Subsection 2

NP-hardness

NP-Hardness

- Do *hard* problems exist? Depends on $\mathbf{P} \neq \mathbf{NP}$
- Next best thing: define *hardest problem in* NP
- A problem P is NP-*hard* if
  Every problem Q in NP can be solved in this way:
  1. given an instance q of Q transform it in polytime to an instance $\rho(q)$ of P s.t. q is YES iff $\rho(q)$ is YES
  2. run the best algorithm for P on $\rho(q)$, get answer $\alpha \in \{\text{YES}, \text{NO}\}$
  3. return α

  ρ is called a *polynomial reduction* from Q to P
- If P is in NP and is NP-hard, it is called NP-*complete*
- Every problem in NP reduces to SAT [Cook 1971]

Cook's theorem

Theorem 1: If a set S of strings is accepted by some nondeterministic Turing machine within polynomial time, then S is P-reducible to {DNF tautologies}.

Boolean decision variables store TM dynamics

Proposition symbols:

$P^i_{s,t}$ for $1 \le i \le \ell$, $1 \le s,t \le T$. $P^i_{s,t}$ is true iff tape square number s at step t contains the symbol σ_i.

Q^i_t for $1 \le i \le r$, $1 \le t \le T$. Q^i_t is true iff at step t the machine is in state q_i.

$S_{s,t}$ for $1 \le s,t \le T$ is true iff at time t square number s is scanned by the tape head.

Definition of TM dynamics in CNF

B_t asserts that at time t one and only one square is scanned:

$$B_t = (S_{1,t} \vee S_{2,t} \vee \ldots \vee S_{T,t}) \;\&\; [\mathop{\&}_{1 \le i < j \le T} (\neg S_{i,t} \vee \neg S_{j,t})]$$

$G^t_{i,j}$ asserts that if at time t the machine is in state q_i scanning symbol σ_j, then at time $t+1$ the machine is in state q_k, where q_k is the state given by the transition function for M.

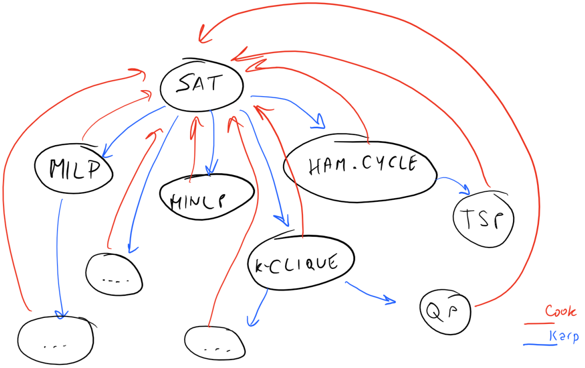
$$G^t_{i,j} = \mathop{\&}_{s=1}^{T} (\neg Q^i_t \vee \neg S_{s,t} \vee \neg P^j_{s,t} \vee Q^k_{t+1})$$

Description of a dynamical system using a declarative programming language (SAT) — what MP is all about!

Reduction graph

After Cook's theorem

To prove NP-hardness of a new problem P, pick a known NP-hard problem Q that "looks similar enough" to P and find a polynomial reduction ρ from Q to P [Karp 1972]



Why it works: suppose P easier than Q, solve Q by calling $\rho \circ \mathrm{Alg}_P$, conclude Q as easy as P, contradiction

Example of polynomial reduction

- STABLE: given $G = (V, E)$ and $k \in \mathbb{N}$, does it contain a stable set of size k?
- We know k-CLIQUE is NP-complete, reduce from it
  - Given instance (G, k) of CLIQUE consider the *complement graph* (computable in polytime)

$$\bar{G} = (V, \bar{E} = \{\{i, j\} \mid i, j \in V \wedge \{i, j\} \notin E\})$$

  - **Thm.:** G has a clique of size k iff $\bar{G}$ has a stable set of size k
  - $\rho(G) = \bar{G}$ is a polynomial reduction from CLIQUE to STABLE
- $\Rightarrow$ STABLE is $\mathbb{NP}$-hard
- STABLE is also in NP

  $U \subseteq V$ is a stable set iff $E(G[U]) = \varnothing$ (polytime verification)
- $\Rightarrow$ STABLE is $\mathbb{NP}$-complete

MILP is NP-hard

- SAT is NP-hard by Cook's theorem, Reduce from SAT in CNF

$$\bigwedge_{i \le m} \bigvee_{j \in C_i} \ell_j$$

  where ℓ_j is either x_j or $\bar{x}_j \equiv \neg x_j$

- Polynomial reduction ρ

SAT	x_j	$\bar{x}_j$	$\vee$	$\wedge$
MILP	x_j	$1 - x_j$	$+$	≥ 1

- E.g. ρ maps $(x_1 \vee x_2) \wedge (\bar{x}_2 \vee x_3)$ to

$$\min\{0 \mid x_1 + x_2 \ge 1 \wedge x_3 - x_2 \ge 0 \wedge x \in \{0,1\}^3\}$$

- SAT is YES iff MILP is feasible
  (same solution, actually)

Complexity of Quadratic Programming

$$\left.\begin{array}{rrcl} \min & x^\top Q x & + & c^\top x \\ & Ax & \geq & b \end{array}\right\}$$

- Quadratic Programming = **QP**
- **Quadratic objective, linear constraints, continuous variables**
- **Many applications (e.g. portfolio selection)**
- **If Q PSD then objective is convex, problem is in P**
- **If Q has at least one negative eigenvalue, NP-hard**
- *Decision problem: "is the min. obj. fun. value $= 0$?"*

QP is NP-hard

- By reduction from SAT, let σ be an instance
- $\hat{\rho}(\sigma, x) \geq 1$: linear constraints of SAT $\rightarrow$ MILP reduction
- Consider QP

$$\left.\begin{array}{ll} \min & f(x) = \sum_{j \leq n} x_j(1 - x_j) \\ & \hat{\rho}(\sigma, x) \geq 1 \\ & 0 \leq x \leq 1 \end{array}\right\} \quad (\dagger)$$

- **Claim:** σ is YES iff $\mathsf{val}(\dagger) = 0$
- *Proof:*
  - assume σ YES with soln. x^*, then $x^* \in \{0, 1\}^n$, hence $f(x^*) = 0$, since $f(x) \geq 0$ for all x, $\mathsf{val}(\dagger) = 0$
  - assume σ NO, suppose $\mathsf{val}(\dagger) = 0$, then $(\dagger)$ feasible with soln. x', since $f(x') = 0$ then $x' \in \{0, 1\}$, feasible in SAT hence σ is YES, contradiction

Box-constrained QP is NP-hard

- **Add surplus vars v to SAT$\to$MILP constraints:**
  $\hat{\rho}(\sigma, x) - 1 - v = 0$
  (denote by $\forall i \le m \ (a_i^\top x - b_i - v_i = 0)$)
- **Now sum them on the objective**

$$\left.\begin{array}{ll} \min & \sum_{j \le n} x_j(1 - x_j) + \sum_{i \le m} (a_i^\top x - b_i - v_i)^2 \\ & 0 \le x \le 1, v \ge 0 \end{array}\right\}$$

- **Issue: v not bounded above**
- **Reduce from 3SAT, get ≤ 3 literals per clause**
  $\Rightarrow$ ***can consider*** $0 \le v \le 2$

cQKP is NP-hard

- CONTINUOUS QUADRATIC KNAPSACK PROBLEM (cQKP)

$$\left.\begin{array}{rcl} \min \quad f(x) = x^\top Q x & + & c^\top x \\ \sum_{j \le n} a_j x_j & = & \gamma \\ x & \in & [0,1]^n, \end{array}\right\}$$

- Reduction from SUBSET-SUM

  given list $a \in \mathbb{Q}^n$ and γ, is there $J \subseteq \{1, \ldots, n\}$ s.t. $\sum_{j \in J} a_j = \gamma$?

  reduce to $f(x) = \sum_j x_j(1 - x_j)$
- σ is a YES instance of SUBSET-SUM
  - let $x^*_j = 1$ iff $j \in J$, $x^*_j = 0$ otherwise
  - feasible by construction
  - f is non-negative on $[0,1]^n$ and $f(x^*) = 0$: optimum
- σ is a NO instance of SUBSET-SUM
  - suppose $\mathsf{opt}(\text{cQKP}) = x^*$ s.t. $f(x^*) = 0$
  - then $x^* \in \{0,1\}^n$ because $f(x^*) = 0$
  - feasibility of $x^* \to \mathsf{supp}(x^*)$ solves σ, contradiction, hence $f(x^*) > 0$

QP on a simplex is NP-hard

$$\left.\begin{array}{rrcl} \min & f(x) = x^\top Q x & + & c^\top x \\ & \sum_{j \le n} x_j & = & 1 \\ \forall j \le n & x_j & \ge & 0 \end{array}\right\}$$

- **Reduce MAX CLIQUE to subclass** $f(x) = -\sum_{\{i,j\} \in E} x_i x_j$

  Motzkin-Straus formulation (MSF)

- **Theorem [Motzkin& Straus 1964]**

  Let C be the maximum clique of the instance $G = (V, E)$ of MAX CLIQUE

  $\exists x^* \in \mathsf{opt}\,(\textbf{MSF}) \qquad f^* = f(x^*) = \frac{1}{2}\left(1 - \frac{1}{\omega(G)}\right)$

  $\forall j \in V \qquad x_j^* = \begin{cases} \frac{1}{\omega(G)} & \textbf{if } j \in C \\ 0 & \textbf{otherwise} \end{cases}$

Proof of the Motzkin-Straus theorem

$$x^* = \mathsf{opt}(\max_{\substack{x\in[0,1]^n \\ \sum_j x_j = 1}} \sum_{ij\in E} x_i x_j) \text{ s.t. } |C = \{j \in V \mid; x_j^* > 0\}| \text{ smallest } (\ddagger)$$

1. *C is a clique*
   - Suppose $1, 2 \in C$ but $\{1, 2\} \not\in E[C]$, then $x_1^*, x_2^* > 0$, can perturb by small $\epsilon \in [-x_1^*, x_2^*]$, get $x^\epsilon = (x_1^* + \epsilon, x_2^* - \epsilon, \ldots)$, feasible w.r.t. simplex and bounds
   - $\{1, 2\} \not\in E \Rightarrow x_1 x_2$ does not appear in $f(x) \Rightarrow f(x^\epsilon)$ depends linearly on ϵ; by optimality of x^*, f achieves max for $\epsilon = 0$, in interior of its range $\Rightarrow f(\epsilon)$ constant
   - set $\epsilon = -x_1^*$ or $= x_2^*$ yields global optima with more zero components than x^*, against assumption ($\ddagger$), hence $\{1, 2\} \in E[C]$, by relabeling C is a clique

Proof of the Motzkin-Straus theorem

$$x^* = \mathsf{opt}(\max_{\substack{x\in[0,1]^n \\ \sum_j x_j = 1}} \sum_{ij\in E} x_i x_j) \text{ s.t. } |C = \{j \in V \mid; x_j^* > 0\}| \text{ smallest } (\ddagger)$$

2. $\boxed{|C| = \omega(G)}$

- square simplex constraint $\sum_j x_j = 1$, get

$$\sum_{j\in V} x_j^2 + 2 \sum_{i<j\in V} x_i x_j = 1$$

- by construction $x_j^* = 0$ for $j \notin C \Rightarrow$

$$\psi(x^*) = \sum_{j\in C} (x_j^*)^2 + 2 \sum_{i<j\in C} x_j^* x_j^* = \sum_{j\in C} (x_j^*)^2 + 2f(x^*) = 1$$

- $\psi(x) = 1$ for all feasible x, so $f(x)$ achieves maximum when $\sum_{j\in C}(x_j^*)^2$ is minimum, i.e. $x_j^* = \frac{1}{|C|}$ for all $j \in C$
- again by simplex constraint

$$f(x^*) = 1 - \sum_{j\in C} (x_j^*)^2 = 1 - |C|\frac{1}{|C|^2} \le 1 - \frac{1}{\omega(G)}$$

so $f(x^*)$ attains maximum $1 - 1/\omega(G)$ when $|C| = \omega(G)$

Portfolio optimization

You, a private investment banker, are seeing a customer. She tells you "I have 3,450,000$ I don't need in the next three years. Invest them in low-risk assets so I get at least 2.5% return per year."

Model the problem of determining the required portfolio. Missing data are part of the fun (and of real life).

[Hint: what are the decision variables, objective, constraints? What data are missing?]