Executive Summary of a Fair Multiflow Problem

Romaji Milton Amulo

April 15, 2021

Road networks. The conveyor belts in factories. Supply chains. There are many kinds of networks, used for transporting items across them. While networks in practice differ from each other, it is useful to simplify to nodes, which represent key stopping or decision points, and arcs, which represent the paths between nodes. Each arc also has a capacity, representing how many items they can carry over a unit of time or can fit in them in transport. This allows for a unified model of networks as a whole, called a "network" in Graph Theory.

In these networks, the movement of items is modeled by "flows". Flows have contents, which are transported from their source to their destination, across the arcs of the network. Flows have two constraints: The contents that enters any node (besides the source or destination) must exit that node and no arc can have more contents on it than its capacity. For many network problems, the goal is transporting as many items as possible or a given amount as fast as possible. For the equivalent flow problem, maximizing the content arriving at the destination, many quick algorithms exist.

However, for problems with more than one flow, such as assembling a car from parts in different locations, good algorithms don't exist. Our paper is about a problem where multiple flows must share a network, where the flow values must be in a given proportion. This proportion restriction on flows arises naturally in situations where an excess of one flow does not help, such as in factories, where an excess of one part does not aid in assembly.

To help solve that problem, we created new ideas: load functions, the utilization, and unrestricted flows ("uflows"). A load function is the most general "load" on a network, usually made by adding the effect of several flows. The utilization of a load function on a network indicates the "fullness" of the network or the required time for the load to move across it. Finally, an unrestricted flow ("uflow" for short), is a flow without the capacity constraint. By using uflows and the utilization, the problem of maximizing amount transported under a given proportion between the flows becomes a problem about reducing the utilization, with the amount for each flow given.

Our algorithm uses this change in perspective to divide the multiple flow problem into sub-problems with a load function and a single uflow each. For each sub-problem, the goal is to change its uflow to reduce the combined utilization with its load function, while preserving the size of the uflow. Fortunately, we found good algorithms for this sub-problem. Additionally, because our algorithm works by incrementally improving an initial solution, it is an "any time" algorithm. This means that our algorithm is more useful than generic techniques for these fixed ratio problems when an answer is needed quickly after a change in the network.