CS498, Spring 2015

Liang Tao (ltao3)
Chen Zhang (czhang49)

Problem 1.

If there is a row or column of A that sums to a non-integer value, there is no way of rounding. Therefore, every row and column of A must sum to an integer. Define arrays B and C as follows [1]: $B[i,j] = \lfloor A[i,j] \rfloor$ and $C[i,j] = A[i,j] - B[i,j]$. Note that if C_R is a legal rounding for C, then, $C_R + B$ is a legal rounding for A. Now define the flow network F as follows: Define a source node S and a target node T. Define a node R_i for row i and an edge from S to R_i with capacity $\sum_j C[i,j]$. Define a node O_j for column j and an edge from O_j to T with capacity $\sum_i C[i,j]$. Finally, Define an edge from R_i to O_j with capacity 1 if $C[i,j] > 0$.

A maximum flow f^* for F is an integer flow since every edge capacity in F is an integer. We want to prove that $C^*[i,j] = f^*(R_i \to O_j)$ is a legal rounding of C (which yields a legal rounding for A as $C^* + B$). Notice that for a flow defined as $f(r_i \to c_j) = C[i,j]$, all edges leaving S and all entering T are saturated since the flow value is $\sum_{i,j} C[i,j]$. Therefore, f is a maximum flow. Moreover, the integral maximum flow f^* has the same value since it is a maximum flow. Because of saturation of all edges leaving S and entering T, C^* has the same sum as C in every pair of corresponding rows and columns. Furthermore, because of integrality, $C^*[i,j]$ is either 0 or 1. Hence, C^* is a legal rounding of C.

Using Orlin's algorithm for maximum flow, the algorithm runs in $O(mn(m+n))$ time since there are $O(m+n)$ vertices and $O(mn)$ edges.

References: [1]: CS473, Homework 9 solutions, Fall 2013.