

- (a) **Solution:** A network can be used to model this problem in the following way. Consider a network with source s and sink t , n "column" vertices c_1, c_2, \dots, c_n and m "row" vertices r_1, r_2, \dots, r_m . Connect the source s with each row vertex r_1, \dots, r_m , where both the lower bound and upper bound capacity along the edge between s and any row vertex r_i is the floor of the sum of that respective row in matrix M , e.g. $\lfloor \sum_j M_{ij} \rfloor$. There will also be an edge from each row vertex r_i to every column node c_i , where the capacity of an edge $r_i \rightarrow c_j$ has lower bound $\lfloor M_{ij} \rfloor$ and upper bound $\lceil M_{ij} \rceil$. Finally, an edge exists between every column vertex c_j to target t , with lower and upper bound capacity $\lfloor \sum_i M_{ij} \rfloor$. Constructing this network could be done in $O(mn)$ time by brute force (compute column and row sums, iterate the matrix to produce the edges). The algorithm would return *TRUE* if the max flow algorithm returns an integral flow and $\lfloor x \rfloor = x$ for all column and row sums x , and false if no integer flow exists in this network at all.

Correctness for this algorithm is contingent on the theorem that a max flow g exists if and only if a valid matrix rounding is possible. To prove this, first it should be shown that the existence of integral flow $g \Rightarrow$ matrix rounding. In the integral flow g , each edge $s \rightarrow r_1, \dots, r_m$ is fully saturated by its lower and upper bound capacity being equivalent. As such, the sum of all the edges leaving equals the sum of the row in matrix M corresponding to r_i by conservation constraint. Similarly, edges $c_1, \dots, c_n \rightarrow t$ are also all saturated, meaning the sum of the inbound edges also equals the sum of the corresponding column in matrix M . Therefore, this flow g would represent a valid rounding for the matrix M that satisfies the column and row sum constraint so long as the values of those sums were integers (s.t. $\lfloor x \rfloor = x$ for all column and row sums x). It can also be shown that the existence of a valid rounding of $M \Rightarrow$ integral flow g . This can be done by simply converting matrix M' (the rounding of M) into this corresponding flow g by setting the value of the flow along each $r_i \rightarrow c_j$ to be M'_{ij} . This constructed flow would satisfy conservation constraint at each row and column vertex by virtue of it satisfying the same constraints in the initial problem, namely it must not change the value of any row or column sum x .

Complexity of this algorithm can be sorted out by determining the number of vertices and the number of edges. The construction contains a source s , sink t , and a vertex set dominated by $m + n$. In other words, it contains a vertex for each row and each column. As stated above, it also contains an edge corresponding to each element in the input matrix M , thus it contains $m \times n$ edges. This means maximum flow runtime $O(VE)$ can be expressed in terms of the inputs as $O((m + n)mn)$. ■

- (b) **Solution:** In the case where some legal rounded matrix M' exists, it is the case that every column and row sum in M' is an integer, and also the cause that for an row or column sum x' in M' , $x = x'$, where x is the corresponding row or column sum in M . As a result, this means every row and column sum in M is also an integer.

In the case where every row and column sum x in M is an integer, it must be the case that a legal rounding is possible in the following way. Chiefly, there exists no bottleneck column vertex c' in the network G' s.t. a fully saturated t is not attainable. As a result, so long as the value of the inbound flow at $t \in G$ is the sum of all the column sums (in

other words, the secondary condition $\lfloor x \rfloor = x$ for all rows and columns is true), then this network's guaranteed-to-exist flow is a solution for a legal rounding. If all the row and column sums in M are integers, this clause of $\lfloor x \rfloor = x$ is true, and thus a legal rounding is always possible. ■