ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Mohsen Ghaffari, Angelika Steger, David Steurer, Emo Welzl, Peter Widmayer

Algorithms, Probability, and Computing — Special Assignment 2 — HS17

- The solution is due on **Tuesday, December 5, 2017** by **2:15 pm**. Please bring a print-out of your solution with you to the lecture. If you cannot attend (and please only then), you may alternatively send your solution as a PDF, likewise **until 2:15pm**, to `njerri@inf.ethz.ch`. We will send out a confirmation that we have received your file. Make sure you receive this confirmation within the day of the due date, otherwise complain timely.
- Please solve the exercises carefully and then write a nice and complete exposition of your solution using a computer, where we strongly recommend to use LaTeX. **We do not grade hand-written solutions**. A tutorial can be found at `http://www.cadmo.ethz.ch/education/thesis/latex`.
- For geometric drawings that can easily be integrated into LaTeX documents, we recommend the drawing editor IPE, retrievable at `http://ipe7.sourceforge.net/` in source code and as an executable for Windows.
- Keep in mind the following premises:
  - When writing in English, write short and simple sentences.
  - When writing a proof, write precise statements.

  The conclusion is, of course, that your solution should consist of sentences that are short, simple, and precise!
- This is a theory course, which means: if an exercise does not explicitly say “you do not need to prove your answer” or “justify intuitively”, then a formal proof is **always** required. You can of course refer in your solutions to the lecture notes and to the exercises, if a result you need has already been proved there.
- We would like to stress that the ETH Disciplinary Code applies to this special assignment as it constitutes part of your final grade. The only exception we make to the Code is that we encourage you to verbally discuss the tasks with your colleagues. It is strictly prohibited to share any (hand)written or electronic (partial) solutions with any of your colleagues. We are obligated to inform the Rector of any violations of the Code.
- There will be two special assignments this semester. Both of them will be graded and the average grade will contribute 20% to your final grade.
- As with all exercises, the material of the special assignments is relevant for the (midterm and final) exams.

Exercise 1 — 15 points
Random matchings

There is a close connection between counting algorithms and sampling algorithms. We have seen in the lecture how to count the number of perfect matchings in a graph (not very efficiently for general graphs) and here your task is to develop algorithms to sample a perfect matching uniformly at random. All the randomness you are allowed to use in this exercise is given by a stream of random bits and extracting one bit from the stream takes unit time.

Throughout, we let n denote the number of vertices in a graph. We assume access to a counting oracle that counts the number of perfect matchings in a graph in time $T(n)$.

(a) Given a positive integer N, how to efficiently sample a uniformly random number from the set $\{1, \dots, N\}$ by using the given stream of random bits? You should give a bound in big O notation on the number of random bits used in expectation.

(b) Show how to sample a uniformly random perfect matching in a given graph by using $O(n^2)$ calls to the counting oracle. You should use $O(n^2 \log n)$ random bits in expectation and your algorithm should run in expected time $O(T(n) \cdot \text{poly}(n))$.

(c) Show how to sample a uniformly random perfect matching in a given planar graph by using $O(n)$ calls to the counting oracle. You should use $O(n^2)$ random bits in expectation and your algorithm should run in expected time $O(nT(n))$. You can assume that $T(n) \in \Omega(n)$.

Exercise 2 — 15 points
Small exercises in linear programming

Let $\mathbf{c} \in \mathbb{R}^n$ be some fixed vector and let $k \in \{1, \dots, n\}$ be an integer. We denote by $\mathbf{1} \in \mathbb{R}^n$ and by $\mathbf{0} \in \mathbb{R}^n$ the vectors of all 1's and all 0s, respectively. Let further x be a vector of n real valued variables. For the questions (a)-(c) describe explicitly the set of all basic feasible solutions and find an optimal basic feasible solution to the linear program. Your answer should of course be phrased in terms of the parameters $\mathbf{c}$ and k.

(a) $\min \mathbf{c}^\mathsf{T}\mathsf{x}$ subject to $\mathbf{0} \leq \mathsf{x} \leq \mathbf{1}$.

(b) $\min \mathbf{c}^\mathsf{T}\mathsf{x}$ subject to $\mathbf{1}^\mathsf{T}\mathsf{x} = 1$ and $\mathsf{x} \geq \mathbf{0}$.

(c) $\max \mathbf{c}^\mathsf{T}\mathsf{x}$ subject to $\mathbf{1}^\mathsf{T}\mathsf{x} \leq k$ and $\mathbf{0} \leq \mathsf{x} \leq \mathbf{1}$.

Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a matrix and let $\mathbf{b} \in \mathbb{R}^m$ be a vector. Formulate the following optimization problem as a linear program.

(d) $\min ||\mathsf{x}||_1$ subject to $||\mathbf{A}\mathsf{x} - \mathbf{b}||_\infty \leq 1$.

Recall that $||\mathsf{x}||_1 = \sum_{i=1}^n |\mathsf{x}_i|$ and $||\mathsf{x}||_\infty = \max_{i=1,\dots,n} |\mathsf{x}_i|$.

Exercise 3 **15 points**

Farkas lemma for s-t paths

Let $D = (V, A)$ be a directed graph and let $s, t \in V$. To any vertex set $S \subseteq V$ we associate a *cut* $C(S) \subseteq A$ that consists of all arcs between S and $V \setminus S$. We say that $C(S)$ is an s-t *cut* if $s \in S$ and $t \notin S$. We say that $C(S)$ is a *strong* s-t *cut* if it is an s-t cut and if all edges in $C(S)$ are directed away from $V \setminus S$. See Figure 1 for an example.

In this exercise we will prove the following lemma and see that it is a special case of the Farkas lemma we have seen in the lecture. Informally, it says that there is a simple certificate for both proving and disproving the existence of a directed s-t path in D.

Lemma 1 (Farkas lemma for s-t-paths). *Exactly one of the following two statements holds for any directed graph $D = (V, A)$ and for any two vertices $s, t \in V$.*

i) *There exists a directed s-t path.*

ii) *There exists a strong s-t cut.*

For every vertex $v \in V$ let $\delta(v)^+ \subseteq A$ denote the arcs that are outgoing from v and let $\delta(v)^- \subseteq A$ denote the arcs that are incoming to v.

(a) Show that there is a directed s-t path in D if and only if the following system of equations and inequalities has a solution over the real valued variables $\{x_e \mid e \in A\}$.

$$\forall v \in V: \quad \sum_{e \in \delta(v)^+} x_e - \sum_{e \in \delta(v)^-} x_e = \begin{cases} 0 & \text{if } v \in V \setminus \{s, t\} \\ 1 & \text{if } v = s \\ -1 & \text{if } v = t \end{cases}$$

$$\forall e \in A: \quad x_e \geq 0$$

(b) Prove Lemma 1 by applying some version of Farkas lemma to the system in (a).

(c) Prove Lemma 1 directly without using (a) or Farkas lemma.

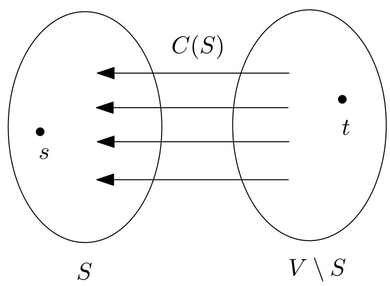

Figure 1: An illustrative example of a strong s-t cut. The cut $C(S)$ is a strong s-t cut because all edges in $C(S)$ are directed away from $V \setminus S$.