1: I would use BFS to solve the sudoku, where the initial state is a sudoku with all instance of an asterisk with all cell filled and goal state is an asterisk with all cell filled and satisfy all constraint. The state will be all possible value assignment state to the asterisk. The goal test of the search problem will be comparing the current state of the asterisk with the goal state. Also, for BFS algorithm, the successor function will be if a cell's assigned value does not falsify any constraint given by an asterisk, then it will move on to the next cell until every cell is complete. Otherwise, it will perform BSF backtracking.

2: A uniform cost search would be preferable to plain backtracking with MRV as the cells with the least remaining value would get a lower cost. A BFS algorithm would be preferable for backtracking as a cell falsify the constraint, then BFS will just backtracking.