ENGR 21
Computer Engineering Fundamentals

Instructor: Emad Masroor

Lecture 19
Tuesday, Nov 11, 2025

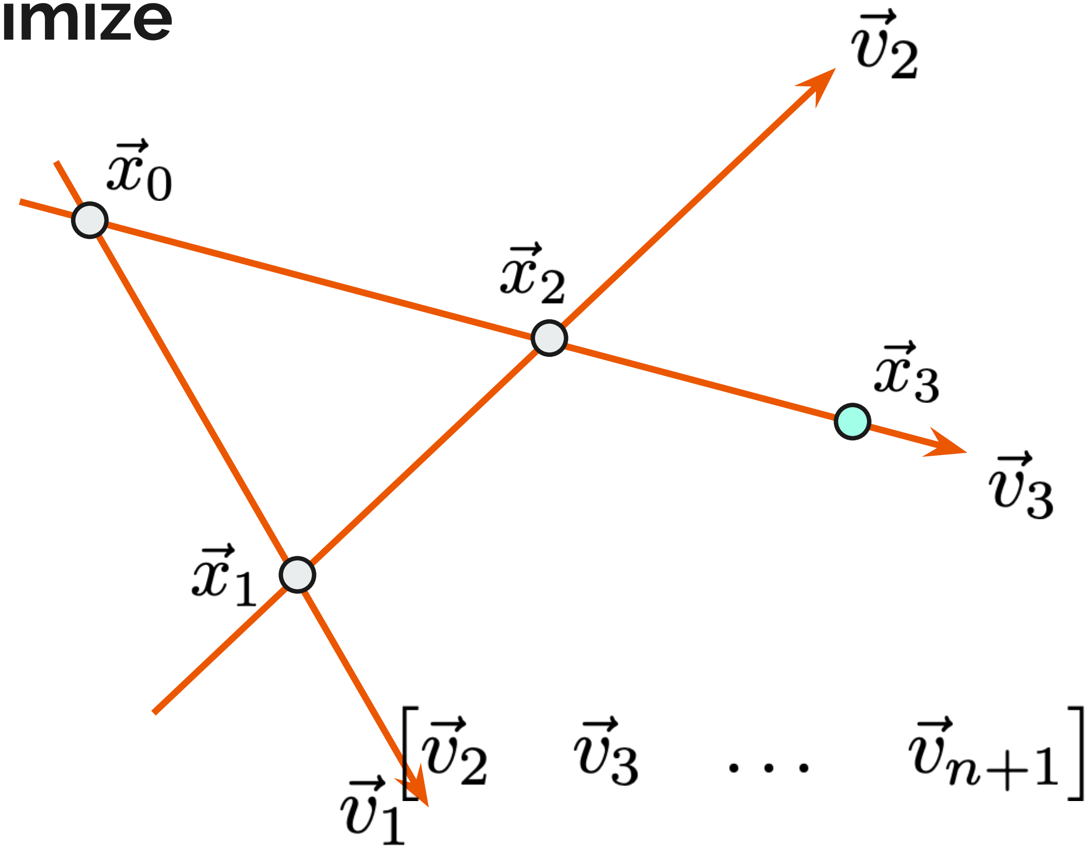
Powell's method gives us a way to find the "best" directions in which to minimize

Starting from $\vec{x}_0$ do naive optimization for **n** steps in any n directions.

After completing **n** steps, define

$$\vec{v}_{n+1} = \vec{x}_0 - \vec{x}_n$$

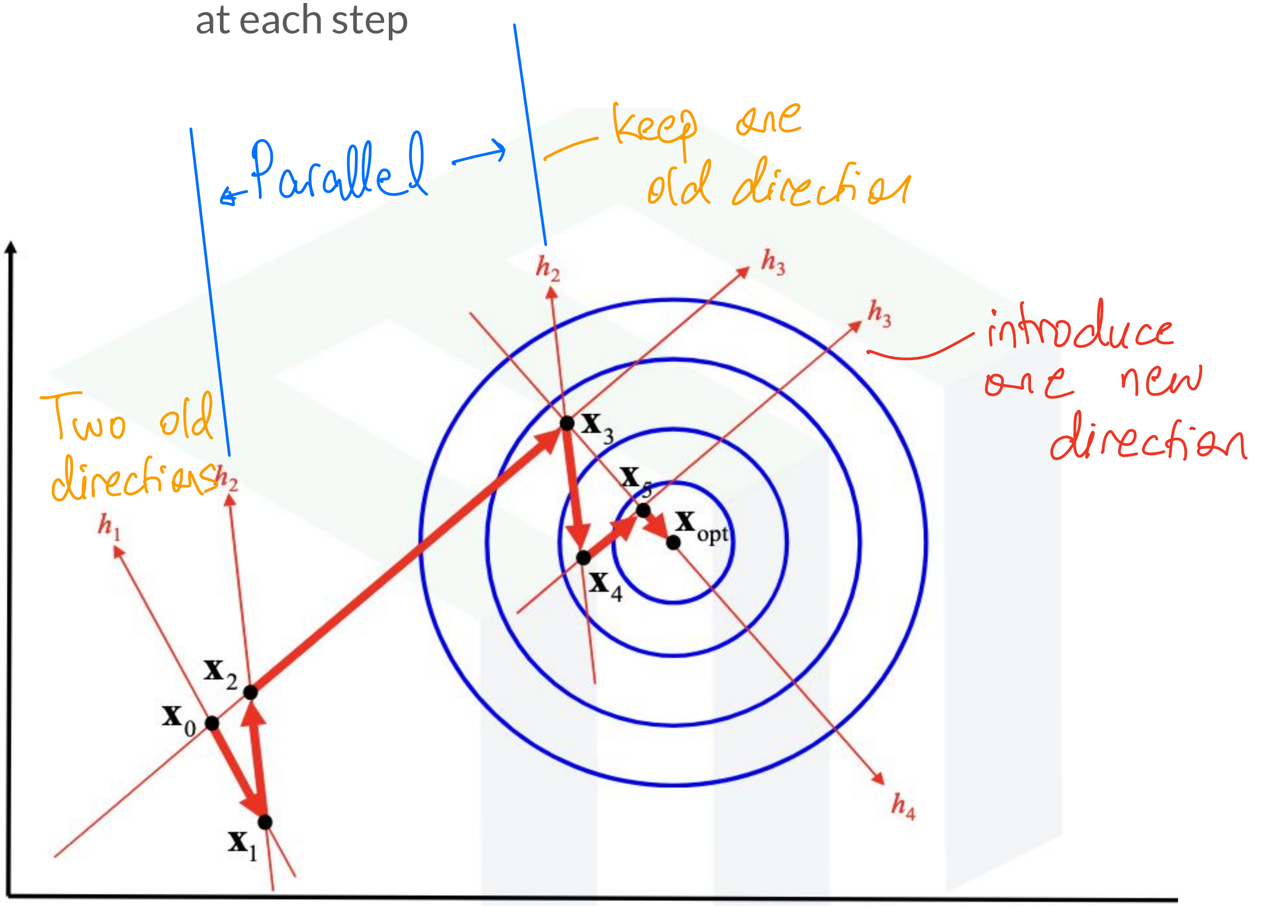
Then, minimize along this new direction.



For the next step, drop $\vec{v}_1$; use $\left[\vec{v}_2, \vec{v}_3, \dots, \vec{v}_{n+1}\right]$ as the new set of directions

Powell's Method

- Provides a set of **n** mutually conjugate directions
  - To find these directions, you need to carry out **n** one-dimensional searches at each step



https://empossible.net

Solving 1-D Optimization problems with Python

Find a minimum of the function

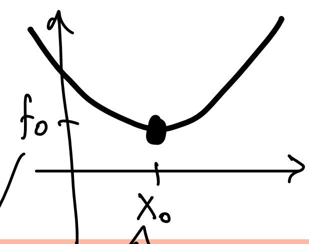
$$f(x) = x^2 - 4x + 7$$

Between [-1, +5].

and a maximum of $f(x) = -x^2 - 4x + 7$

```
pip install scipy
```

get code from the Resources page !




```python
from scipy.optimize import minimize_scalar
import numpy as np
from matplotlib import pyplot as plt

def test_f(x):
    return x**2 - 4*x + 7

solution = minimize_scalar(test_f,[-1,5])

print(f"Function minimized at x = {solution.x:.4f}")

x = np.linspace(-1,5)
plt.plot(x,test_f(x))
plt.grid()
plt.xlabel('y')
plt.ylabel('Shear stress')

plt.scatter(solution.x,test_f(solution.x),marker='o',color='black')
plt.show()
```

What is this? check type

```
>>> solution
 message:
         Optimization terminated
successfully;
         The returned value
satisfies the termination criteria
         (using xtol = 1.48e-08 )
 success: True
     fun: 3.0
       x: 2.0
     nit: 5
    nfev: 8
```

value of x that minimizes f

Solving n-D Optimization problems with Python

Find a minimum of the function

```
pip install scipy
```

$$f(x, y) = (x - 1)^2 + (y + 1)^2 + xy$$

Near the origin.

scipy.optimize.minimize wants f to have one input only (+ optional parameters)

```
from scipy.optimize import minimize
import numpy as np

def f(xx):
    x = xx[0]
    y = xx[1]
    return (x-1)**2 + (y+1)**2 + x*y

solution = minimize(f,np.array([0.,0.]),method='Powell')
# followed by more code to plot the result
```

unpack input variable info n elements

guess

see manual for other methods

get code from the Resources page !

Solving n-D Optimization problems with Python

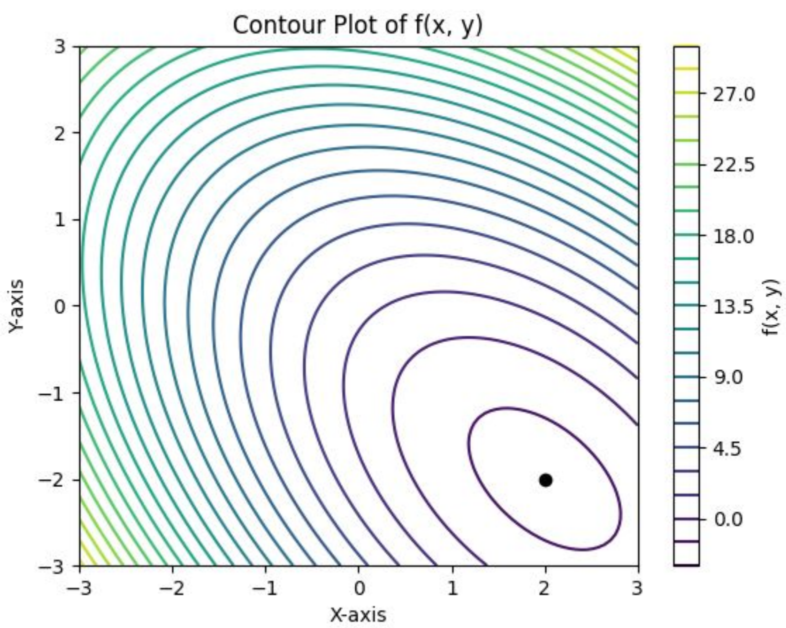
Find a minimum of the function

$$f(x, y) = (x - 1)^2 + (y + 1)^2 + xy$$

Near the origin.

```
pip install scipy
```

```
x = np.linspace(-3, 3, 100)
y = np.linspace(-3, 3, 100)
X, Y = np.meshgrid(x, y)

# Calculate the function values over the grid
Z = f([X,Y])

# Create a contour plot
contours = plt.contour(X, Y, Z, levels=20, cmap='viridis')

# Add labels and a color bar
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.colorbar(contours, label='f(x, y)')

# plot the minimum
plt.scatter(solution.x[0],solution.x[1],marker='o',color='black')

# Show the plot
plt.title('Contour Plot of f(x, y)')
plt.show()
```



get code from the Resources page !

with constraints!

Solving n-D Optimization problems with Python

```
pip install scipy
```

Find a minimum of the function

$$f(x,y) = (x-1)^2 + (y+1)^2 + xy$$

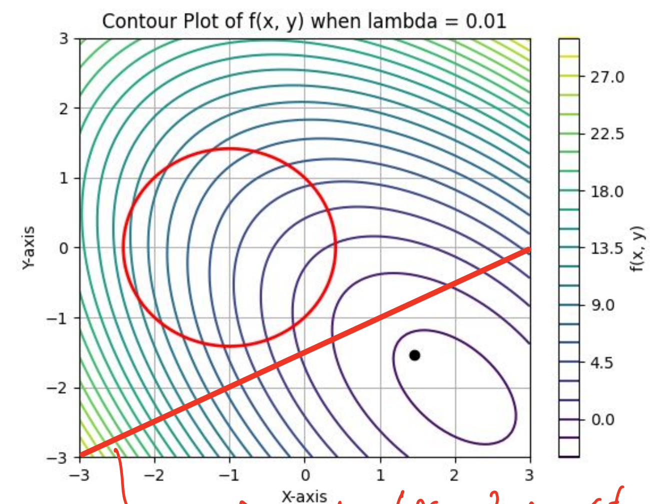
Subject to the constraint that the minimum should fall on the circle

$$(x+1)^2 + y^2 - 2 = 0$$

Circle with radius √2 centered at (-1,0).

```
# GLOBAL VARIABLE
LAMBDA = 0.01

def f(xx):
    x = xx[0]
    y = xx[1]
    return (x-1)**2 + (y+1)**2 + x*y

def constraint(x,y):
    return (x+1)**2 + (y)**2 - 2

def f_star(x_vector):
    x = x_vector[0]
    y = x_vector[1]
    lam = LAMBDA      # parameter to change
    return f(x_vector) + lam*(constraint(x,y))**2

solution = minimize(f_star,np.array([0.,0.]),method='Powell')
```

see what happens when you change



new constraint: {x,y} must be on this line.

How does changing λ affect result?

Find a minimum of the function

$$f(x, y) = (x-1)^2 + (y+1)^2 + xy$$

Subject to the constraint that the minimum should fall on the circle

$$(x+1)^2 + y^2 - 2 = 0$$

Find a minimum of the function

$$f(x, y) = (x-1)^2 + (y+1)^2 + xy$$

Subject to the constraint that the minimum should fall on the circle

$$(x+1)^2 + y^2 - 2 = 0$$

How does changing λ affect result?