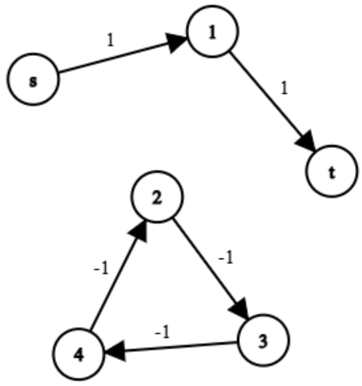
The Shortest Path Problem

Subtour detection



Interestingly, our graph above is an example for which the formulation (16-21) in the assignment produces an optimal solution that is not a path from source to sink, while at the same time respecting the SEC in (22). We therefore answered the following questions:

- Show an example of a graph in which the above formulation produces an optimal solution that is not a path from source to sink.
- Show an example of a graph in which constraint (22) does not prevent a subtour from forming.

First, it is quite obvious that all the constraints in 16-21 are met but that there is still a sub-tour due to the negative costs. This happens because even the target of going from source (s) to sink (t) is fulfilled, is more optimal to complete the sub-tour, adding a negative cost to the objective function and thus obtaining a better value than if only the desired path from s to t was selected.
Regarding the second point, we can check if the SEC prevents the sub-tour by iterating through all possible subsets S. The naive formulation given by the enunciate has a subtle error that makes it unsuitable to detect sub-tours. It states that subset S must contain s and subset T must contain t. Then, if there is any path between source and sink, the constraint is unable to detect subtours as there will be crossing arcs from S to T independently that a subtour is present or not. For each S containing source, we check if there is a crossing arc to the corresponding T. As it turns out, there are always crossing arcs due the path *s-1-t* and the condition holds, not achieving the desired Subtour detection.

Branch and Cut with SEC to prevent subtours

The branch-and-cut algorithm to solve the required formulation can be seen in the attached ipynb file *Project_fractional.ipynb* and *Project_integer.ipynb*. In short, the first one corresponds to the code handling integer solutions while the second one also works for fractional costs but still correctly identifies subtours by using the SEC described below. We tested our procedure on multiple different graphs and found no instance where our procedure does not succeed.

Proof of applied SEC

The applied SEC is the following:

$$\sum_{i \in S} \sum_{j \in V \setminus S} x^*_{ij} \geq \sum_{k \in V} x^*_{lk} \quad \forall S \subsetneq V, l \in S$$

This is done by solving the min-cut problem on a support graph $G^* = (V, A^*)$, defined as a graph in which we only keep the edges associated with positive variables x^*. This allows us not considering isolated vertices from the solution when checking subtour elimination.

To prove how our approach for subtour detection works , we coul try to show that $\sum_{i\in S}\sum_{j\in V\setminus S} x^*_{ij} < \sum_{k\in V} x^*_{tk}$ contradicts the given SEC by the problem $\sum_{i\in T}\sum_{j\in T} x^*_{ij} > |T|-1$ as originally intended. However, knowing that $\sum_{i\in T}\sum_{j\in T} x^*_{ij} > |T|-1$ is a valid SEC, it is sufficient to prove that $\sum_{i\in S}\sum_{j\in V\setminus S} x^*_{ij} < \sum_{k\in V} x^*_{tk} \quad \forall S \subsetneq V, l\in S$ contradicts any other valid SEC for the Shortest Path problem.

Put differently, it is sufficient to show that the applied SEC formulation is capable of fully preventing subtours with integer solutions.
We know from the initial problem constraints that:

$$1 \geq \sum_{k\in V} x^*_{lk} \quad \forall l \in S$$

As the maximum flow exiting a vertex can at most be 1. It is thus sufficient to check that

$$\sum_{i\in S}\sum_{j\in V\setminus S} x^*_{ij} \geq 1 \quad \forall S \subsetneq V$$

We recall this expression from somewhere, specifically from the original TSP problem. This expression was a valid SEC for the problem but now we should understand how it fits our problem on the support graph. The support graph only admits positive arcs, so isolated vertices are not present in that formulation. Now if we take *s* as source and any other vertex as sink, we can check for integer solutions if the number of crossing arcs between the partitions is more than 1. If this is not the case, we are having a subtour.
Thus, this formulation presents a valid SEC for the support graph and is able to recognize the presence of a subtour. Then the SEC presented for the problem is also a valid SEC and can be used for this purpose.