PROBLEM
Goal: Implement a reinforcement learning based path planning heuristic for autonomous robot agents that can explore and map unknown environments.

Method: RL algorithm, called Q-Nav, is proposed to train the agent learn how to interact with the environment, avoid obstacles and find optimal control policy for finding a path that allow more area coverage.

Use case: Autonomous robots that can explore unknown environments without relying on external positioning systems like GPS or human supervision, with practical applications in situations like rescue and mining.

IMPLEMENTATION
Q-Nav algorithm to predict the utility Q, of an action at each state and build a control policy:

1. Move one step selecting one of the possible actions at current state S(t)
2. Observe reward value (R) and new state S(t+1)
3. Update the state-action function Q(s, a) in order to reduce the estimation error.

\[
Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left( r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right)
\]

Exploration/Exploitation

Problem: Dilemma of whether to keep following actions/policy from past experience or explore other unknown parts of the world in hope of finding better paths?

- Greedy action selection: sometimes with a small probability select a random action instead of always the best according to the control policy.

- Exploration starts: start each new episode at a randomly selected state.

Reward Accumulation

Problem: Given enough time to discover rewarded states and greedy action selection policy, the agent can easily be confused to consider the intermediate states as its goal and stall the algorithm for longer execution time without terminating.

- Reward Shaping: add a shaping function F, that provide potential based values other than the native rewards R, in order to guide the agent in correct direction.

Credit Assignment Problem

Problem: Determine which state-action pair has to be awarded for delayed rewards. An action or combination of actions may have occurred several steps ago generating rewards at a later time.

EVALUATION
Various training techniques including reward shaping, eligibility traces and configuration parameters such as transition noise, discounting factor were observed for their effect on the training process.

CONCLUSION
The project provided details on how area exploration problem can be modeled as an optimization problem. A modified Q-learning algorithm was successfully implemented and showed to have performed better than A* algorithm in simple grid world simulations.

Different ways of improving the RL training process, especially, the importance of ‘reward shaping’ technique for complex environments were discussed.