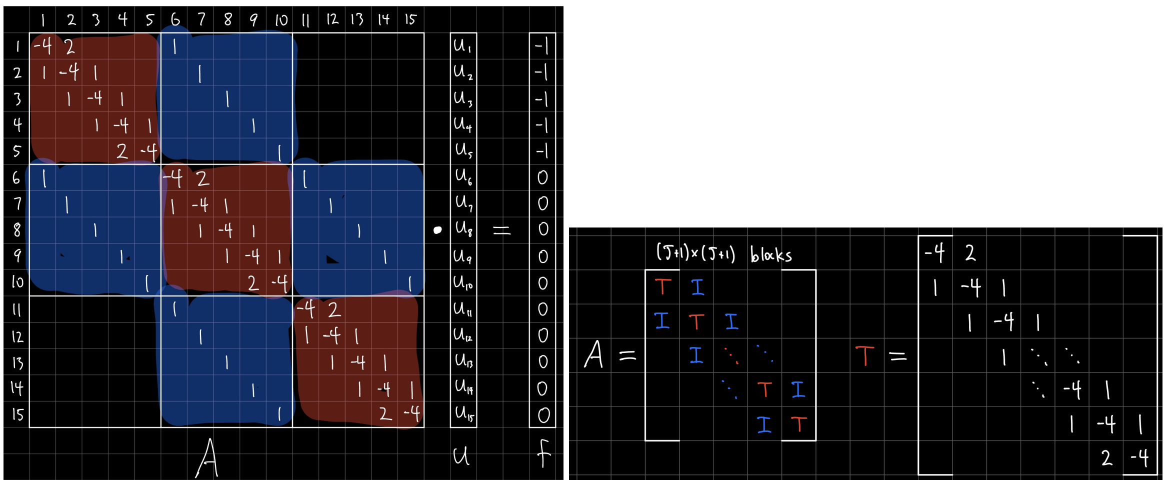
Scientific Computing HW 6

Ryan Chen

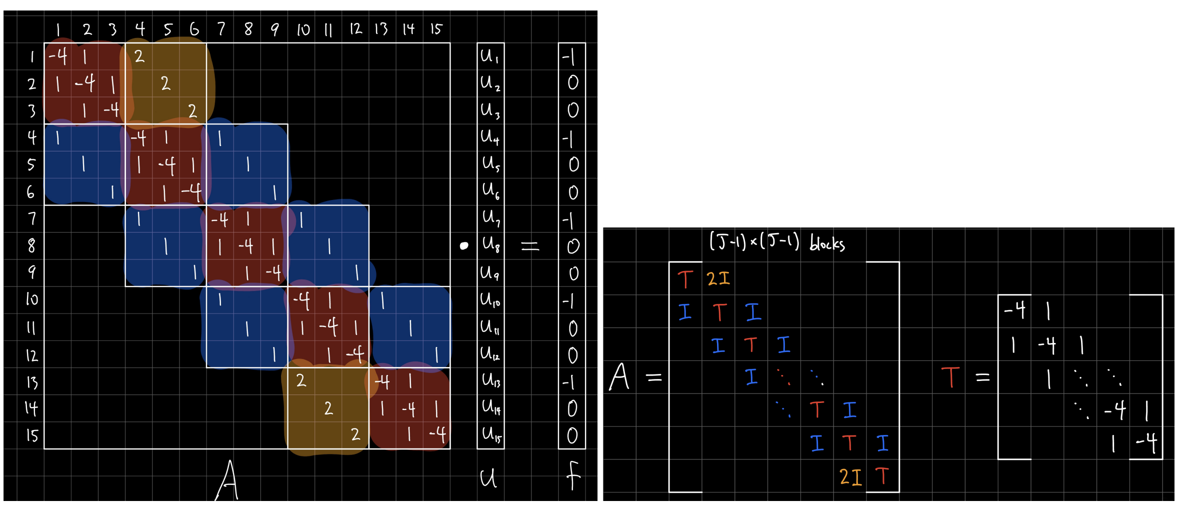
March 14, 2024

1. In the following cases, the system $Au = f$ is shown on the left, and the block structure of A (for a general $J \times J$ mesh) is compactly rewritten on the right.

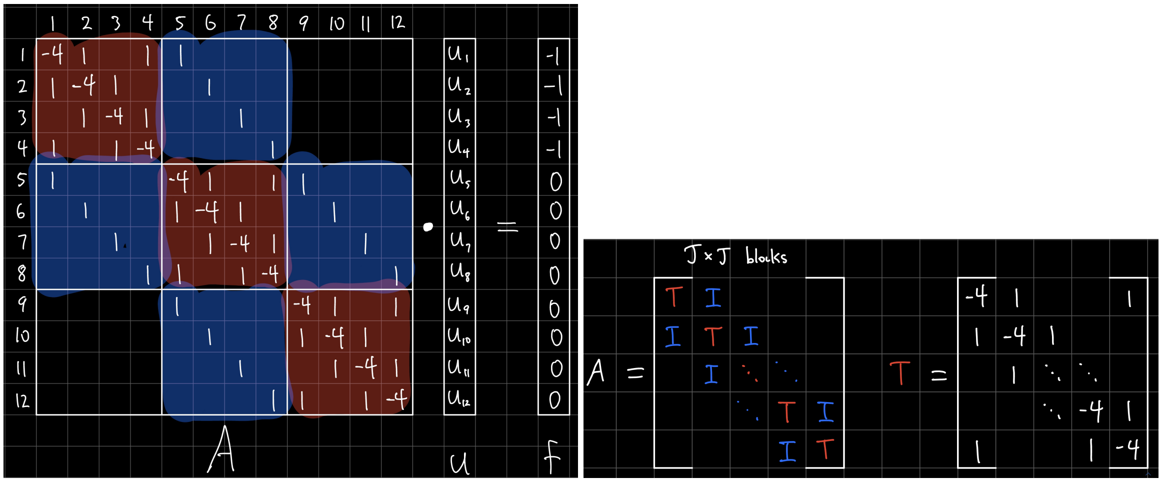
(a)



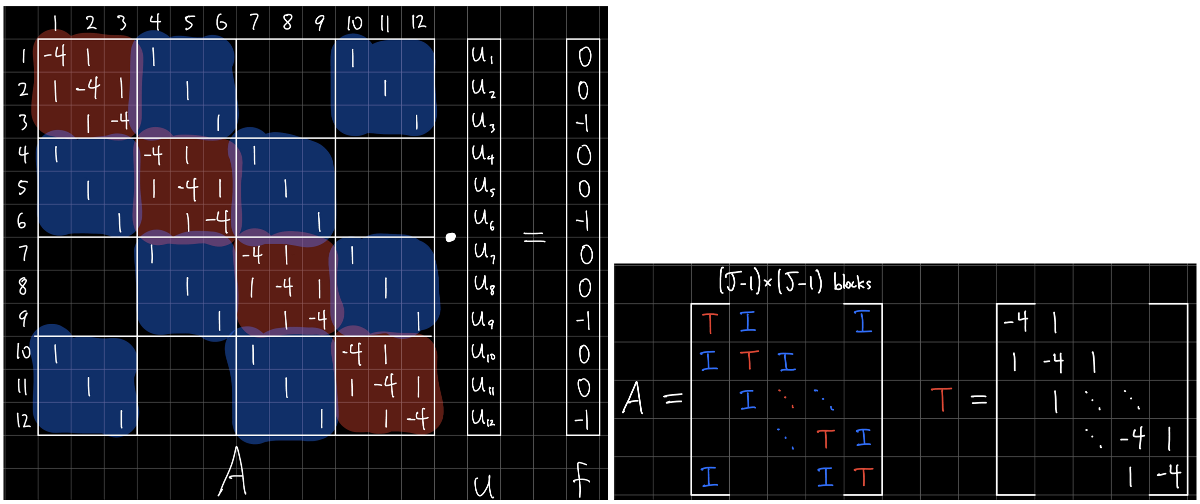
(b)



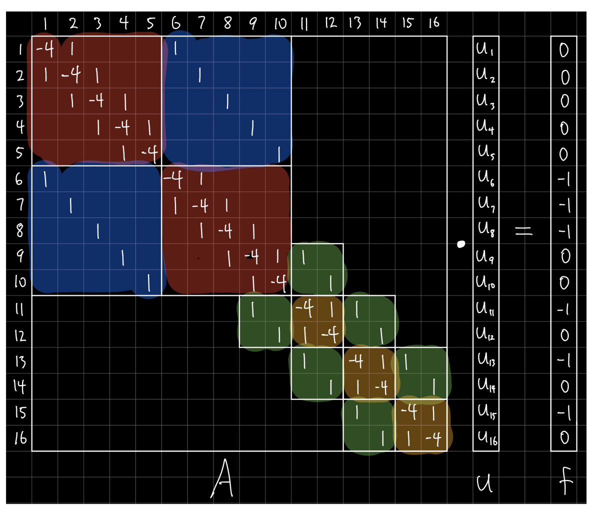
(c)



(d)



(e) In this case we do not compactly rewrite the block structure of A, but it is still clear.



2. (a) The BVP on the domain $\Omega := [-\pi, \pi] \times [0, 2]$ is

$$u_{xx} + u_{yy} = g(x) := \begin{cases} -\cos x, & -\frac{\pi}{2} \le x \le \frac{\pi}{2} \\ 0, & \text{else} \end{cases} \qquad (2.1)$$

with BCs

$$u\Big|_{x=\pi} = u\Big|_{x=-\pi}, \quad u_x\Big|_{x=\pi} = u_x\Big|_{x=-\pi}, \quad u\Big|_{y=0} = 0, \quad u_y\Big|_{y=2} = 0$$

(b) Fix $J \in \mathbb{N}$ and set mesh steps in the x and y axes,

$$h_x := \frac{2\pi}{J}, \quad h_y := \frac{2}{J}$$

Then

$$u_{xx}(x, y) = \frac{1}{h_x^2}\left[u(x + h_x, y) - 2u(x, y) + u(x - h_x, y)\right] + O(h_x^2)$$

$$u_{yy}(x, y) = \frac{1}{h_y^2}\left[u(x, y + h_y) - 2u(x, y) + u(x, y - h_y)\right] + O(h_y^2)$$

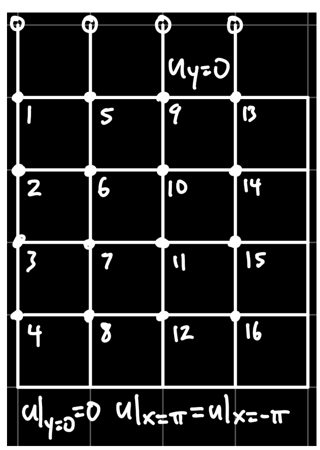
Plug these expressions into (2.1) and use the compass direction notation from lecture.

$$-2u_P\left[\frac{1}{h_x^2} + \frac{1}{h_y^2}\right] + \frac{1}{h_x^2}[u_E + u_W] + \frac{1}{h_y^2}[u_N + u_S] = g_P$$

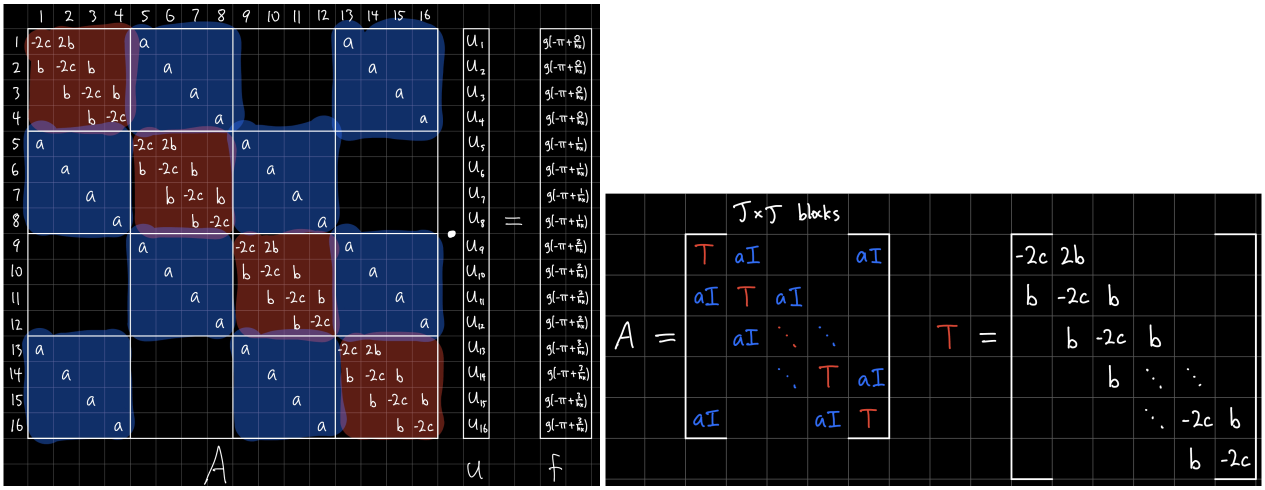
Set $a := \frac{1}{h_x^2}$, $b := \frac{1}{h_y^2}$, $c := a + b$, so that

$$-2cu_P + a[u_E + u_W] + b[u_N + u_S] = g_P \qquad (2.2)$$

To explore the appropriate numerical method, take $J = 4$. The mesh steps create a mesh from Ω with $(J + 1)^2 = 25$ points. Using the BCs, it is enough to solve for the values of 16 points, labeled below.



Using the fact $u\big|_{y=0} = 0$, the values of the points on the bottom edge are all 0. Using the fact $u_y\big|_{y=2} = 0$, the values of the "ghost points" above the mesh from left to right are, respectively, u_2, u_6, u_{10}, u_{14}. Using the fact $u\big|_{x=-\pi} = u\big|_{x=\pi}$, the following pairs of points are treated as adjacent: 1 and 13, 2 and 14, 3 and 15, 4 and 16. Apply (2.2) to each point in the mesh to obtain a system $Au = f$, shown below on the left, with the block structure of A compactly rewritten on the right. The values of a, b, c are given in the last part.



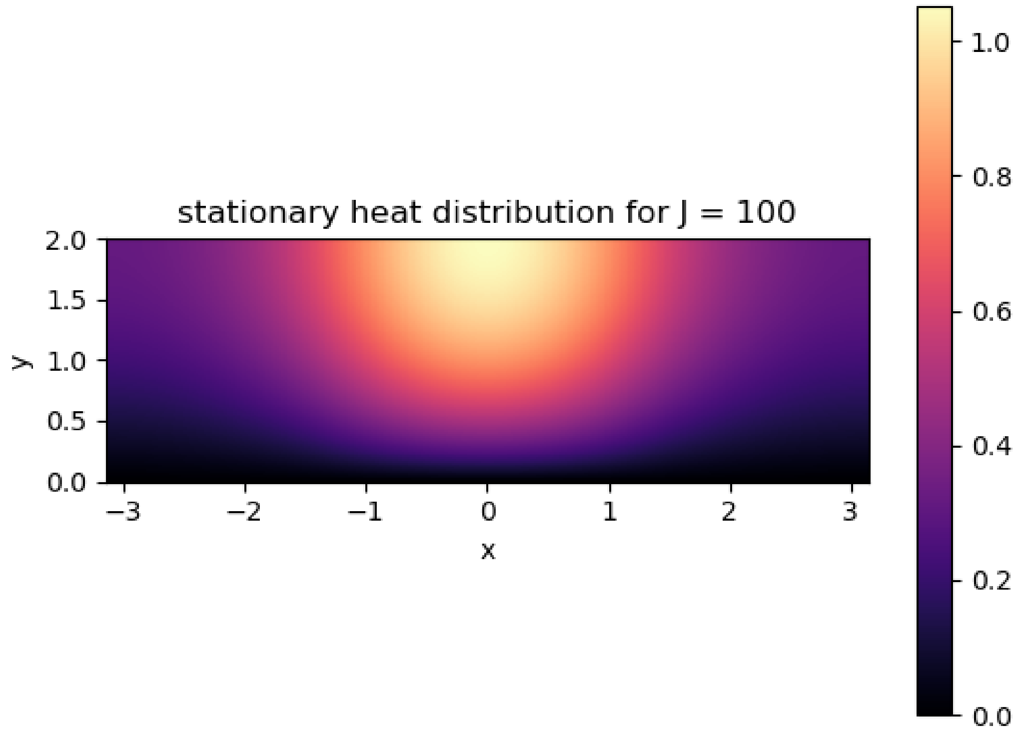
(c) The stationary heat distribution is solved for $J = 100$.
Code: `https://github.com/RokettoJanpu/scientific-computing-2-redux/blob/main/hw6.ipynb`

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.sparse import spdiags
```

```
# right hand side of Laplace's equation
def g(x):
    return -np.cos(x) if -np.pi/2 <= x <= np.pi/2 else 0
```

```
# set mesh size
J = 100
hx = 2*np.pi/J
hy = 2/J
a = 1/hx**2
b = 1/hy**2
c = a + b

# define block structure of A
v = b*np.ones(J)
v[1] = 2*b
T = spdiags([b*np.ones(J), -2*c*np.ones(J), v], [-1, 0, 1]).toarray()
S = spdiags([np.ones(J), np.ones(J), np.ones(J), np.ones(J)], [-J+1, -1, 1, J-1]).toarray()
A = np.kron(np.identity(J), T) + np.kron(S, a*np.identity(J))

# define f
f = np.zeros(J**2)
for j in range(J**2):
    f[j] = g(-np.pi + int(j/J)*hx)

# solve Au=f for u, convert u to 2d array, fill in BCs
u = np.linalg.solve(A, f)
u = np.reshape(u, (J, J), order = 'F')
u = np.pad(u, (0, 1))
for j in range(J):
    u[j][-1] = u[j][0]

# plot u
plt.title('stationary heat distribution for J = {}'.format(J))
plt.xlabel('x')
plt.ylabel('y')
im = plt.imshow(u, cmap = plt.cm.magma, extent = [-np.pi, np.pi, 0, 2])
plt.colorbar(im)
plt.savefig('hw6 2 plot.png')
plt.show()
```

Stationary heat distribution:


stationary heat distribution for J = 100
2.0
1.5
1.0
0.5
0.0
y
−3
−2
−1
0
1
2
3
x
1.0
0.8
0.6
0.4
0.2
0.0