

## Abstract

In the Unsplittable Flow on a Path problem (UFP) we are given a path with non-negative edge capacities and a set of tasks, each one characterized by a subpath, a demand, and a profit. Our goal is to select a subset of tasks of maximum total profit so that the total demand of the selected tasks on each edge does not exceed the respective edge capacity. UFP naturally captures several applications in bandwidth allocation, job scheduling, and caching. Following a sequence of improvements, the current best polynomial time approximation factor for UFP is  $2 + \epsilon$  (Anagnostopoulos et al. SODA'14). UFP also admits a QPTAS (Bansal et al. STOC'06, Batra et al. SODA'15), and finding a PTAS is considered a challenging open problem. In this paper we make progress in the direction of the mentioned open problem. Informally, we introduce a technique to obtain real PTASs from PTASs with *resource augmentation* where edge capacities can be violated by a  $1 + \epsilon$  factor. While unfortunately we do not have a resource-augmentation PTAS for the general case of UFP, for many relevant special cases we have such an algorithm or we provide one in this paper. For example, our approach leads to a PTAS for the rooted case of UFP, where all tasks share a common edge. This is one of the simplest natural restrictions of UFP where the best-known approximation was  $2 + \epsilon$  (like for the general case). At a high level, our technique is to sacrifice a few tasks in the optimal solution (with a small loss of profit) in order to create a sufficient amount of *slack capacity* on each edge. This slack turns out to be large enough to substitute the additional capacity we would gain from resource augmentation. Crucial for our approach is that we obtain slack from tasks with relatively small and relatively large demand simultaneously. In all prior polynomial time approximation algorithms the sacrificed tasks came from only one of these two groups.