Coping With NP-Completeness

Special Cases
Average Case
Approximation Algorithms
Intelligent Brute Force
Heuristics
Special Cases

2-CNF Sat:

Use resolution:

\((xvyvz) \land (\overline{z} \lor w \lor \overline{v})\)

resolve to

\((xvyv\lor vwv\overline{v})\)

Sat iff \(\square\) (the empty clause)

cannot be obtained by resolution

Eliminate one variable at a time by resolving it in all possible ways

For 3-Sat, or general sat, clauses can get arbitrarily long: \(2^n\)
2-sat

\((x \lor \neg z) \land (\neg z \lor y)\) gives \(x \lor y\)

Resolution preserves 2-sat

\(O(n^2)\) possible clauses

\(O(n^3)\) time

Like transitive closure,

all-pairs shortest paths
Faster: Formula $\Rightarrow$ Graph

$(x \lor y) \Rightarrow \overline{x} \rightarrow y$ (edges)

$x \leftarrow \overline{y}$

Literals are vertices

Satisfiable iff no literal and its negation are in the same strong component. (Why?)

$O(m + n)$ where $m = \# \text{ clauses}$

$n = \# \text{ literals}$

(Can propagate single-literal clauses first, or use $x \equiv x \lor x \Rightarrow \overline{x} \rightarrow x$)
Special cases

Min vertex cover for a bipartite graph

Find a maximum matching.

Search for augmenting paths from free vertices on A side. Let reached vertices be $S$, others $\bar{S}$.

Let $C = (B \cap S) \cup (A \cap \bar{S})$

This is a minimum vertex cover.
Bns: all matched in $S$

No matched Bns to An$\bar{S}$ edges
No unmatched An$S$ to Bn$\bar{S}$ edges

$\Rightarrow (Bns) \cup (An\bar{S})$ is a vertex cover of size $\leq$ maximum matching

$\Rightarrow$ minimum (every edge of a matching must be covered)
Approximation

General vertex cover

Find a maximal matching (no new edges can be added)

$C = \text{both ends of all matched edges}

\text{covers since maximal matching is a 2-approximation}

O(m)-time
Approximation

Minimum tour (TSP) with $\Delta$

inequality:

\[ d(x, y) + d(y, z) \geq d(x, z) \]

\(\Rightarrow\) given any tour with repeats, can

find a tour no longer by dropping

repeats

Find a minimum spanning tree, build a tour as a depth-first

traversal (each edge used twice), delete repeated vertices.

2-approximation
1.5 approximation

Find an MST $T$

# odd-degree vertices is even

Find a min-cost perfect matching on odd degree vertices $P$

$T \cup P$ has all vertices of even degree:
Find an Eulerian tour, delete repeated vertices.

If $R$ is a min-cost tour $|T| \leq |R|$, $|P| \leq |R|/2 \Rightarrow |T+P| \leq 1.5|R|$

($\parallel$ denotes cost)
$R$ decomposes into two ways of pairing odd-degree vertices, gives to matchings of odd-degree vertices.