Chapter 2
Integer Programming

Paragraph 1
Total Unimodularity

What we did so far

- We studied algorithms for solving linear programs
  - Simplex (primal, dual, and primal-dual)
  - Ellipsoid Method (proving LP is in P)
  - Interior Point Algorithms
- We looked into standard formats that interface between the LPs that we want to solve and the available programs that we can use to solve our problems.
  - MPS
  - LP
  - CPLEX callable library

How Powerful is Linear Programming?

- LPs can be solved in polynomial time.
- Is that good news or bad news?
  - Of course, it is good news, because it means that we can efficiently solve a very large class of problems. ☺
  - On the other hand, if we believe that NP≠P, it also means that we can solve no NP-hard problems. ☹
- In order to model NP-hard problems as well, we need more expressiveness:
  - We need to be able to make real discrete decisions.
  - By allowing one more type of constraint, we can achieve this expressiveness.
  - The constraint being: Let some or all variables be integer.

Integer Programming

- Standard from: Minimize $c^T x$ such that
  - $Ax = b$
  - $x \geq 0$
  - x_i integer for all $i \in I \subseteq \{1,..,n\}$.
- Canonical form analogously to LP.
- Can we solve NP-hard problems now? Model 3-SAT:
  - One binary variable $x_i \in \{0,1\}$ for every X_i in the formula.
  - For each clause $(X_i \vee \neg X_j \vee X_k)$ in the formula add a constraint
    - $x_i + (1-x_j) + x_k \geq 1 \Leftrightarrow x_i - x_j + x_k \geq 0$
  - For any clause: $\bigvee_{i \in I} X_i \ \bigvee_j \neg X_{j \in J}$ is modeled as
    - $\Sigma_{i \in I} x_i - \Sigma_{j \in J} x_j \geq 1 - |J|$

Integer Programming

- Model the Knapsack Problem
  - Max $p^T x$
    - $w^T x \le C$
    - $x \in \{0,1\}^n$
- Model the Shortest Path Problem
  - Min $\Sigma_{(i,j) \in E}\, c_{ij}\, x_{ij}$
    - $\Sigma_{(s,j) \in E}\, x_{sj} - \Sigma_{(i,s) \in E}\, x_{is} = 1$ for the source node s
    - $\Sigma_{(k,j) \in E}\, x_{kj} - \Sigma_{(i,k) \in E}\, x_{ik} = 0$ for all nodes $s \ne k \ne t$
    - $\Sigma_{(i,t) \in E}\, x_{it} - \Sigma_{(t,j) \in E}\, x_{tj} = -1$ for the sink node t
    - $x_{ij} \in \{0,1\}$ for all $(i,j) \in E$

Integer Programming

- Model the Assignment Problem
  - Min $\Sigma_{(i,j) \in E}\, c_{ij}\, x_{ij}$
    - $\Sigma_j\, x_{ij} = 1$ for all jobs j
    - $\Sigma_j\, x_{ij} = 1$ for all machines i
    - $x \in \{0,1\}^n$
- Model the Transportation Problem
  - Min $\Sigma_{fr}\, c_{fr}\, x_{fr}$
    - $\forall_f\, X_{fr} = D_r$ for all retailers r
    - $\forall_r\, X_{fr} \leq S_f$ for all factories f
    - x integer

Integer Programming

- Model Graph Bisection
  - Min $\Sigma_{(i,j) \in E} z_{ij}$
    - $z_{ij} \geq x_i - x_j$
    - $z_{ij} \geq x_j - x_i$
    - $\Sigma_j x_i = n/2$
    - $z_{ij}, x_i \in \{0,1\}$

Complexity

- Some of the above problems are
  - NP-hard but approximable (like KP), some are even
  - NP-hard in the strong sense (like Graph Bisection),
  - but others are poly-time solvable (like the Shortest Path Problem).
- Can we model problems like Shortest Path as an LP? More generally: Under what circumstances is an LP solution guaranteed to be integer?

Total Unimodularity

- The solution to an LP returned by the simplex algorithm is $x^T=(b^T A_B^{-T}, 0_N^T)$.
- Consider the quadratic system By=b (where we think of $B=A_B$ and $y=x_B$, of course). Denote with B_b^j the matrix B where the j'th column is replaced by b, i.e. $B_b^j = (B_1, .., B_{j-1}, b, B_{j+1}, .., B_m)$.
- According to Cramer's rule, we have that
  - $y_j = |B_b^j| / |B|$, where |X| denotes the determinant of matrix X.

Total Unimodularity

- Denote with B^{ij} the matrix that evolves from B when deleting the ith row and the jth column.
- We can compute the determinant of B_b^j in the following way:
  - $|B_b^j| = \Sigma_i (-1)^{i+j} b_i |B^{ij}|$.
- Consequently, when
  - $|B| = -1$ or $|B| = 1$ and
  - $|B^{ij}| \in \{-1,0,1\}$ for all i,j,

then y is integer.

Total Unimodularity

- Definition
  - A submatrix of a matrix A is any square matrix that evolves from A by deleting some columns and rows from A.
  - A matrix A is called totally unimodular (TU), iff the determinants of all submatrices of A are either
    -1, 0, or 1.
- Theorem
  - A polytope $P=\{ x \mid Ax=b, x \geq 0 \}$ with A TU and b integer has only integer basic solutions.
  - An IP in standard form over a TU matrix and with integer right hand side is solvable in polynomial time.
  - Any IP over a TU matrix and with integer right hand side is solvable in polynomial time.

Total Unimodularity

- Theorem [TU Partition]
  - A is TU iff for all $I \subseteq \{1,..,m\}$ there exists a partition of I into K and L such that for all $j \in \{1,..,n\}$ it holds that $| \Sigma_{i \in K} a_{ij} - \Sigma_{i \in L} a_{ij} | \leq 1$.
- Proof: See for instance Nemhauser/Wolsey "Integer and Combinatorial Optimization" #543.
- Corollary
  - A is TU if
    - it only has at most two non-zero entries 1 or -1 in every column, and
    - for all columns with two non-zero coefficients, the column-sum is 0.

Total Unimodularity

- Theorem
  - Given A TU. Then, the following matrices are TU:
    1. A^T is TU.
    2. (A,I_m) is TU.
    3. (A,-A) is TU.
    4. A^{-1} (if A is square and non-singular).
  - A remains TU under the following operations:
    5. Deleting or duplicating a row or column.
    6. Multiplying a row or column with -1.
    7. Permuting rows or columns.
    8. Performing a pivot operation on A.

Total Unimodularity

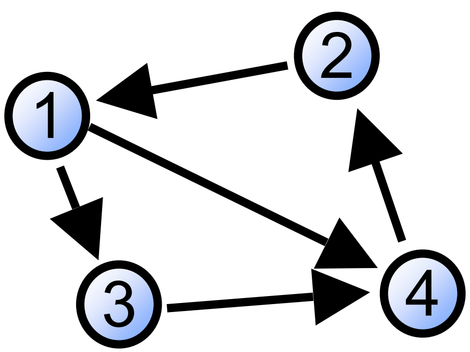
- Proof:
  1. $|X| = |X^T|$.
  2. Any submatrix of (A,I) can be row-permuted so that it takes form $$C = \begin{pmatrix} B & 0 \\ D & I_k \end{pmatrix}$$ Therefore, $|C| = |B|$.
  3. Implied by 5 and 6.
  4. Implied by 2 and 8.
  5. Any non-singular submatrix only contains one of the two rows or columns in question.

  6. and 7. Follows from the TU Partition Theorem.

  8. Exercise

Graph LPs

- A node-arc incidence matrix has a row for each node and a column for each edge of a given graph. Every column belonging to edge $(i,j) \in E$ contains exactly two non-zero entries: a -1 in row i and a 1 in row j.



-1	-1	1		
		-1		1
1			-1	
	1		1	-1

Assignments

- An assignment matrix looks like this:

1	1	1							
			1	1	1				J_1
						1	1	1	
1			1			1			
	1			1			1		J_2
		1			1			1	

Total Unimodularity

- Corollary
  - If A is a node-arc incidence matrix, then it is TU. This implies that the Shortest Path Problem and the Transportation Problem are in P. So are all kinds of flow problems.
  - If A is an assignment matrix, then it is TU. This implies that the Assignment Problem is in P.

Thank you!