1.

The configuration space (C-space) is the set of all possible robot configurations. A robot has limitations on the states that it can occupy, thus not being able to visit the entire state space of an environment. The restrictions on the robot are due to limitations on its degrees of freedom. A configuration of the robot can be represented as a list of each segment's length and the angles at each joint.

The given problem of finding of moving the robot arm between its original and goal state is one that can be solved through the use of a probabilistic roadmap (PRM). PRM is a motion planning algorithm which determines a path between a start and goal configuration while avoiding collisions with obstacles. PRM samples a set of states uniformly at random, thus the algorithm's performance can be unpredictable. To improve PRMs consistency and performance, a sampling strategy must be used as opposed to randomly sampling the configuration space.

2.

A sampling strategy is akin to a heuristic in the case of a discrete search, in that a certain strategy can be used to try and maximise efficiency. The implemented strategy samples the state space randomly, but takes advantage of a heuristic to determine the lowest cost of a randomly generated configuration. The first heuristic function used is the Euclidean distance between the tip of the robot arm (EE2) to the robot's next goal. The next goal of the arm is the closest unvisited grapple point, and in the case that all grapple points have been visited, the arm's goal becomes the goal configuration provided. The second heuristic used to calculate the cost of a given configuration is the sum of the distances between two configurations respective joints. This heuristic is a measure of how "easy" it is to move from the robot arm's current position to the next.

After all of the randomly generated configurations are prioritised according to our heuristics, a collision check is done to ensure that a configuration is not in collision with any obstacles or with itself. The highest prioritised configuration which is not in collision is then expanded, and new configurations are randomly generated from that configuration. This is a depth first search, taking a path via one node per level. To maximise the probability that the randomly generated configurations are near the goal state, as the euclidean distance of the robot's current configuration to the goal state reduces, so does the randomness of the configurations generated. This in turn causes the robot's path to include more fine steps as it approaches the goal state, reducing the probability of collision due to interpolation. Checking whether a state of the robot is in collision with an obstacle is trivial in this environment by using a combination of the bounding box method and line segment collision. This is because the obstacles are all rectangular and fully observable.

3.

The method used to solve the problem is complete, meaning that the arm will be able to find a path to its goal state if one exists (as tested on the provided levels). However, the solution will not always result in the shortest path to the goal. Our robot in essence uses a DFS search - meaning it is efficient timewise, however it is not optimal in the number of steps it finds for the robot to take. This means if a situation required the arm to reach the final position in the fewest steps possible, our robot would fail (whereas a BFS would perform better). On the other hand, our robot would be more successful in situations where it's necessary to process the path as fast as possible.

A negative aspect to the sampling strategy used is in the case of a very restricted environment, with many obstacles and small gaps between obstacles. The random sampling strategy used does not selectively sample closer to obstacles and inside passageways, and as such, could have its efficiency greatly increased in these scenarios if modified to do so. By sampling randomly in conjunction with the heuristics used however, the probability that the goal state is found quickly is very high in the case that the environment does not have many small passageways. The sampling strategy also makes it difficult for the arm to connect to a grapple point, in the scenario where the the grapple point 'collision' area is extremely small (and/or just has an extremely low tolerance value), due to the spread of the random sampling.