Mid-Point Circle Generation Algorithm

April 2, 2025

[1]:
```
import matplotlib.pyplot as plt
```

[6]:
```
def midPointCircleDraw(x_centre, y_centre, r):
    x = r
    y = 0
    points = []
    points.append((x + x_centre, y + y_centre))
    if r > 0:
        points.append((x + x_centre, -y + y_centre))
        points.append((y + x_centre, x + y_centre))
        points.append((-y + x_centre, x + y_centre))
    P = 1 - r
    while x > y:
        y += 1
        if P <= 0:
            P = P + 2 * y + 1
        else:
            x -= 1
            P = P + 2 * y - 2 * x + 1
        if x < y:
            break
        points.append((x + x_centre, y + y_centre))
        points.append((-x + x_centre, y + y_centre))
        points.append((x + x_centre, -y + y_centre))
        points.append((-x + x_centre, -y + y_centre))
        if x != y:
            points.append((y + x_centre, x + y_centre))
            points.append((-y + x_centre, x + y_centre))
            points.append((y + x_centre, -x + y_centre))
            points.append((-y + x_centre, -x + y_centre))

    return points
```

[7]:
```
if __name__ == '__main__':
    points = midPointCircleDraw(0, 0, 3)

    x_vals = [point[0] for point in points]
    y_vals = [point[1] for point in points]
```

```
    plt.figure(figsize=(6,6))
    plt.plot(x_vals, y_vals, 'ro', markersize=4)
    plt.title("Mid-Point Circle Algorithm Visualization")
    plt.gca().set_aspect('equal', adjustable='box')
    plt.xlabel("X-axis")
    plt.ylabel("Y-axis")
    plt.grid(True)
    plt.show()
```