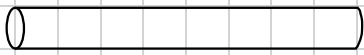


VARMELIKNINGEN I TO DIMENSJONER

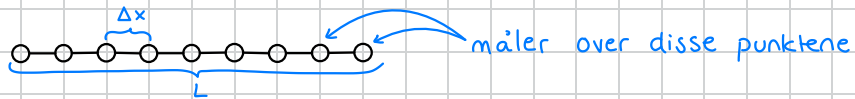
- Av Ingrid Lyngstad og Tuva Tvedt Serigstad

LØSER FØRST I ÉN DIMENSJON

Kontinuerlig metall stang:



Diskret metall stang:



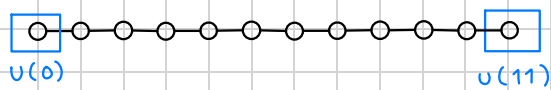
Stangens karakteristikk

- lengde
- diffusivitet, $\alpha = \frac{k}{\rho c}$ $k = \text{termisk konduktivitet}$, $\rho = \text{masse tetthet}$, $c = \text{spesifikk varmekapasitet}$

Diskrete elementer:

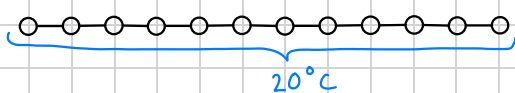
- steg i x-retning $\rightarrow h$ jo mindre h , jo mer presisjon
- steg i tid $\rightarrow k$

Setter randkrav:



Temperatur ved randen er alltid kjent

Setter initialkrav:



Stangen starter ved 20°C ($t = 0$ s)

Itererer gjennom hver node for å beregne the differential scheme for hver node.

$$T_{t+1,i} = T_{t,i} + \alpha \frac{\partial t}{h^2} (T_{t,i+1} - 2T_{t,i} + T_{t,i-1})$$

Løser med python:

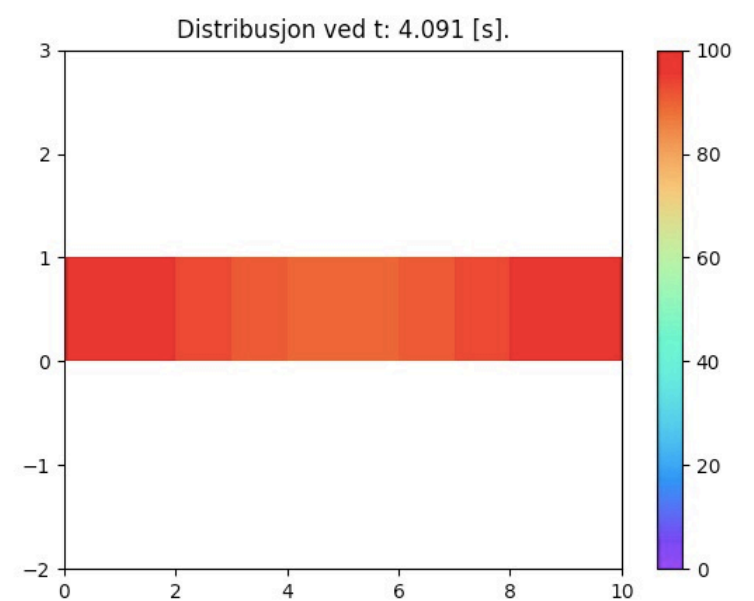
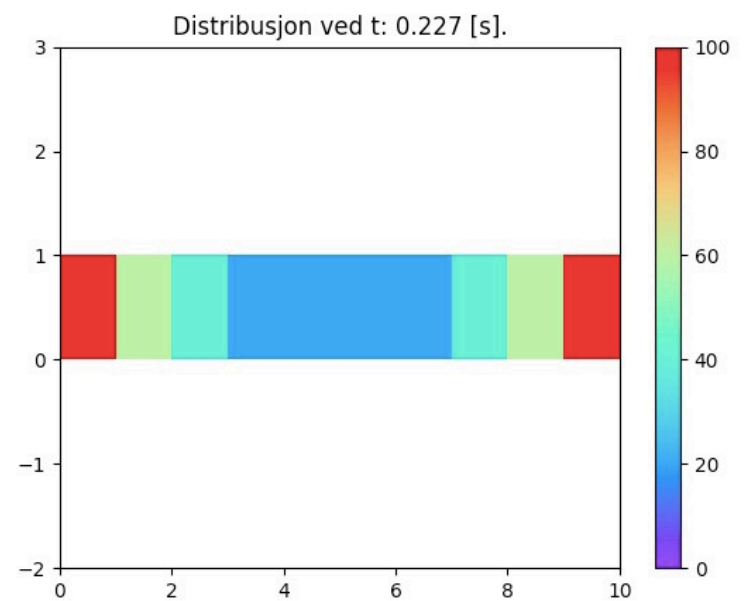
```
varmelikning1D.py > ...
1  import numpy as np
2  import matplotlib.pyplot as plt
3
4  #Definerer problemet som skal løses
5
6  a = 110
7  lengde = 50 #mm
8  tid = 4 #sekunder
9  noder = 10
10
11 #Initiering
12
13 h = lengde/noder
14 k = 0.5 * h**2 / a
15 t_noder = int(tid/k)
16
17 u = np.zeros(noder) + 20 #Stanger starter på 20 grader
18
19 #Randkrav
20
21 u[0] = 100
22 u[-1] = 100
23
24 #Lager simulering
25
26 counter = 0
27
28 while counter < tid :
29     w = u.copy() #Lager kopi for å ikke endre u
30
31     for i in range(1, noder - 1):
32         u[i] = k * a * (w[i-1] - 2 * w[i] + w[i+1])/h**2 + w[i]
33
34     counter += k
35     print("t: {:.3f} [s], Gjennomsnittlig temperatur: {:.2f} grader Celsius".format(counter, np.average(u)))
36
```

Terminal :

```
t: 0.568 [s], Gjennomsnittlig temperatur: 58.00 grader Celsius
t: 0.682 [s], Gjennomsnittlig temperatur: 60.50 grader Celsius
t: 0.795 [s], Gjennomsnittlig temperatur: 63.00 grader Celsius
t: 0.909 [s], Gjennomsnittlig temperatur: 65.19 grader Celsius
t: 1.023 [s], Gjennomsnittlig temperatur: 67.34 grader Celsius
t: 1.136 [s], Gjennomsnittlig temperatur: 69.28 grader Celsius
t: 1.250 [s], Gjennomsnittlig temperatur: 71.16 grader Celsius
t: 1.364 [s], Gjennomsnittlig temperatur: 72.88 grader Celsius
t: 1.477 [s], Gjennomsnittlig temperatur: 74.54 grader Celsius
t: 1.591 [s], Gjennomsnittlig temperatur: 76.06 grader Celsius
t: 1.705 [s], Gjennomsnittlig temperatur: 77.51 grader Celsius
t: 1.818 [s], Gjennomsnittlig temperatur: 78.86 grader Celsius
t: 1.932 [s], Gjennomsnittlig temperatur: 80.14 grader Celsius
t: 2.045 [s], Gjennomsnittlig temperatur: 81.33 grader Celsius
t: 2.159 [s], Gjennomsnittlig temperatur: 82.46 grader Celsius
t: 2.273 [s], Gjennomsnittlig temperatur: 83.52 grader Celsius
t: 2.386 [s], Gjennomsnittlig temperatur: 84.51 grader Celsius
t: 2.500 [s], Gjennomsnittlig temperatur: 85.45 grader Celsius
t: 2.614 [s], Gjennomsnittlig temperatur: 86.33 grader Celsius
t: 2.727 [s], Gjennomsnittlig temperatur: 87.15 grader Celsius
t: 2.841 [s], Gjennomsnittlig temperatur: 87.93 grader Celsius
t: 2.955 [s], Gjennomsnittlig temperatur: 88.65 grader Celsius
t: 3.068 [s], Gjennomsnittlig temperatur: 89.34 grader Celsius
t: 3.182 [s], Gjennomsnittlig temperatur: 89.98 grader Celsius
t: 3.295 [s], Gjennomsnittlig temperatur: 90.58 grader Celsius
t: 3.409 [s], Gjennomsnittlig temperatur: 91.15 grader Celsius
t: 3.523 [s], Gjennomsnittlig temperatur: 91.69 grader Celsius
t: 3.636 [s], Gjennomsnittlig temperatur: 92.19 grader Celsius
t: 3.750 [s], Gjennomsnittlig temperatur: 92.66 grader Celsius
t: 3.864 [s], Gjennomsnittlig temperatur: 93.10 grader Celsius
t: 3.977 [s], Gjennomsnittlig temperatur: 93.52 grader Celsius
t: 4.091 [s], Gjennomsnittlig temperatur: 93.91 grader Celsius
```

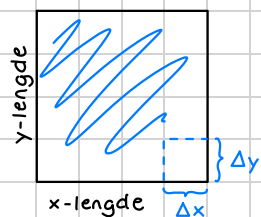
Legger til kode for å visualisere :

```
varmellikning1D.py > ...
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 #Definerer problemet som skal løses
5
6 a = 110
7 lengde = 50 #mm
8 tid = 4 #sekunder
9 noder = 10
10
11 #Initiering
12
13 h = lengde/noder
14 k = 0.5 * h**2 / a
15 t_noder = int(tid/k)
16
17 u = np.zeros(noder) + 20 #Stanger starter på 20 grader
18
19 #Randkrav
20
21 u[0] = 100
22 u[-1] = 100
23
24 #Visualisering
25
26 fig, axis = plt.subplots()
27 pcm = axis.pcolormesh([u], cmap = plt.cm.rainbow, vmin=0, vmax=100)
28 plt.colorbar(pcm, ax=axis)
29 axis.set_ylim([-2, 3])
30 #Lager simulering
31
32 counter = 0
33
34 while counter < tid :
35     w = u.copy() #Lager kopi for å ikke endre u
36
37     for i in range(1, noder - 1):
38         u[i] = k * a * (w[i-1] - 2 * w[i] + w[i+1])/h**2 + w[i]
39
40     counter += k
41     print("t: {:.3f} [s], Gjennomsnittlig temperatur: {:.2f} grader Celsius".format(counter, np.average(u)))
42
43     #Oppdaterer plot
44
45     pcm.set_array([u])
46     axis.set_title("Distribusjon ved t: {:.3f} [s]".format(counter))
47     plt.pause(0.01)
48
49 plt.show()
```

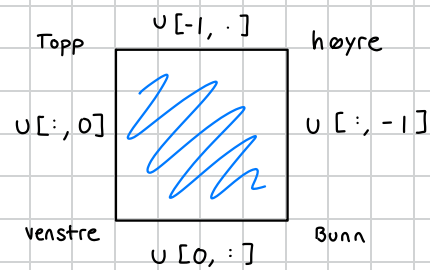


Varmen sprer seg ved tiden

GJØR OM TIL 2D



Randkrav :



Diskrete elementer :

- Steg i x-retning $\rightarrow h$
- Steg i y-retning $\rightarrow y$
- Steg i tid $\rightarrow k$

For å hindre at systemet vil divergere så reduserer vi tidssteget og øker antall noder

Modifisert kode

```
varmelikning2D.py > ...
1  import numpy as np
2  import matplotlib.pyplot as plt
3
4  #Definerer problemet som skal løses
5
6  a = 110
7  lengde = 50 #mm
8  tid = 4 #sekunder
9  noder = 30
10
11  #Initiering
12
13  h = lengde/noder
14  y = lengde/noder
15  k = min(h**2/(4*a), y**2/(4*a))
16  t_noder = int(tid/k)
17
18  u = np.zeros((noder, noder)) + 20 #Stangen starter på 20 grader
19
20  #Randkrav
21
22  u[0, :] = 100
23  u[-1, :] = 100
24
25  #Visualisering
26
27  fig, axis = plt.subplots()
28  pcm = axis.pcolormesh(u, cmap = plt.cm.jet, vmin=0, vmax=100)
29  plt.colorbar(pcm, ax=axis)
30
31  #Lager simulering
32
33  counter = 0
34
35  while counter < tid :
36      w = u.copy() #Lager kopi for å ikke endre u
37
38      for i in range(1, noder - 1):
39          for j in range(1, noder - 1):
40
41              dd_ux = (w[i+1, j] - 2*w[i, j] + w[i-1, j])/h**2 #Andrederiverte av x
42              dd_uy = (w[i, j+1] - 2*w[i, j] + w[i, j-1])/y**2 #Andrederiverte av y
43              u[i, j] = k * a * (dd_ux + dd_uy) + w[i, j]
44
45      counter += k
46      print("t: {:.3f} [s], Gjennomsnittlig temperatur: {:.2f} grader Celsius".format(counter, np.average(u)))
47
48      #Oppdaterer plot
49
50      pcm.set_array(u)
51      axis.set_title("Distribusjon ved t: {:.3f} [s]".format(counter))
52      plt.pause(0.01)
53
54  plt.show()
```


Terminal:

```
t: 3.870 [s], Gjennomsnittlig temperatur: 59.51 grader Celsius
t: 3.876 [s], Gjennomsnittlig temperatur: 59.52 grader Celsius
t: 3.883 [s], Gjennomsnittlig temperatur: 59.52 grader Celsius
t: 3.889 [s], Gjennomsnittlig temperatur: 59.52 grader Celsius
t: 3.895 [s], Gjennomsnittlig temperatur: 59.53 grader Celsius
t: 3.902 [s], Gjennomsnittlig temperatur: 59.53 grader Celsius
t: 3.908 [s], Gjennomsnittlig temperatur: 59.54 grader Celsius
t: 3.914 [s], Gjennomsnittlig temperatur: 59.54 grader Celsius
t: 3.920 [s], Gjennomsnittlig temperatur: 59.54 grader Celsius
t: 3.927 [s], Gjennomsnittlig temperatur: 59.55 grader Celsius
t: 3.933 [s], Gjennomsnittlig temperatur: 59.55 grader Celsius
t: 3.939 [s], Gjennomsnittlig temperatur: 59.55 grader Celsius
t: 3.946 [s], Gjennomsnittlig temperatur: 59.56 grader Celsius
t: 3.952 [s], Gjennomsnittlig temperatur: 59.56 grader Celsius
t: 3.958 [s], Gjennomsnittlig temperatur: 59.56 grader Celsius
t: 3.965 [s], Gjennomsnittlig temperatur: 59.57 grader Celsius
t: 3.971 [s], Gjennomsnittlig temperatur: 59.57 grader Celsius
t: 3.977 [s], Gjennomsnittlig temperatur: 59.58 grader Celsius
t: 3.984 [s], Gjennomsnittlig temperatur: 59.58 grader Celsius
t: 3.990 [s], Gjennomsnittlig temperatur: 59.58 grader Celsius
t: 3.996 [s], Gjennomsnittlig temperatur: 59.59 grader Celsius
t: 4.003 [s], Gjennomsnittlig temperatur: 59.59 grader Celsius
```

Visualisering:

