# CS 214 – Algorithms and Complexity

Shanghai Jiaotong University, Fall 2016

- Handed out: Wednesday, 2016-12-07
- Due: Sunday, 2016-12-11
- Feedback: Tuesday, 2016-12-13
- Revision due: Friday, 2016-12-16

## 5 Push-Relabel, Karger's Algorithm, Stoer-Wagner

**Exercise 1.** [Push-Relabel Algorithm] Consider the following directed graph with edge capacities.


Suppose you run the push-relabel algorithm on it. Your selection policy is as follows: of all overflowing vertices (meaning inflow ¿ outflow), select the right-most one and apply an operation to it (either relabel or push). How many push/relabel steps does your algorithm make? Don't give an exact formula, give a $\Theta(\cdot)$ answer.

Now suppose you change your vertex selection policy and always select the left-most overflowing vertex. What is now the number of push/relabel steps performed by the algorithm?

**Exercise 2.** Review the analysis of Karger's algorithm.

1. Consider $C_n$, the undirected cycle on $n$ vertices. Characterize its global mincuts! How many global mincuts does it have?

2. Show that an undirected graph on $n$ vertices has at most $\binom{n}{2}$ global minimum cuts. **Hint.** Look at the analysis of Karger's algorithm and its success probability.

3. Give an undirected graph (possibly with edge weights) which has an exponential number of minimum $s$-$t-$cuts!

**Exercise 3.** Suppose we modify Stoer-Wagner algorithm as follows: In every execution of `mincut-phase`, we always select the *least* tightly connected vertex next, not the *most* tightly connected vertex. Also, of all runs of this subroutine, we select the cut of the *highest* value, not the smallest.

Show that this modified algorithm does not find a maximum cut in general. That is, find an undirected graph (possibly with edge weights) on which it fails to return a maximum cut. **Remark.** Don't simply give a graph $G$ and claim it serves your purpose. Describe what the modified Stoer-Wagner algorithm does on $G$.