Exam WI4410 Advanced Discrete Optimization

June 28, 2017, 13:30–16:30

The exam consists of 6 questions. In total you can obtain 60 points. Your grade is calculated by dividing the number of points you obtained by 6. You may use a non-graphical calculator during the exam. Using a graphical calculator, notes, phone, smart-watch, etc. is **not** permitted. The total number of pages of this exam is 8. Good luck!

---

1. Given a set $\{\boldsymbol{x} \in \mathbb{Z}^n \mid \sum_{j=1}^n a_j x_j = a_0\}$ with $a_0 \notin \mathbb{Z}$, the Gomory fractional cut based on this set is:

$$\sum_{j=1}^n f_j x_j \geq f_0,$$

where $f_j := a_j - \lfloor a_j \rfloor$ and $f_0 := a_0 - \lfloor a_0 \rfloor$.

Similarly, the Gomory Mixed-Integer Cut based on the set $\{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n \times \mathbb{R}^P \mid \sum_{j=1}^n a_j x_j + \sum_{j=1}^p g_j y_j = a_0\}$ with $a_0 \notin \mathbb{Z}$, is

$$\sum_{f_j \leq f_0} f_j x_j + \sum_{f_j > f_0} \frac{f_0(1-f_j)}{1-f_0} x_j + \sum_{g_j > 0} g_j y_j - \sum_{g_j < 0} \frac{f_0}{1-f_0} g_j y_j \geq f_0$$

Consider the following integer optimization problem:

$$\begin{aligned} \max\ & 4x_1 + 3x_2 \\ \text{subject to } & 2x_1 + x_2 \leq 11 \\ & -x_1 + 2x_2 \leq 6 \\ & x_1,\ x_2 \geq 0, \text{ integer} \end{aligned}$$

Let s_1 and s_2 be integer slack variables in the above constraints. After solving the LP-relaxation of the problem we obtain:

$$\begin{array}{lllllll} z & & & +\frac{11}{5}s_1 & +\frac{2}{5}s_2 & = & \frac{133}{5} \\ & & x_2 & +\frac{1}{5}s_1 & +\frac{2}{5}s_2 & = & \frac{23}{5} \\ & x_1 & & +\frac{2}{5}s_1 & -\frac{1}{5}s_2 & = & \frac{16}{5} \end{array}$$

(a) (4 points) Generate a Gomory fractional cut *and* a Gomory Mixed-Integer Cut from the last row of the system of equations above.

Solution:

$$f_0 = \frac{1}{5},\ f_{s_1} = \frac{2}{5},\ f_{s_2} = \frac{4}{5}\,.$$

Gomory fractional cut:

$$\frac{2}{5}s_1 + \frac{4}{5}s_2 \geq \frac{1}{5}, \quad \text{or } 2s_1 + 4s_2 \geq 1\,.$$

Gomory mixed-integer cut:

$$\frac{\frac{1}{5}\cdot\frac{3}{5}}{\frac{4}{5}}s_1 + \frac{\frac{1}{5}\cdot\frac{1}{5}}{\frac{4}{5}}s_2 \geq \frac{1}{5}$$

or

$$\frac{3}{4}s_1 + \frac{1}{4}s_2 \geq 1\,.$$

(b) (2 points) Which of the two inequalities is stronger? A brief motivation suffices.

Solution: The two inequalities have the same right-hand side, and each of the coefficients in the left-hand side of the GMIC is smaller than or equal to the corresponding coefficient in the Gomory fractional cut. Therefore the GMIC is stronger. This also holds in general!

2. Let $N := \{1, \ldots, n\}$. Consider the following set:

$$S_K := \{\boldsymbol{x} \in \{0,1\}^n \mid \sum_{j \in N} a_j x_j \leq b\}\,.$$

Assume that

- $a_j \in \mathbb{Z}_+$ for all $j \in N$,
- $b \in \mathbb{Z}_+$,
- $a_j < b$ for all $j \in N$,

(a) (4 points) Determine the dimension of conv(S_K).

Solution: Due to the assumptions, each of the unit vectors is feasible. These unit vectors form, together with the zero-vector, a set of $n+1$ affinely independent points. Hence, the dimension of conv(S_K) is equal to n.

(b) (3 points) Consider the specific knapsack set

$$S_K = \{x \in \{0,1\}^4 \mid 25x_1 + 20x_2 + 15x_3 + 10x_4 \leq 44\}\,.$$

The knapsack cover inequality $x_2 + x_3 \leq 1$ is valid for the set

$$\text{conv}(S_K \cap \{x \in \mathbb{R}^4 \mid x_4 = 1\})\,.$$

Apply maximal lifting to the variable x_4 and give the resulting valid inequality.

Solution: Introduce x_4 in the inequality. Don't forget that x_4 is currently set equal to 1:

$$x_2 + x_3 + \beta x_4 \leq 1 + \beta\,.$$

Now set x_4 equal to 0 and apply maximal lifting

$$\beta = \max\{x_2 + x_3 \mid 25x_1 + 20x_2 + 15x_3 \leq 44\} - 1\,,$$

which yields $\beta = 1$ and the resulting inequality $x_2 + x_3 + x_4 \leq 2$.

(c) (2 points) Consider the specific knapsack set

$$S_K = \{x \in \{0,1\}^5 \mid 27x_1 + 25x_2 + 20x_3 + 15x_4 + 10x_5 \leq 44\}\,.$$

The inequality $x_3 + x_4 + x_5 \leq 2$ is a valid cover inequality for this knapsack set. Argue why the inequality

$$x_1 + x_2 + x_3 + x_4 + x_5 \leq 2$$

is also valid for this set.

Solution: Since both x_1 and x_2 have larger coefficients in S_K than all the variables in the cover, x_1 and x_2 are part of the extension $E(C)$ of the cover C. Recall, $E(C) := C \cup \{j \in N \setminus C \mid a_k \geq a_j \text{ for all } j \in C\}$. The inequality

$$\sum_{j \in E(C)} x_j \leq |C| - 1$$

is valid for S_K, and hence the inequality $x_1 + x_2 + x_3 + x_4 + x_5 \leq 2$ is valid for the given knapsack set.

(d) (4 points) Let $N = \{1, \ldots, n\}$ be a set of arcs. Consider the single-node flow set:

$$S_{SNF} := \{(\boldsymbol{x},\ \boldsymbol{y}) \in \mathbb{R}^n_+ \times \{0,1\}^n \mid \sum_{j \in N} x_j = b,\ x_j \leq u_j y_j\}\,.$$

A set $C \subseteq N$ is a *flow cover* if $\sum_{j \in C} u_j > b$. Let $\lambda := \sum_{j \in C} u_j - b$ and $(u_j - \lambda)^+ := \max(u_j - \lambda,\ 0)$. Prove that the flow cover inequality

$$\sum_{j \in C} x_j + \sum_{j \in C} (u_j - \lambda)^+ (1 - y_j) \leq b$$

is valid for conv(S_{SNF}). (Hint: Consider the case where $y_j = 1$ for all $j \in C$ and then the case where an arbitrary arc $k \in C$ is closed.)

Solution: Let $y_j = 1$ for all $j \in C$:
Then $(1 - y_j) = 0$ for all $j \in C$ and the flow cover inequality becomes $\sum_{j \in C} x_j \leq b$ which is valid by definition.
Let $y_k = 0$ and $y_j = 1$ for all $j \in C \setminus \{k\}$:

Now, $\sum_{j\in C} x_j \le \min(b,\ \sum_{j\in C} u_j - u_k)$. Substitute $\sum_{j\in C} u_j$ by $b+\lambda$, which yields $\sum_{j\in C} x_j \le \min(b,\ b+\lambda-u_k) = \min(b,\ b-(u_k-\lambda)) = (b-(u_k-\lambda)^+)$. The coefficient for $(1-y_k)$ in the flow cover inequality is precisely $(u_k-\lambda)^+$ and will "count" as soon as y_k takes value 0.

(e) (1 point) Illustrate by a 2-dimensional example why branch-and-bound is not a polynomial-time algorithms in fixed dimension.

Solution: See slides 33 and 34 of Lecture 4.

3. Consider the quadratic assignment problem $QAP(A,B,C)$:

$$z^* = \min_{\varphi\in\mathcal{S}_n} z(\varphi), \text{ where } z(\varphi) = \sum_{i=1}^n\sum_{j=1}^n a_{ij}b_{\varphi(i)\varphi(j)} + \sum_{i=1}^n\sum_{j=1}^n c_{i\varphi(j)},$$

where $A=(a_{ij})$, $B=(b_{ij})$, and $C=(c_{ij})$ are real $n\times n$ matrices, and $\mathcal{S}_n$ denotes the set of all permutations of $\{1,\dots,n\}$.

(a) (5 points) Derive an expression for the average value of the objective function, namely

$$\mu(A,B,C) := \frac{1}{n!}\sum_{\varphi\in\mathcal{S}_n} z(\varphi),$$

in terms of A, B, and C. Motivate your answer. You may not assume that the diagonal entries of A and B are zero.

Solution: First prove Proposition 7.7 in the book, and then proceed as follows.
Without the assumption $a_{ii}=b_{ii}=0\ \forall i$, the average objective value of $QAP(A,B,C)$ is given by

$$\mu(A,B,C) = \frac{1}{n(n-1)}\sum_{i\neq j,k\neq l} a_{ij}b_{kl} + \frac{1}{n}\sum_{i,j=1}^n (c_{ij}+a_{ii}b_{jj}).$$

To prove this, we replace given A and B by the matrices $\hat{A}$ and $\hat{B}$ obtained by setting their respective diagonals to zero, and replace the given C by $\hat{C}=(c_{ij}+a_{ii}b_{jj})$.
Then $QAP(A,B,C)$ and $QAP(\hat{A},\hat{B},\hat{C})$ have the same objective function, since:

$$\begin{aligned} z(\varphi) &= \sum_{i=1}^n\sum_{j=1}^n a_{ij}b_{\varphi(i)\varphi(j)} + \sum_{i=1}^n c_{i\varphi(i)} \\ &= \sum_{i\neq j} a_{ij}b_{\varphi(i)\varphi(j)} + \sum_{i=1}^n \left(c_{i\varphi(i)} + a_{ii}b_{\varphi(i)\varphi(i)}\right), \end{aligned}$$

and the last expression is exactly the objective function of $QAP(\hat{A},\hat{B},\hat{C})$. Applying Proposition 7.7 to $QAP(\hat{A},\hat{B},\hat{C})$ now yields the required result.

(b) (5 points) Assume that the values $\varphi(i)$ $(i \in S)$ have been fixed for some given set $S \subset \{1, \ldots, n\}$. Derive the corresponding QAP problem, say $QAP(A', B', C')$, that corresponds to this node in the polytomic branching tree, i.e. give the matrices A', B' and C' in terms of A, B, C and S (and its complement).

Solution: $QAP(A, B, C)$:

$$\min_{\varphi \in \mathcal{S}_n} \sum_{i=1}^{n} \sum_{k=1}^{n} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i=1}^{n} c_{i\varphi(i)}$$

Split the sums in the objective over S and its complement, say $\bar{S}$:

$$\begin{aligned}
& \sum_{i=1}^{n} \sum_{k=1}^{n} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i=1}^{n} c_{i\varphi(i)} \\
= \; & \sum_{i \in \bar{S}} \sum_{k \in \bar{S}} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i \in \bar{S}}^{n} c_{i\varphi(i)} \\
& + \sum_{i \in S} \sum_{k \in S} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i \in S}^{n} c_{i\varphi(i)} \\
& + \sum_{i \in S} \sum_{k \in \bar{S}} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i \in \bar{S}} \sum_{k \in S} a_{ik} b_{\varphi(i)\varphi(k)}
\end{aligned}$$

Thus we get a new QAP, say $QAP(A', B', C')$ with $A' = (a'_{ij})$, $B' = (b'_{ij})$, $C' = (c'_{ij})$, and

$$c'_{ik} = \sum_{j \in S} (a_{ij} b_{kj} + a_{ji} b_{jk}) + c_{ik} \quad i, k \in \bar{S},$$

and

$$a'_{ij} = a_{ij}, \; b'_{ij} = b_{ij} \quad i, j \in \bar{S},$$

and a constant part

$$\text{const} = \sum_{i \in S} \sum_{k \in S} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i \in S}^{n} c_{i\varphi(i)}.$$

Thus we get:

$$\begin{aligned}
& \sum_{i=1}^{n} \sum_{k=1}^{n} a_{ik} b_{\varphi(i)\varphi(k)} + \sum_{i=1}^{n} c_{i\varphi(i)} \\
= \; & \text{const} + \sum_{i,k \in \bar{S}} a'_{ik} b'_{\varphi(i)\varphi(k)} + \sum_{i \in \bar{S}} c'_{i\varphi(i)}.
\end{aligned}$$

4. This question again concerns $QAP(A, B, C)$ as defined in the previous question. Also recall the

following notation:

$$\langle v, u \rangle^- = \min_{\varphi \in \mathcal{S}_n} \sum_{i=1}^{n} v_i u_{\varphi(i)} \qquad v, u \in \mathbb{R}^n.$$

(a) (4 points) Show that, if $v_1 \le v_2 \le \ldots \le v_n$ and $u_1 \ge u_2 \ge \ldots \ge u_n$, then

$$\langle v, u \rangle^- = \sum_{i=1}^{n} u_i v_i.$$

Solution: See the proof of Proposition 5.8 in the book.

(b) (4 points) It is given that, for any pair of symmetric matrices, say $A = A^\top$ and $B = B^\top$, it holds that $\mathrm{tr}(AB) \ge \langle \lambda, \mu \rangle^-$, where λ and μ are vectors containing the eigenvalues of A and B respectively. Use this fact to show that:

$$\langle \lambda, \mu \rangle^- = \min_{X \in \mathcal{O}_n} \mathrm{tr}(AXBX^T),$$

where $\mathcal{O}_n$ denotes the set of $n \times n$ orthogonal matrices.

Solution: It is given that

$$\mathrm{tr}(AB) \ge \langle \lambda, \mu \rangle^-.$$

Since this lower bound is valid for all symmetric A and B and only depends on their eigenvalues, we have

$$\mathrm{tr}(AXBX^T) \ge \langle \lambda, \mu \rangle^- \quad \forall X \in \mathcal{O}_n.$$

Thus

$$\min_{X \in \mathcal{O}_n} \mathrm{tr}(AXBX^T) \ge \langle \lambda, \mu \rangle^-.$$

We now prove the reverse inequality. Recall our notation for the spectral decompositions of A and B:

$$A = \sum_{i=1}^{n} \lambda_i p_i p_i^T, \quad B = \sum_{j=1}^{n} \mu_j q_j q_j^T,$$

where p_i and q_j orthornormal eigenvectors.

Also define the orthogonal matrices $P = [p_1\ p_2\ \ldots\ p_n]$ and $Q = [q_1\ q_2\ \ldots\ q_n]$. Note that P and Q diagonalise A and B resp.

Also assume w.l.o.g. that $\lambda_1 \le \lambda_2 \le \ldots \le \lambda_n$, and $\mu_1 \ge \mu_2 \ge \ldots \ge \mu_n$.

Now

$$\langle \lambda, \mu \rangle^- = \mathrm{tr}(P^T A P Q^T B Q) = \mathrm{tr}(A P Q^T B Q P^T) = \mathrm{tr}(AXBX^T),$$

where $X = PQ^T$. Noting that $X \in \mathcal{O}_n$ completes the proof.

(c) (2 points) Use any method of your choice to solve the following instance of $QAP(A, B, C)$:

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{pmatrix}, \ B = \begin{pmatrix} 8 & 12 & 16 \\ 12 & 18 & 24 \\ 16 & 24 & 32 \end{pmatrix},$$

and C is the 3×3 zero matrix. Give the optimal value as well as the optimal permutation.

Solution: $A = uu^T$ with $u = [1\ 2\ 3]^T$, $B = vv^T$ with $v = \sqrt{2}[2\ 3\ 4]'$. Therefore, by Proposition 8.9, $z^* = (< u, v >^-)^2 = 2(1 \cdot 4 + 2 \cdot 3 + 3 \cdot 2)^2 = 2 \cdot 16^2 = 512$, and $\varphi^* = (3, 2, 1)$.

5. Consider the following parameterized problem.

Bounded Degree Deletion
Instance: graph $G = (V, E)$ and numbers $d, k \in \mathbb{N}$.
Parameters: d and k
Question: is it possible to delete at most k vertices from G such that the resulting graph has maximum degree at most d?

(a) (4 points) Consider the following reduction rule. If there exists a vertex with degree larger than $d+k$, delete this vertex and reduce k by one. Show correctness of this reduction rule, i.e. show that the obtained instance is a yes-instance if and only if the original instance is a yes-instance.

Solution: If the vertex v with degree larger than $d + k$ is not deleted, then more than k of its neighbours would need to be deleted to get the degree of v down to d. This is not allowed since we may only delete k vertices. Hence, v needs to be deleted. This shows that if the orginal instance is a yes-instance, then the obtained instance is also a yes-instance. To show the converse, assume that the reduced instance is a yes-instance. Then there exists a solution to the reduced instance consisting of at most $k - 1$ vertices. Those vertices together with v give a solution for the original instance of at most k vertices.

(b) (4 points) Consider the following reduction rule. If there exists an edge connecting two vertices of degree at most d, delete this edge. Show correctness of this reduction rule.

Solution: Suppose edge $\{u, v\}$ is deleted. If the original instance is a yes-instance, then the reduced instance is clearly also a yes-instance: delete exactly the same vertices as in the original instance (we have only lowered the degrees). Now suppose that the reduced instance is a yes-instance. Then we can obtain a yes-instance of the original instance by deleting the same vertices. Adding the edge $\{u, v\}$ only increases the degrees of u and v and they can never get bigger than d since they had degree at most d in the original graph.

(c) (4 points) A third reduction rule deletes vertices of degree 0. Show that if none of these three reduction rules is applicable and there are more than $k + k(d + k) + k(d + k)^2$ vertices left, then the instance is a no-instance. Hence, the problem Bounded Degree Deletion has a polynomial kernel.

Solution: Suppose it is a yes-instance. Let U be the set of at most k vertices that are deleted in some solution. Each vertex of degree bigger than d must have at least one neighbour in U. Each other vertex has degree at least 1 (by the third reduction rule) and is hence adjacent to a vertex of degree bigger than d (by the second reduction rule). Hence, each vertex is in U, has a neighbour in U, or has a neighbour that has a neighbour in U.

Since there are at most k vertices in U and each vertex has degree at most $d+k$ (by the first reduction rule), the total number of vertices is at most

$$k + k(d+k) + k(d+k)^2.$$

6. A *tournament* is a directed graph $D = (V, A)$ with for each pair of vertices $u, v \in V$ exactly one of the arcs (u, v) and (v, u) in A. Consider the following parameterized problem.

FEEDBACK VERTEX SET IN TOURNAMENTS
Instance: tournament $D = (V, A)$ and number $k \in \mathbb{N}$.
Parameters: k
Question: is it possible to delete k vertices from D such that the resulting tournament is acyclic (i.e. has no directed cycle)?

(a) (4 points) Prove that a tournament is acyclic if and only if it has no directed triangle (i.e. a directed cycle of three vertices).

Solution: The "only if" direction is trivial. Now suppose that a tournament has some directed cycle but no directed triangle. Consider a smallest directed cycle $(u_1, u_2, u_3 \ldots, u_p, u_1)$ with $p \geq 4$. Then exactly one of the arcs (u_1, u_3) and (u_3, u_1) is in the tournament. If it is (u_1, u_3), then $(u_1, u_3, u_4, \ldots, u_p, u_1)$ is a smaller directed cycle. If it is (u_3, u_1), then (u_1, u_2, u_3, u_1) is a directed triangle. In both cases we have obtained a contradiction.

(b) (4 points) Design an FPT algorithm for FEEDBACK VERTEX SET IN TOURNAMENTS and analyze its running time.

Solution: Check if there is a directed triangle. If no such triangle exists then we are done. Otherwise, at least one of the three vertices from the triangle needs to be deleted. Hence, branch into three subproblems, in each subproblem deleting one of the three vertices and reducing k by one. The running time is $O(3^k n^3)$, with $n = |V|$ (if we use a brute-force algorithm for finding a directed triangle).

End of test