Length: 52.6161, Loss: 0.004431047
Scale: 1.7400, Length: 52.5406, Loss: 0.004269215
Scale: 1.7350, Length: 52.4651, Loss: 0.009229529
Scale: 1.7300, Length: 52.3894, Loss: 0.009970738
Scale: 1.7250, Length: 52.3137, Loss: 0.02451891
  Stopping: could not reach desired loss.
Scale: 1.7350, Length: 52.4651, Loss: 0.008463465
Scale: 1.7300, Length: 52.3894, Loss: 0.010339652
  Stopping: could not reach desired loss.
Scale: 1.7400, Length: 52.5406, Loss: 0.0057659894
Scale: 1.7350, Length: 52.4651, Loss: 0.013517897
  Stopping: could not reach desired loss.
Scale: 1.7450, Length: 52.6161, Loss: 0.0052669896
Scale: 1.7400, Length: 52.5406, Loss: 0.008014853
Scale: 1.7350, Length: 52.4651, Loss: 0.005241532
Scale: 1.7300, Length: 52.3894, Loss: 0.010773647
  Stopping: could not reach desired loss.
Scale: 1.7400, Length: 52.5406, Loss: 0.004263869
Scale: 1.7350, Length: 52.4651, Loss: 0.010101009
  Stopping: could not reach desired loss.
Scale: 1.7450, Length: 53.3918, Loss: 0.0002472814
Scale: 1.7400, Length: 53.3152, Loss: 0.0002978974
Scale: 1.7350, Length: 53.2386, Loss: 0.00030390464
Scale: 1.7300, Length: 53.1618, Loss: 0.00037298023
Scale: 1.7250, Length: 53.0849, Loss: 0.0004658981
Scale: 1.7200, Length: 53.0079, Loss: 0.0003976599
Scale: 1.7150, Length: 52.9308, Loss: 0.0007170472
Scale: 1.7100, Length: 52.8536, Loss: 0.00085936906
Scale: 1.7050, Length: 52.7763, Loss: 0.0017000668
Scale: 1.7000, Length: 52.6988, Loss: 0.0015212661
Scale: 1.6950, Length: 52.6213, Loss: 0.0023325938
Scale: 1.6900, Length: 52.5436, Loss: 0.0050830496
Scale: 1.6850, Length: 52.4658, Loss: 0.00979675
Scale: 1.6800, Length: 52.3879, Loss: 0.010735048
  Stopping: could not reach desired loss.
Scale: 1.6900, Length: 52.5436, Loss: 0.004172388
```

```
Scale: 1.6850, Length: 52.4658, Loss: 0.010648791
  Stopping: could not reach desired loss.
Scale: 1.6950, Length: 52.6213, Loss: 0.0034288084
Scale: 1.6900, Length: 52.5436, Loss: 0.0030970452
Scale: 1.6850, Length: 52.4658, Loss: 0.008085414
Scale: 1.6800, Length: 52.3879, Loss: 0.013052359
  Stopping: could not reach desired loss.
Scale: 1.6900, Length: 52.5436, Loss: 0.004409286
Scale: 1.6850, Length: 52.4658, Loss: 0.006569191
Scale: 1.6800, Length: 52.3879, Loss: 0.013385027
  Stopping: could not reach desired loss.
Scale: 1.6900, Length: 52.5436, Loss: 0.008020394
Scale: 1.6850, Length: 52.4658, Loss: 0.00544089
Scale: 1.6800, Length: 52.3879, Loss: 0.01653538
  Stopping: could not reach desired loss.
Scale: 1.6900, Length: 53.7919, Loss: 0.0001789867
Scale: 1.6850, Length: 53.7122, Loss: 0.00016472807
Scale: 1.6800, Length: 53.6325, Loss: 0.00012221618
Scale: 1.6750, Length: 53.5526, Loss: 0.00022350324
Scale: 1.6700, Length: 53.4726, Loss: 0.00015988469
Scale: 1.6650, Length: 53.3925, Loss: 0.000418917
Scale: 1.6600, Length: 53.3123, Loss: 0.00020954112
Scale: 1.6550, Length: 53.2319, Loss: 0.0007944118
Scale: 1.6500, Length: 53.1515, Loss: 0.00020306904
Scale: 1.6450, Length: 53.0709, Loss: 0.00060530525
Scale: 1.6400, Length: 52.9901, Loss: 0.0033692205
Scale: 1.6350, Length: 52.9093, Loss: 0.0015786462
Scale: 1.6300, Length: 52.8283, Loss: 0.0058244453
Scale: 1.6250, Length: 52.7473, Loss: 0.0018671395
Scale: 1.6200, Length: 52.6660, Loss: 0.0018570753
Scale: 1.6150, Length: 52.5847, Loss: 0.011855923
  Stopping: could not reach desired loss.
Scale: 1.6250, Length: 52.7473, Loss: 0.0015699635
Scale: 1.6200, Length: 52.6660, Loss: 0.0022357388
Scale: 1.6150, Length: 52.5847, Loss: 0.012715914
  Stopping: could not reach desired loss.
Scale: 1.6250, Length: 52.7473, Loss: 0.001996387
Scale: 1.6200, Length: 52.6660, Loss: 0.00203285
Scale: 1.6150, Length: 52.5847, Loss: 0.009145933
Scale: 1.6100, Length: 52.5032, Loss: 0.007390192
Scale: 1.6050, Length: 52.4217, Loss: 0.010694127
  Stopping: could not reach desired loss.
Scale: 1.6150, Length: 52.5847, Loss: 0.004313571
Scale: 1.6100, Length: 52.5032, Loss: 0.0051139337
Scale: 1.6050, Length: 52.4217, Loss: 0.007414905
Scale: 1.6000, Length: 52.3399, Loss: 0.010595448
  Stopping: could not reach desired loss.
Scale: 1.6100, Length: 52.5032, Loss: 0.011501972
```

```
  Stopping: could not reach desired loss.
Scale: 1.6200, Length: 54.3308, Loss: 7.73257e-05
Scale: 1.6150, Length: 54.2469, Loss: 6.871651e-05
Scale: 1.6100, Length: 54.1629, Loss: 6.184909e-05
Scale: 1.6050, Length: 54.0787, Loss: 0.0001018905
Scale: 1.6000, Length: 53.9944, Loss: 8.91178e-05
Scale: 1.5950, Length: 53.9100, Loss: 7.98106e-05
Scale: 1.5900, Length: 53.8254, Loss: 0.00010776819
Scale: 1.5850, Length: 53.7407, Loss: 7.969285e-05
Scale: 1.5800, Length: 53.6559, Loss: 0.00011198812
Scale: 1.5750, Length: 53.5709, Loss: 0.00019120205
Scale: 1.5700, Length: 53.4858, Loss: 0.0002754162
Scale: 1.5650, Length: 53.4006, Loss: 0.00016692869
Scale: 1.5600, Length: 53.3152, Loss: 0.00033162322
Scale: 1.5550, Length: 53.2297, Loss: 0.0006680111
Scale: 1.5500, Length: 53.1441, Loss: 0.0003465405
Scale: 1.5450, Length: 53.0583, Loss: 0.00072880957
Scale: 1.5400, Length: 52.9724, Loss: 0.0005406613
Scale: 1.5350, Length: 52.8863, Loss: 0.0006326314
Scale: 1.5300, Length: 52.8001, Loss: 0.0022328454
Scale: 1.5250, Length: 52.7137, Loss: 0.002751682
Scale: 1.5200, Length: 52.6273, Loss: 0.003876542
Scale: 1.5150, Length: 52.5406, Loss: 0.006861472
Scale: 1.5100, Length: 52.4539, Loss: 0.005217535
Scale: 1.5050, Length: 52.3669, Loss: 0.020931792
  Stopping: could not reach desired loss.
Scale: 1.5150, Length: 52.5406, Loss: 0.0051941513
Scale: 1.5100, Length: 52.4539, Loss: 0.00984861
Scale: 1.5050, Length: 52.3669, Loss: 0.010161586
  Stopping: could not reach desired loss.
Scale: 1.5150, Length: 52.5406, Loss: 0.0048030824
Scale: 1.5100, Length: 52.4539, Loss: 0.010042914
  Stopping: could not reach desired loss.
Scale: 1.5200, Length: 52.6273, Loss: 0.0048391293
Scale: 1.5150, Length: 52.5406, Loss: 0.0044691246
Scale: 1.5100, Length: 52.4539, Loss: 0.005739112
Scale: 1.5050, Length: 52.3669, Loss: 0.012800946
  Stopping: could not reach desired loss.
Scale: 1.5150, Length: 52.5406, Loss: 0.0031455022
Scale: 1.5100, Length: 52.4539, Loss: 0.007880188
Scale: 1.5050, Length: 52.3669, Loss: 0.013629046
  Stopping: could not reach desired loss.
Scale: 1.5150, Length: 54.5408, Loss: 5.3961525e-05
Scale: 1.5100, Length: 54.4507, Loss: 4.5532546e-05
Scale: 1.5050, Length: 54.3605, Loss: 5.0869523e-05
Scale: 1.5000, Length: 54.2701, Loss: 0.00023266944
Scale: 1.4950, Length: 54.1796, Loss: 7.040358e-05
Scale: 1.4900, Length: 54.0889, Loss: 7.829652e-05
```

```
Scale: 1.4850, Length: 53.9981, Loss: 8.990837e-05
Scale: 1.4800, Length: 53.9071, Loss: 9.2892726e-05
Scale: 1.4750, Length: 53.8159, Loss: 0.00010271502
Scale: 1.4700, Length: 53.7247, Loss: 0.00016310331
Scale: 1.4650, Length: 53.6332, Loss: 7.0818045e-05
Scale: 1.4600, Length: 53.5416, Loss: 0.0001103436
Scale: 1.4550, Length: 53.4498, Loss: 0.00012213211
Scale: 1.4500, Length: 53.3579, Loss: 0.0002310278
Scale: 1.4450, Length: 53.2659, Loss: 0.00027851027
Scale: 1.4400, Length: 53.1736, Loss: 0.00026458254
Scale: 1.4350, Length: 53.0812, Loss: 0.00038746517
Scale: 1.4300, Length: 52.9887, Loss: 0.00070468744
Scale: 1.4250, Length: 52.8959, Loss: 0.0006691568
Scale: 1.4200, Length: 52.8031, Loss: 0.0013504289
Scale: 1.4150, Length: 52.7100, Loss: 0.0016127102
Scale: 1.4100, Length: 52.6168, Loss: 0.002196862
Scale: 1.4050, Length: 52.5234, Loss: 0.005161219
Scale: 1.4000, Length: 52.4299, Loss: 0.016694646
  Stopping: could not reach desired loss.
Scale: 1.4100, Length: 52.6168, Loss: 0.003192573
Scale: 1.4050, Length: 52.5234, Loss: 0.0043843305
Scale: 1.4000, Length: 52.4299, Loss: 0.006659313
Scale: 1.3950, Length: 52.3362, Loss: 0.014741554
  Stopping: could not reach desired loss.
Scale: 1.4050, Length: 52.5234, Loss: 0.006753005
Scale: 1.4000, Length: 52.4299, Loss: 0.0061772717
Scale: 1.3950, Length: 52.3362, Loss: 0.018472163
  Stopping: could not reach desired loss.
Scale: 1.4050, Length: 52.5234, Loss: 0.00453473
Scale: 1.4000, Length: 52.4299, Loss: 0.017831516
  Stopping: could not reach desired loss.
Scale: 1.4100, Length: 52.6168, Loss: 0.0043702107
Scale: 1.4050, Length: 52.5234, Loss: 0.0055429796
Scale: 1.4000, Length: 52.4299, Loss: 0.009534927
Scale: 1.3950, Length: 52.3362, Loss: 0.01654898
  Stopping: could not reach desired loss.
Scale: 1.4050, Length: 54.7857, Loss: 3.414184e-05
Scale: 1.4000, Length: 54.6882, Loss: 4.9502858e-05
Scale: 1.3950, Length: 54.5904, Loss: 5.5988574e-05
Scale: 1.3900, Length: 54.4925, Loss: 5.167305e-05
Scale: 1.3850, Length: 54.3944, Loss: 0.0002528838
Scale: 1.3800, Length: 54.2961, Loss: 7.312891e-05
Scale: 1.3750, Length: 54.1977, Loss: 5.918917e-05
Scale: 1.3700, Length: 54.0991, Loss: 3.314749e-05
Scale: 1.3650, Length: 54.0002, Loss: 8.496771e-05
Scale: 1.3600, Length: 53.9013, Loss: 8.2646955e-05
Scale: 1.3550, Length: 53.8021, Loss: 0.00020307436
Scale: 1.3500, Length: 53.7027, Loss: 0.00017977756
```

```
Scale: 1.3450, Length: 53.6032, Loss: 7.555239e-05
Scale: 1.3400, Length: 53.5035, Loss: 0.00013103712
Scale: 1.3350, Length: 53.4035, Loss: 0.00022399556
Scale: 1.3300, Length: 53.3034, Loss: 0.00031834038
Scale: 1.3250, Length: 53.2031, Loss: 0.00029330753
Scale: 1.3200, Length: 53.1027, Loss: 0.00029242598
Scale: 1.3150, Length: 53.0020, Loss: 0.000582543
Scale: 1.3100, Length: 52.9011, Loss: 0.0008014595
Scale: 1.3050, Length: 52.8001, Loss: 0.0024934276
Scale: 1.3000, Length: 52.6988, Loss: 0.0021652346
Scale: 1.2950, Length: 52.5974, Loss: 0.008923315
Scale: 1.2900, Length: 52.4958, Loss: 0.0058167386
Scale: 1.2850, Length: 52.3939, Loss: 0.009169066
Scale: 1.2800, Length: 52.2919, Loss: 0.022460835
  Stopping: could not reach desired loss.
Scale: 1.2900, Length: 52.4958, Loss: 0.009282654
Scale: 1.2850, Length: 52.3939, Loss: 0.014188256
  Stopping: could not reach desired loss.
Scale: 1.2950, Length: 52.5974, Loss: 0.004838792
Scale: 1.2900, Length: 52.4958, Loss: 0.0058636046
Scale: 1.2850, Length: 52.3939, Loss: 0.009495788
Scale: 1.2800, Length: 52.2919, Loss: 0.021117965
  Stopping: could not reach desired loss.
Scale: 1.2900, Length: 52.4958, Loss: 0.008541174
Scale: 1.2850, Length: 52.3939, Loss: 0.013992373
  Stopping: could not reach desired loss.
Scale: 1.2950, Length: 52.5974, Loss: 0.0023672532
Scale: 1.2900, Length: 52.4958, Loss: 0.0069415616
Scale: 1.2850, Length: 52.3939, Loss: 0.0079062125
Scale: 1.2800, Length: 52.2919, Loss: 0.031381622
  Stopping: could not reach desired loss.
Scale: 1.2900, Length: 54.9475, Loss: 2.8978338e-05
Scale: 1.2850, Length: 54.8409, Loss: 3.7752616e-05
Scale: 1.2800, Length: 54.7341, Loss: 4.5935798e-05
Scale: 1.2750, Length: 54.6271, Loss: 2.9732171e-05
Scale: 1.2700, Length: 54.5199, Loss: 4.9396356e-05
Scale: 1.2650, Length: 54.4125, Loss: 7.789236e-05
Scale: 1.2600, Length: 54.3048, Loss: 6.0375554e-05
Scale: 1.2550, Length: 54.1970, Loss: 5.3093303e-05
Scale: 1.2500, Length: 54.0889, Loss: 9.045002e-05
Scale: 1.2450, Length: 53.9806, Loss: 5.4147007e-05
Scale: 1.2400, Length: 53.8721, Loss: 8.485515e-05
Scale: 1.2350, Length: 53.7634, Loss: 0.00015724132
Scale: 1.2300, Length: 53.6544, Loss: 0.0005260292
Scale: 1.2250, Length: 53.5453, Loss: 0.00012884793
Scale: 1.2200, Length: 53.4359, Loss: 0.00039176302
Scale: 1.2150, Length: 53.3263, Loss: 0.00026566896
Scale: 1.2100, Length: 53.2164, Loss: 0.022083558
```

```
  Stopping: could not reach desired loss.
Scale: 1.2200, Length: 53.4359, Loss: 0.00013672498
Scale: 1.2150, Length: 53.3263, Loss: 0.0002144369
Scale: 1.2100, Length: 53.2164, Loss: 0.0006752005
Scale: 1.2050, Length: 53.1064, Loss: 0.0008249576
Scale: 1.2000, Length: 52.9961, Loss: 0.0006541304
Scale: 1.1950, Length: 52.8856, Loss: 0.0012778498
Scale: 1.1900, Length: 52.7748, Loss: 0.0041895295
Scale: 1.1850, Length: 52.6638, Loss: 0.0025304286
Scale: 1.1800, Length: 52.5526, Loss: 0.003450785
Scale: 1.1750, Length: 52.4411, Loss: 0.008011074
Scale: 1.1700, Length: 52.3294, Loss: 0.014251502
  Stopping: could not reach desired loss.
Scale: 1.1800, Length: 52.5526, Loss: 0.003378096
Scale: 1.1750, Length: 52.4411, Loss: 0.0111683635
  Stopping: could not reach desired loss.
Scale: 1.1850, Length: 52.6638, Loss: 0.0018959236
Scale: 1.1800, Length: 52.5526, Loss: 0.003619959
Scale: 1.1750, Length: 52.4411, Loss: 0.009545923
Scale: 1.1700, Length: 52.3294, Loss: 0.02489631
  Stopping: could not reach desired loss.
Scale: 1.1800, Length: 52.5526, Loss: 0.006673266
Scale: 1.1750, Length: 52.4411, Loss: 0.006586942
Scale: 1.1700, Length: 52.3294, Loss: 0.015423933
  Stopping: could not reach desired loss.
Scale: 1.1800, Length: 55.1344, Loss: 2.2443955e-05
Scale: 1.1750, Length: 55.0175, Loss: 2.668862e-05
Scale: 1.1700, Length: 54.9003, Loss: 2.2896758e-05
Scale: 1.1650, Length: 54.7829, Loss: 4.1482457e-05
Scale: 1.1600, Length: 54.6652, Loss: 5.2064308e-05
Scale: 1.1550, Length: 54.5472, Loss: 5.5846864e-05
Scale: 1.1500, Length: 54.4291, Loss: 0.00012426917
Scale: 1.1450, Length: 54.3106, Loss: 8.79768e-05
Scale: 1.1400, Length: 54.1919, Loss: 5.6764406e-05
Scale: 1.1350, Length: 54.0729, Loss: 0.00011459261
Scale: 1.1300, Length: 53.9537, Loss: 0.0001088426
Scale: 1.1250, Length: 53.8342, Loss: 0.00019109523
Scale: 1.1200, Length: 53.7144, Loss: 0.00014478131
Scale: 1.1150, Length: 53.5944, Loss: 0.00013185499
Scale: 1.1100, Length: 53.4741, Loss: 0.0002566584
Scale: 1.1050, Length: 53.3535, Loss: 0.00020346949
Scale: 1.1000, Length: 53.2327, Loss: 0.0003652425
Scale: 1.0950, Length: 53.1115, Loss: 0.00077926484
Scale: 1.0900, Length: 52.9901, Loss: 0.00094330014
Scale: 1.0850, Length: 52.8685, Loss: 0.00082072575
Scale: 1.0800, Length: 52.7465, Loss: 0.0011770072
Scale: 1.0750, Length: 52.6243, Loss: 0.0032556802
Scale: 1.0700, Length: 52.5017, Loss: 0.0051787216
```

```
Scale: 1.0650, Length: 52.3789, Loss: 0.009168962
Scale: 1.0600, Length: 52.2558, Loss: 0.03178361
  Stopping: could not reach desired loss.
Scale: 1.0700, Length: 52.5017, Loss: 0.0036320286
Scale: 1.0650, Length: 52.3789, Loss: 0.013989715
  Stopping: could not reach desired loss.
Scale: 1.0750, Length: 52.6243, Loss: 0.0025640966
Scale: 1.0700, Length: 52.5017, Loss: 0.010576287
  Stopping: could not reach desired loss.
Scale: 1.0800, Length: 52.7465, Loss: 0.0056110797
Scale: 1.0750, Length: 52.6243, Loss: 0.013790595
  Stopping: could not reach desired loss.
Scale: 1.0850, Length: 52.8685, Loss: 0.0007296136
Scale: 1.0800, Length: 52.7465, Loss: 0.0012489832
Scale: 1.0750, Length: 52.6243, Loss: 0.0031038763
Scale: 1.0700, Length: 52.5017, Loss: 0.005348717
Scale: 1.0650, Length: 52.3789, Loss: 0.012075235
  Stopping: could not reach desired loss.
Scale: 1.0750, Length: 55.2845, Loss: 4.2874475e-05
Scale: 1.0700, Length: 55.1558, Loss: 4.934983e-05
Scale: 1.0650, Length: 55.0268, Loss: 4.104059e-05
Scale: 1.0600, Length: 54.8974, Loss: 6.387046e-05
Scale: 1.0550, Length: 54.7678, Loss: 3.0678475e-05
Scale: 1.0500, Length: 54.6379, Loss: 4.532417e-05
Scale: 1.0450, Length: 54.5076, Loss: 6.711303e-05
Scale: 1.0400, Length: 54.3771, Loss: 6.03876e-05
Scale: 1.0350, Length: 54.2462, Loss: 7.409656e-05
Scale: 1.0300, Length: 54.1150, Loss: 6.0175134e-05
Scale: 1.0250, Length: 53.9835, Loss: 9.9893856e-05
Scale: 1.0200, Length: 53.8517, Loss: 0.0001403003
Scale: 1.0150, Length: 53.7195, Loss: 0.00010658317
Scale: 1.0100, Length: 53.5871, Loss: 0.00018616037
Scale: 1.0050, Length: 53.4543, Loss: 0.0003135944
Scale: 1.0000, Length: 53.3211, Loss: 0.00048518376
Scale: 0.9950, Length: 53.1876, Loss: 0.00029507244
Scale: 0.9900, Length: 53.0538, Loss: 0.000484645
Scale: 0.9850, Length: 52.9197, Loss: 0.0012433943
Scale: 0.9800, Length: 52.7852, Loss: 0.0013242699
Scale: 0.9750, Length: 52.6504, Loss: 0.002940551
Scale: 0.9700, Length: 52.5152, Loss: 0.003846882
Scale: 0.9650, Length: 52.3797, Loss: 0.010403545
  Stopping: could not reach desired loss.
Scale: 0.9750, Length: 52.6504, Loss: 0.0029966042
Scale: 0.9700, Length: 52.5152, Loss: 0.006529188
Scale: 0.9650, Length: 52.3797, Loss: 0.0114279315
  Stopping: could not reach desired loss.
Scale: 0.9750, Length: 52.6504, Loss: 0.0025332838
Scale: 0.9700, Length: 52.5152, Loss: 0.0035239717
```

```
Scale: 0.9650, Length: 52.3797, Loss: 0.011148836
  Stopping: could not reach desired loss.
Scale: 0.9750, Length: 52.6504, Loss: 0.0027369603
Scale: 0.9700, Length: 52.5152, Loss: 0.0046790405
Scale: 0.9650, Length: 52.3797, Loss: 0.007276036
Scale: 0.9600, Length: 52.2438, Loss: 0.031635188
  Stopping: could not reach desired loss.
Scale: 0.9700, Length: 52.5152, Loss: 0.0033453272
Scale: 0.9650, Length: 52.3797, Loss: 0.011799998
  Stopping: could not reach desired loss.
Scale: 0.9750, Length: 55.3449, Loss: 4.34484e-05
Scale: 0.9700, Length: 55.2028, Loss: 3.339965e-05
Scale: 0.9650, Length: 55.0603, Loss: 2.897315e-05
Scale: 0.9600, Length: 54.9175, Loss: 4.19163e-05
Scale: 0.9550, Length: 54.7743, Loss: 4.3339973e-05
Scale: 0.9500, Length: 54.6307, Loss: 4.4772045e-05
Scale: 0.9450, Length: 54.4867, Loss: 7.7420875e-05
Scale: 0.9400, Length: 54.3424, Loss: 8.881101e-05
Scale: 0.9350, Length: 54.1977, Loss: 6.2265e-05
Scale: 0.9300, Length: 54.0526, Loss: 8.951164e-05
Scale: 0.9250, Length: 53.9071, Loss: 7.9181504e-05
Scale: 0.9200, Length: 53.7612, Loss: 0.00012006566
Scale: 0.9150, Length: 53.6149, Loss: 0.000121693345
Scale: 0.9100, Length: 53.4682, Loss: 0.00014363868
Scale: 0.9050, Length: 53.3211, Loss: 0.00022413963
Scale: 0.9000, Length: 53.1736, Loss: 0.0006402304
Scale: 0.8950, Length: 53.0257, Loss: 0.00064598606
Scale: 0.8900, Length: 52.8774, Loss: 0.0007608905
Scale: 0.8850, Length: 52.7286, Loss: 0.0034598166
Scale: 0.8800, Length: 52.5795, Loss: 0.004368591
Scale: 0.8750, Length: 52.4299, Loss: 0.004625618
Scale: 0.8700, Length: 52.2799, Loss: 0.017409764
  Stopping: could not reach desired loss.
Scale: 0.8800, Length: 52.5795, Loss: 0.005906227
Scale: 0.8750, Length: 52.4299, Loss: 0.008382049
Scale: 0.8700, Length: 52.2799, Loss: 0.022467662
  Stopping: could not reach desired loss.
Scale: 0.8800, Length: 52.5795, Loss: 0.003071791
Scale: 0.8750, Length: 52.4299, Loss: 0.006912636
Scale: 0.8700, Length: 52.2799, Loss: 0.01693666
  Stopping: could not reach desired loss.
Scale: 0.8800, Length: 52.5795, Loss: 0.00487911
Scale: 0.8750, Length: 52.4299, Loss: 0.00831042
Scale: 0.8700, Length: 52.2799, Loss: 0.024905276
  Stopping: could not reach desired loss.
Scale: 0.8800, Length: 52.5795, Loss: 0.002265291
Scale: 0.8750, Length: 52.4299, Loss: 0.00951398
Scale: 0.8700, Length: 52.2799, Loss: 0.019289149
```

```
Stopping: could not reach desired loss.
```

[ ]:
```
%%capture
!jupyter nbconvert --to pdf --output /content/drive/MyDrive/Fisica/ADL4P/
↪Exercise_3.pdf /content/drive/MyDrive/Fisica/ADL4P/Exercise_3.ipynb
```

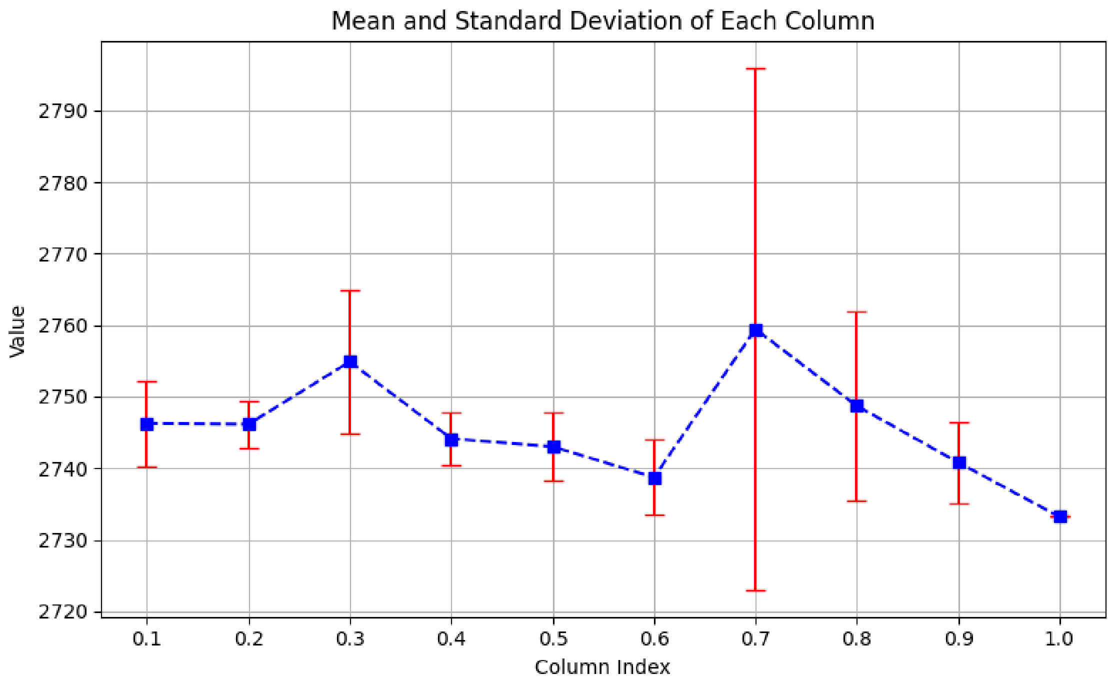
```
(pos =1000, vector =x,y) 24.407 ± 13.759
(1e-01…5e+01)
```

[133]:
```
print(data)
```

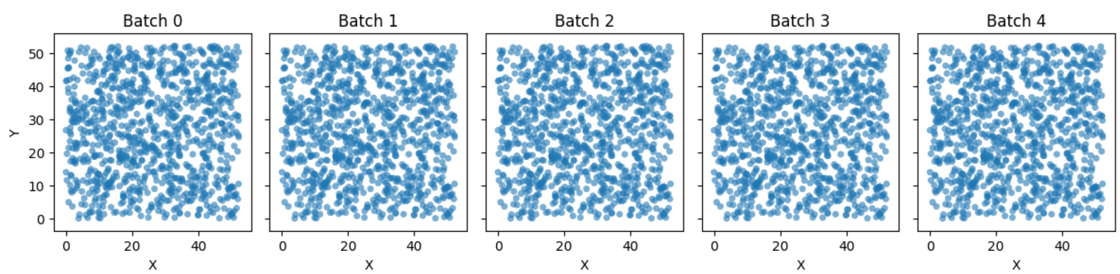
```
[[2736.71990036 2744.65242181 2752.58494326 2744.65242181 2752.58494326]
 [2744.49534218 2752.66348308 2744.49534218 2744.49534218 2744.49534218]
 [2765.15131387 2765.15131387 2748.02963391 2739.46879393 2756.59047389]
 [2742.29622732 2742.29622732 2751.40684601 2742.29622732 2742.29622732]
 [2748.89357189 2739.07609485 2739.07609485 2748.89357189 2739.07609485]
 [2734.44224568 2745.12366071 2734.44224568 2745.12366071 2734.44224568]
 [2831.98869758 2738.3692365  2750.07166914 2738.3692365  2738.3692365 ]
 [2730.6723345  2743.55286438 2756.43339426 2769.31392414 2743.55286438]
 [2743.6314042  2743.6314042  2743.6314042  2729.41569744 2743.6314042 ]
 [2733.18560862 2733.18560862 2733.18560862 2733.18560862 2733.18560862]]
```

[136]:
```
import numpy as np
import matplotlib.pyplot as plt

# Compute mean and standard deviation along rows (i.e., for each column)
means = np.mean(data, axis=1)
std_devs = np.std(data, axis=1)

# Plotting with error bars
x = np.linspace(0.1,1,10)
plt.figure(figsize=(8, 5))
plt.errorbar(x, means, yerr=std_devs, fmt='o', capsize=5, ecolor='red',␣
↪marker='s', linestyle='--', color='blue')

plt.title("Mean and Standard Deviation of Each Column")
plt.xlabel("Column Index")
plt.ylabel("Value")
plt.xticks(x)  # Show x-axis ticks at each column index
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
<ipython-input-136-f8be6efb0ecc>:11: UserWarning: marker is redundantly defined
by the 'marker' keyword argument and the fmt string "o" (-> marker='o'). The
keyword argument will take precedence.
  plt.errorbar(x, means, yerr=std_devs, fmt='o', capsize=5, ecolor='red',
marker='s', linestyle='--', color='blue')
```



[139]:
```
show(Sphere(x0,r),lib='matplotlib',alpha=0.6)
```



Comment: the problem is due to he fact that in the loss function i redefine the radius of the sphere so i have alvays computed the system for radius of the sphere 1. and 0.7.

[ ]:
```
```