1. If we observe the algorithm used for implementing RRT, We can see that there is no constraint on how the tree expands so that the final path will be the shortest. It iteratively expands until it reaches the goal. Once, we reach the goal, the algorithm terminates. Thus, we are left with only one path and there is no notion of finding other paths as well and then comparing the paths for their optimality. The goal of RRT is to find a path from start to goal if it exists and not the optimal path.
In RRT, new point is sampled randomly. Thus, the branches of RRT grow as per the underlying probability distribution that governs sampling of new point. The tree does not grow with the path optimization criteria.
Because of the above reasons, Path returned by RRT is not guaranteed to be optimal.

2. Increasing the step-size will lead to having longer edges of the RRT tree. And as a result, the RRT will not be as dense. Also, increasing the step size, may lead to jerky (not-feasible) path. If there are fewer obstacles, RRT may converge quickly. However, it will not be able to explore/search the state space with finer resolution. As the number of obstacles change, larger step size might be a problem for identifying a path from source to destination.

3. According to S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning.Technical Report 98-11, Computer Science Dept., Iowa State University, Oct. 1998, “expansion in RRT is heavily based towards unexplored portions of the state space”. Also, according to the RRT algorithm, it samples new points randomly. Thus, if we increase the state space, it will take a longer time for RRT to converge. The computational cost of building the tree will also increase, as there is more space to cover.
Increasing the number of dimensions of the state space, indirectly increase the state space itself, thus it will increase the computational-cost of creating the RRT graph and time for convergence.

4. It is important to have a relatively smaller step-size to aid in refine search of the state-space. If we have a lot of small obstacles, it will become difficult to identify feasible points in the search space, that avoid obstacles and at the same time contribute towards the final path from source to destination (the path would be more jerky and non-feasible with larger step-size). As there are a lot of small obstacles, we won't be able to find ample number of points that avoid all the obstacles and yet reach the destination.