Summary of Time and Space Complexity for NP-Hard Problems

Problem	Worst-case Time Complexity	Best-case Time Complexity	Worst-case Space Complexity	Best-case Space Complexity
Traveling Salesman	O(n!) or O(2^n * n)	Can be polynomial for small inputs or specific heuristics	O(n * 2^n) (dynamic programming)	O(n) (for greedy heuristics)
Knapsack Problem	O(n * W) (dynamic programming)	O(n) for greedy heuristic	O(n * W) (dynamic programming)	O(n) (for greedy approach)
Vertex Cover	O(2^n) or O(n!)	Polynomial for small graphs	O(n) for some special cases (like bounded treewidth)	O(n) for bounded treewidth graphs
Subset Sum Problem	O(2^n)	O(n) for small problems	O(n * W) (if using dynamic programming)	O(n) for specific instances
Graph Coloring	O(2^n) or O(n!)	Polynomial for certain graph classes	O(n^2) (if using DP for small graphs)	O(n) for specific graph structures
Independent Set	O(2^n)	Polynomial for certain graph types	O(2^n) or O(n)	O(n) for specific instances
Set Cover	O(2^m) or O(m * 2^m)	Polynomial for small k or special cases	O(2^m) or O(m * n)	O(m + n)
Clique Decision	O(2^n)	Polynomial for sparse graphs	O(2^n) or O(n * 2^n)	O(n) or better for sparse graphs
3-Satisfiability (3-SAT)	O(2^n)	Polynomial for specific instances	O(2^n)	O(n)