Assignment 19

Alex Clemmer
Student number: u0458675

1

We have some set of currencies C and some set of edges between them E—note that edges are NOT the raw exchange rates. To compute some edge $\in E$, we simply subtract the exchange rate corresponding to that edge by the complement. For example: the weight of $e_{ij} \in= r_i - r_j$, and the weight of $w_{ji} = r_j - r_i$. In this way, one will be negative and the other will be positive. This lends itself well to Bellman-Ford.

The trick here is that each edge has some cost, and some are positive, while others are negative. This means that the "best" path ends up being the one with the smallest net distance, or the "shortest" path, which is classic Bellman-Ford terrain.

2

Knowing that there is a positive cycle means that we can just exchange in sequence through that loop, adding some small delta of money each time until the loop disappears.

3

Adapt the G^R algorithm we learned. First, reverse the polarity and apply a depth-first search, saving the pre- and post-numbers as you go. Reverse the polarity again, and depth-first search again based on the post numbers, noting the weights as you go; if the sink has a net-path that's positive, then you have found a strongly-connected component that has a positive net weight. Simple.