04_01_while_teacher

May 2, 2025

1 Flying with `while` Loops + MultiRanger Obstacle Sensing

This notebook shows how to use `while` loops to repeatedly check flight conditions and make decisions based on MultiRanger sensor data. The `CrazyflieSimulator` now simulates realistic sensor values, just like a real drone.

[1]:
```
#  Setup
import time
from crazyflie_sim import CrazyflieSimulator

drone = CrazyflieSimulator(real=True)
```

```
Error no LogEntry to handle id=1
Error no LogEntry to handle id=1
Error no LogEntry to handle id=1
Error no LogEntry to handle id=1
```

```
 Connecting to real Crazyflie…
```

[ ]:
```

```

```
Z-range height: 12.0 meters
```

```
Got link error callback [Too many packets lost] in state [2]
```

1.1 Takeoff First

[3]:
```
drone.takeoff(1.0, 0.3)
time.sleep(2)
```

```
 Taking off to 1.0m at 0.3m/s!
Executing command: takeoff 1.0 0.3
```

1.2 Example 1: Hover until height is above 0.8m

[4]:
```
while drone.get_height() < 0.8:
    print(" Waiting to reach 0.8m...")
    time.sleep(0.5)
print(" Altitude reached!")
```

```
Altitude reached!
```

1.3 Example 2: Move forward until X position > 1.0

[6]:
```
while drone.get_position()[0] < 1.0:
    drone.forward(0.1, 0.2)
    time.sleep(1)
print(" Stopping forward motion.")
```

```
Stopping forward motion.
```

1.4 Example 3: Rotate until yaw < 90°

[ ]:
```
while drone.get_yaw() > 90:
    drone.rotate(-10, 0.5)
    time.sleep(0.5)
print(" Yaw aligned!")
```

1.5 MultiRanger Example: Move forward until obstacle < 0.5m

[ ]:
```
while drone.get_distances()['front'] > 0.5:
    print("Path clear. Moving forward...")
    drone.forward(0.1, 0.2)
    time.sleep(0.5)
print(" Obstacle too close. Stopping.")
```

1.6 Exercise 1: Fly upward until height is at least 1.5m

[ ]:
```
# while drone.get_height() < :
#     drone.up( , )
#     time.sleep( )
```

1.7 Exercise 2: Hover until yaw < 45

[ ]:
```
# while drone.get_yaw() > :
#     drone.rotate( , )
#     time.sleep( )
```

1.8 Exercise 3: Zig-zag left/right until X > 1.2

[ ]:
```
# while drone.get_position()[0] < :
#     drone.left( , )
#     time.sleep( )
#     drone.right( , )
#     time.sleep( )
```

1.9 Exercise 4: Rotate until yaw is between 80 and 100

[ ]:
```
# while not (80 <= drone.get_yaw() <= 100):
#     drone.rotate( , )
#     time.sleep( )
```

1.10 Exercise 5: Move up until 'up' sensor sees obstacle < 0.4m

[ ]:
```
# while drone.get_distances()['up'] > :
#     drone.up( , )
#     time.sleep( )
```

1.11 Exercise 6: Back up until object behind is closer than 0.3m

[ ]:
```
# while drone.get_distances()['back'] > :
#     drone.backward( , )
#     time.sleep( )
```

1.12 Land and Close

[ ]:
```
drone.land(0.3)
drone.close()
```