# Local (Heuristic) Search Methods

## IN4082

**Tomas Klos**

February 25, 2008



**Algorithmics Group**

Delft University of Technology

# Outline

"The course covers local search and its variants in combinatorial optimisation."

# Outline

“The course covers local search and its variants in combinatorial optimisation.”

- local search in SAT solving

# Outline

"The course covers local search and its variants in combinatorial optimisation."

- local search in SAT solving
- general principle

# Outline

"The course covers local search and its variants in combinatorial optimisation."

- local search in SAT solving
- general principle
- example technique

# Outline

"The course covers local search and its variants in combinatorial optimisation."

- local search in SAT solving
- general principle
- example technique
- more techniques

# Local Search in SAT solving

**for** $i = 1$ to MAX-TRIES **do**
  $\varphi \leftarrow$ random initial assignment
  **for** $j = 1$ to MAX-STEPS **do**
    **if** $\varphi$ satisfies $\mathcal{F}$ **then**
      **return** satisfiable
    **end if**
    $\varphi \leftarrow$ FLIP$(\varphi)$
  **end for**
**end for**
**return** unknown

# Local Search in SAT solving

**for** $i = 1$ to MAX-TRIES **do**
    $\varphi \leftarrow$ random initial assignment
    **for** $j = 1$ to MAX-STEPS **do**
        **if** $\varphi$ satisfies $\mathcal{F}$ **then**
            **return** satisfiable
        **end if**
        $\varphi \leftarrow$ FLIP($\varphi$)
    **end for**
**end for**
**return** unknown

Global, Local flips; WalkSAT, UnitWalk flips

# Local Search: General Approach

Outline:

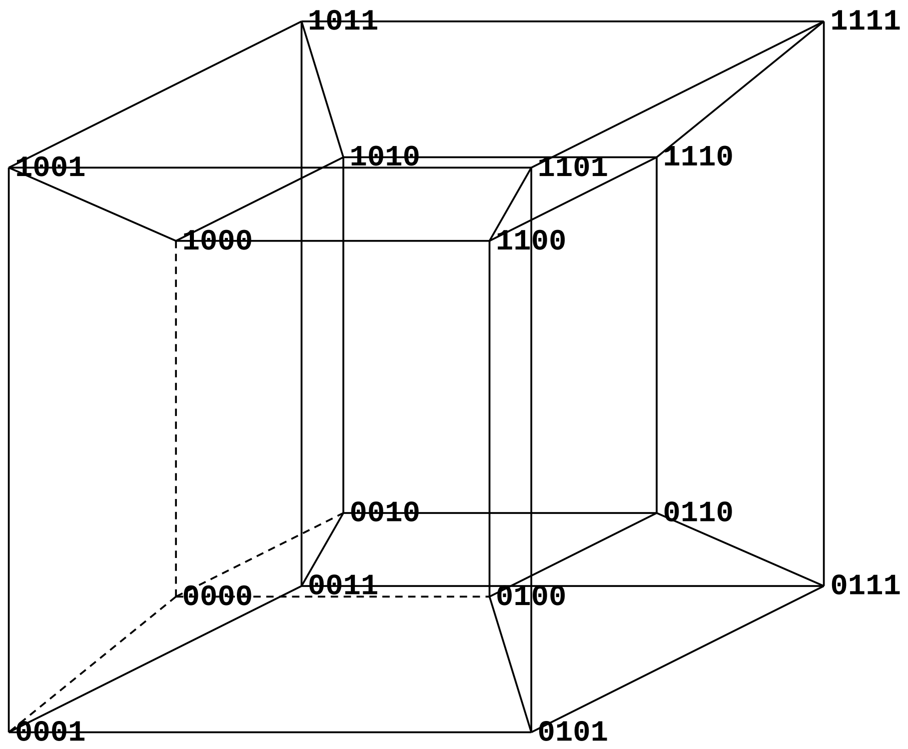
- solutions have ‘quality’ (cost $c$)
- define *neighborhood relation* on solutions
  e.g. flipping bits in bitstrings (graphs)
- start at (random) initial solution
- explore the space of possible solutions sequentially

# Local Search: General Approach

Outline:

- solutions have 'quality' (cost $c$)
- define *neighborhood relation* on solutions
  e.g. flipping bits in bitstrings (graphs)
- start at (random) initial solution
- explore the space of possible solutions sequentially
- in each step:
  - from current $S$, evaluate neighbors $S'$
  - move to $S'$ depending on $\Delta_c$ ($p$ in WalkSAT)

# Example: flipping bits








TUDelft

# Landscape Metaphor

Visualization:

- Solutions are points in an $N$-dimensional space
- Solution quality is the $N+1$-th dimension
- Myopically look for minima by traversing the landscape

# Landscape Metaphor

Visualization:

- Solutions are points in an $N$-dimensional space
- Solution quality is the $N + 1$-th dimension
- Myopically look for minima by traversing the landscape

Natural approach: gradient descent

- easily gets stuck in local optima
- $\rightarrow$ accept inferior neighbor with small probability

# Metropolis algorithm

Gibbs-Boltzmann function from Statistical Mechanics:

$$\text{Pr}(\text{state with energy } E) = e^{-E/(kT)}$$

# Metropolis algorithm

Gibbs-Boltzmann function from Statistical Mechanics:

$$\text{Pr}(\text{state with energy } E) = e^{-E/(kT)}$$

Metropolis Algorithm:

- current state is $S$, perturbation is $S'$
- if $E(S') \leq E(S)$, update system to $S$
- else $\Delta E = E(S') - E(S) > 0$
- update to $S'$ with prob. $e^{-\Delta E/(kT)}$

# Metropolis algorithm

Gibbs-Boltzmann function from Statistical Mechanics:

$$\text{Pr}(\text{state with energy } E) = e^{-E/(kT)}$$

Metropolis Algorithm:

- current state is $S$, perturbation is $S'$
- if $E(S') \leq E(S)$, update system to $S$
- else $\Delta E = E(S') - E(S) > 0$
- update to $S'$ with prob. $e^{-\Delta E/(kT)}$

Fails to converge when close to OPT: decrease temperature over time (Simulated Annealing)

# More Local Search heuristics

Physics

- simulated annealing
- particle swarm optimization

Biology

- artificial neural networks
- artificial immune systems
- ant colony optimization

Evolution (multiple parallel solutions)

- evolutionary algorithms (GA's, GP)

# Other Topics

- easy to construct LS algorithm
- hard to get analytical performance guarantees
- evaluate performance experimentally

# LS: Best Response Dynamics

Congestion game:

- Many agents, conflicting objectives
- Cost of an edge depends on # of agents using it
- Conflict between individual vs. global
- Best Response: make a choice given others' choices
- Iteratively: Best Response Dynamics
- Stability: all strategies are best responses (Nash)
- Optimality: what is the global cost?

How does global relate to individual performance?

# LS: Combinatorial Auctions

Setup:

- $n$ agents, $m$ goods $\Omega$ on auction
- agents value bundles of goods $S \subseteq \Omega$
- $v_i(S) + v_i(T) \leq v_i(S \cup T)$

# LS: Combinatorial Auctions

Setup:

- $n$ agents, $m$ goods $\Omega$ on auction
- agents value bundles of goods $S \subseteq \Omega$
- $v_i(S) + v_i(T) \leq v_i(S \cup T)$

Problem: find allocation $A = (S_1, S_2, \ldots, S_n)$

# LS: Combinatorial Auctions

Setup:

- $n$ agents, $m$ goods $\Omega$ on auction
- agents value bundles of goods $S \subseteq \Omega$
- $v_i(S) + v_i(T) \leq v_i(S \cup T)$

Problem: find allocation $A = (S_1, S_2, \ldots, S_n)$ s.t.

$$A = \arg\max_{A \in \mathcal{A}} \sum_{i=1}^{n} v_i(S_i)$$

# LS: Combinatorial Auctions

Setup:

- $n$ agents, $m$ goods $\Omega$ on auction
- agents value bundles of goods $S \subseteq \Omega$
- $v_i(S) + v_i(T) \leq v_i(S \cup T)$

Problem: find allocation $A = (S_1, S_2, \ldots, S_n)$ s.t.

$$A = \arg\max_{A \in \mathcal{A}} \sum_{i=1}^{n} v_i(S_i) \quad \text{(NP-hard)}$$

# LS: Combinatorial Auctions

Setup:

- $n$ agents, $m$ goods $\Omega$ on auction
- agents value bundles of goods $S \subseteq \Omega$
- $v_i(S) + v_i(T) \leq v_i(S \cup T)$

Problem: find allocation $A = (S_1, S_2, \ldots, S_n)$ s.t.

$$A = \arg\max_{A \in \mathcal{A}} \sum_{i=1}^{n} v_i(S_i) \quad \text{(NP-hard)}$$

LS: start with $A = (\{1, 2\}, \{13, 6\}, \ldots, \{4, 7, 10\})$