אלג' ברשתות - שיעורי בית 2 - לדצמבר 3 19.11.06

① יהי $G=(V,E)$ גרף מכוון.

2 מסלולים p_1,p_2 בין s ל t נקראים

<u>זרים בצמתים</u> אם $p_1 \cap p_2=\{s,t\}$.

הוכיחו שמקסימום מספר המסלולים הזרים בצמתים מ-s ל-t שווה למינימום מספר הצמתים שהסרתן מ-G מנתקת את כל המסלולים מ-s ל-t.

[האם ההוכחה שלכם עובדת גם עבור G לא מכוון? איזה מקרה יותר כללי?]

② זרימה מעגלית היא זרימה שבה מתקיים אילוץ שימור זרימה בכל צומת.

הוכיחו את משפט הזרימה המעגלית של הופמן:

תהי N רשת זרימה עם חסמים תחתונים $\{b(e)\}$.

קיימת זרימה מעגלית פיזיבילית ב-N אמ"מ לכל $A \subseteq V$ מתקיים: $c(\delta(V \setminus A)) \geq b(\delta(A))$

יתרה מזאת, אם כל הקיבולים והחסמים התחתונים שלמים, אזי קיימת זרימה מעגלית בשלמים אם קיימת זרימה מעגלית פיזיבלית (כלשהי).

③ הציעו אלגוריתם לפתרון הבעיה הבאה (transportation problem):

נתון גרף דו-צדדי $G=(A\cup B,E)$. לכל צומת $a\in A$ יש יכולת אספקה $s_a\in\mathbb{N}$. לכל צומת $b\in B$ יש דרישה $d_b\in\mathbb{N}$. מצא העברה $f: A\times B\to\mathbb{N}$ המקיימת:

1) לכל a: $\sum_b f(a,b)\le s_a$

2) לכל b: $\sum_a f(a,b)=d_b$

Question #4. (Bit scaling algorithm of Gabow [1985]). Let $k = \lceil \log_2 U \rceil$ in a network N. In the bit-scaling algorithm for maximum flow, each arc capacity $u(e)$ is represented as a k-bit binary number (leading zeros are added, if needed).

The problem P_i is a flow problem over the same network but with the capacity $u_i(e)$ defined as follows. If $u(e)$ is encoded by $b_{k-1}b_{k-2}\cdots b_0 \in \{0,1\}^k$. Then $u_i(e)$ is the binary number $b_{k-1}b_{k-2}\cdots b_{k-i}$ and $u_i(e) = \sum_{j=0}^{i-1} b_{k-i+j} \cdot 2^j$.

Let f_i^* denote a maximum flow in P_i.

The algorithm finds a max-flow in the network N by computing maximum flows f_i^* in P_i, starting with P_1. Note that P_k is the original problem.

After computing f_i^* (a max-flow in P_i), the algorithm computes f_{i+1}^* by using $2 \cdot f_i^*$ as an initial flow in P_{i+1}.

1. Show that $2 \cdot f_i^*$ is a feasible flow in P_{i+1}.
2. Show that the difference between the maximum flow value in P_{i+1} and the flow value $2 \cdot |f_i^*|$ is at most m.
3. Show that f_{i+1}^* can be computed in $O(mn)$ time using shortest augmenting paths if one starts with $2 \cdot f_i^*$.
4. Analyze the total running time of this algorithm.