Why negative edge weights?

Vassos Hadzilacos

Suppose that the exchange rate to convert Canadian dollars to British pounds is r, and the exchange rate to convert British pounds to Canadian dollars in r'. Presumably $r' = 1/r$: If $r' > 1/r$, then starting with a positive amount of Canadian dollars you can make an arbitrary amount of money by converting them to British pounds and then back to Canadian dollars repeatedly. If $r' < 1/r$, you can achieve the same goal starting with a positive amount of British pounds.

The world currency market has many currencies, so you can convert Canadian dollars to British pounds in many different ways. You can do so directly, or you can first convert your Canadian dollars to Chinese yuan, then the yuan to euros, then the euros to Indian rupees, then the rupees to Japanese yen, the yen to Swedish krona, and the krona to British pounds.

Imagine now a weighted directed graph where the nodes are currencies, and there is an edge from currency c to currency c' with weight r if and only if we can exchange one unit of c to r units of c'. Note that r is a positive number, possibly a fraction. A particular way of converting currency c to currency c' corresponds to a path $c_1, c_2, \ldots, c_k$ on this graph, where $c_1 = c$ and $c_k = c'$. The effective exchange rate from c to c' along this path is the product $\Pi_{i=1}^{k-1} r_i = r_1 \cdot r_2 \cdot \ldots \cdot r_{k-1}$, where r_i is the exchange rate for converting currency c_i to currency c_{i+1}. To maximize the amount of money we receive for converting c to c', we want to find a path that maximizes the product of the exchange rates on the edges of the path.

This is beginning to look like a shortest path problem, except for two things: First, shortest path is a minimization, not maximization, problem. But this is easy to fix: Maximizing $\Pi_{i=1}^{k-1} r_i$ is the same as minimizing $\Pi_{i=1}^{k-1} 1/r_i$. Second, in shortest paths we want to minimize the ***sum*** of the weights of edges on a path, not their ***product***. But we can turn a product into a sum by taking logarithms: Minimizing $\Pi_{i=1}^{k-1} 1/r_i$ is the same as minimizing its logarithm, and $\log\bigl(\Pi_{i=1}^{k-1} 1/r_i\bigr) = \Sigma_{i=1}^{k-1} \log 1/r_i$. Thus, a path that maximizes $\Pi_{i=1}^{k-1} r_i$ in the currency graph is a path that minimizes $\Sigma_{i=1}^{k-1} \log 1/r_i$. And so, to find the best way of converting currency c to currency c', we look for a shortest $c \to c'$ path in the currency graph where edge (a, b) has weight $\log 1/r$, where r is the exchange rate of currency a to currency b.

Note, however, that

$$\log 1/r \begin{cases} < 0, & \text{if } r > 1 \\ = 0, & \text{if } r = 1 \\ > 0, & \text{if } 0 < r < 1. \end{cases}$$

This means that the graph with nodes corresponding to the currencies and edge weights corresponding to the logarithms of the inverses of the exchange rates may contain edges with weight that is negative, zero, or positive. Thus, to solve a shortest path problem in this graph we must use an algorithm that works if edges can have negative weights: Dijkstra's algorithm is not appropriate for this application, but Bellman-Ford, Floyd-Warshall, and Johnson's algorithms are.

What about negative-weight cycles? A negative-weight cycle corresponds to a sequence of currency exchanges in which starting with x units of some currency we wind up with more than x units of the same currency! Traders call this an ***arbitrage opportunity***. It is a way to make money without engaging in any productive activity — just by moving money around. In the world of trade, graphs such as the one described above contain nodes for all exchangeable goods, not just currencies, and there are actually people who look for negative weight cycles in such graphs — arbitrage opportunities that they hope to exploit.

Of course in real life exchange rates are dynamic and negative-weight cycles that arise last only for very short periods of time. If you are in the middle of a long sequence of trades hoping to exploit what now appears to be an arbitrage opportunity, exchange rates may change while you are executing your trades and you may end up losing money. There is no free lunch: unproductive activities that supposedly generate money can blow up and cause a great deal of damage — and, unfortunately, not always to the individuals or firms involved. But we have now entered the more important realm of ethics, and shortest path algorithms have nothing to teach us about that.