

Integer Linear Programming

Tomáš Vinař
December 16, 2021

Practical programs for NP-hard problems

They always find the optimal solution, often in reasonable time, but on some inputs very long runtimes

- ILP: CPLEX, Gurobi (commercial), SCIP (non-commercial)
- SAT: Minisat, Lingeling, glucose, CryptoMiniSat, painsless
- TSP: Concorde

Other NP-complete problems can be transformed to one of these problems

ILP: Integer linear programming

Linear programming:

real-valued variables x_1, \dots, x_n

minimize $\sum_i a_i x_i$ for given weights a_1, \dots, a_n

under constraints of the form $\sum_i b_i x_i \leq c$

LP can be solved in polynomial time

Integer linear programming:

Add a constraint that some variables are integers or binary

NP-hard problem

Expressing known NP-hard problems as ILP

Knapsack

Given n items with weights $w_1 \dots w_n$ and costs $c_1 \dots c_n$.

Choose a subset so that overall weight is at most T and the overall cost is highest possible?

Expressing known NP-hard problems as ILP

Set cover

We have n subsets $S_1 \dots, S_n$ of a set $U = \{1 \dots m\}$.

Choose the smallest number of the input subsets so that their union is the whole set U .

Protein threading

Protein A has a known sequence and structure, protein B only sequence.

Align A and B so that if two amino acids are close in A , their equivalents in B should be “compatible”.

Choose “cores” in A which should remain conserved without insertions, deletions and in the same order

Cores are separated by “loops”, whose length can arbitrarily change and whose alignments will not be scored

Protein threading, problem formulation

Input: sequence $B = b_1 \dots b_n$,

lengths of m cores $c_1 \dots c_m$,

scoring tables

- E_{ij} : how well $b_j \dots b_{j+c_i-1}$ agrees with sequence of core i ,
- E_{ijkl} : how well would cores i and k interact, if they start at pos. j, ℓ .

Task: choose starts of cores x_1, x_2, \dots, x_m so that

- they are in the correct order and without overlaps,
- they achieve maximum possible score

Note: we do not specify how to choose cores and scoring tables, which is a modeling, not an algorithmic problem

Protein threading, ILP

Notation: sequence $B = b_1 \dots b_n$, lengths of m cores $c_1 \dots c_m$,

E_{ij} : how well $b_j \dots b_{j+c_i-1}$ agrees with sequence of core i ,

E_{ijkl} : how well would cores i and k interact, if they start at pos. j, ℓ ,
unknown starts of cores x_1, \dots, x_m .

ILP formulation:

Protein threading, ILP

Notation: sequence $B = b_1 \dots b_n$, lengths of m cores $c_1 \dots c_m$,

E_{ij} : how well $b_j \dots b_{j+c_i-1}$ agrees with sequence of core i ,

E_{ijkl} : how well would cores i and k interact, if they start at pos. j, ℓ ,
unknown starts of cores x_1, \dots, x_m .

ILP formulation: