Max-flow min-cut

Author: Vincent Delecroix

Let G be an oriented graph with non-negative weights $\phi : E \to \mathbb{R}^+$ on the edges. The weights have to be thought of a capacity (ie a maximum quantity of fluid) on each edge. Now let u and v be two vertices. We are interested in *flows* in this graph from u to v that is a function $f : E(G) \to \mathbb{R}_+$ so that $f(e) \leq \phi(e)$ and for each vertex w we have

$$\sum_{e \text{ incoming at } w} f(e) = \sum_{e \text{ outgoing at w}} f(e)$$

Write an Linear Program that solves the max-flow (the input is the weighted graph G and two vertices u and v).

Note that linear programing is not the most efficient way to solve the max-flow problem. You might want to have in a book about "combinatorial optimization". Could you find an other implementation available in Sage?

Now let G be a (non-oriented) graph. A *matching* in G is a set of edges $F \subset E(G)$ so that no pair of edges in F shares a vertex. Write an Integral Linear Program that find the matching with maximal cardinality on a graph.

Do you know the algorithmic complexity of the maximum matching problem?

We will now consider the maximum matching on bipartite graph. Given a bipartite graph G with bi-partition $V = V_1 \cup V_2$ we consider the weighted oriented graph G' whose vertex set is $V \cup \{s, e\}$ and the following edges and weights:

- for each $v \in V_1$ an edge $s \to v$ with weight $+\infty$
- for each $e = \{v_1, v_2\}$ in G where $v_i \in V_i$ an edge $v_1 \to v_2$ with weight 1
- for each $v \in V_2$ an edge $v \to e$ with weight $+\infty$

Show that the maximum matching problem on G can be solved by considering the max-flow on G' from s to e. Then implement this for bipartite graphs

How does the performance compare with your integer program?

(*hint*: for comparing performance you might want to have a look at the `%time` and `%timeit` IPython magic command)

Now, use the bipartite version of your program to solve the following weighted version of the Hall mariage problem. Given M be a $n \times n$ matrix with positive entries, compute

$$\max_{\sigma \in S_n} \sum M_{i,\sigma(i)}$$