1)

State Space (S): The state space includes:

- Position: The drone's current latitude, longitude, and altitude.
- Velocity: Speed and direction of movement.
- Orientation: Pitch, roll, and yaw.
- Battery Level: Current battery percentage.
- Obstacles: Positions, sizes, and velocities of nearby obstacles.
- Time: Time elapsed since the start of the delivery.
- Delivery Status: Whether the drone is en route, has completed the delivery, or is returning to base.

Action Space (A): The action space consists of:

- Movement Controls: Adjustments in altitude, direction (yaw, pitch, roll), and speed.
- Hover: The ability to maintain a stationary position to avoid obstacles or wait for optimal conditions.
- Avoidance Manoeuvres: Actions to navigate around detected obstacles.
- Route Adjustments: Recalibrating the flight path based on real-time data.
- Battery Management: Actions to optimize battery usage, such as changing speed or taking more direct routes.

Reward Function (R): The reward function is designed to balance efficiency and safety:

- Positive Rewards: For successful deliveries, minimal battery usage, and optimal path efficiency.
- Negative Rewards: For collisions with obstacles, excessive battery consumption, or significant delays.
- Intermediate Penalties: For close proximity to obstacles and minor deviations from the optimal path.

This formulation encourages the drone to learn a policy that efficiently completes deliveries while avoiding collisions and conserving battery.

2)

In Reinforcement Learning, the exploration-exploitation trade-off involves choosing between exploring new actions to discover their potential or exploiting known actions that yield high rewards. Exploration can uncover better strategies but might result in suboptimal rewards in the short term. Exploitation maximizes immediate rewards but may miss out on better long-term strategies.

The ε-greedy strategy balances this trade-off by occasionally choosing a random action (exploration) with probability ε, and most of the time selecting the best-known action (exploitation) with probability 1-ε. This approach allows the agent to explore new possibilities while primarily leveraging learned strategies to maximize rewards. As ε decreases over time, the strategy shifts from exploration to more focused exploitation, improving the overall policy efficiency as the agent becomes more knowledgeable.

3)

The expected value of the state v(s) can be calculated using the Bellman equation:

$v(s)=\sum P(s'|s,a)*[R(s'|s,a)+\gamma*v(s')]$

Given:

- $P(s'|s,a)=0.4$, $R=10$, and $V(s')=5$.
- $P(s''|s,a)=0.6$, $R=2$, and $V(s'')=3$.
- $\gamma=0.5$.

Now, plug in the values:

$V(s)=0.4*[10+0.5*5]+0.6*[2+0.5*3]$

Simplify the expressions inside the brackets:

$V(s)=0.4*[10+2.5]+0.6*[2+1.5]$

$V(s)=0.4*12.5+0.6*3.5$

$V(s)=5+2.1=7.1$

So, the expected value of the state V(s) is **7.1**