A New Formulation of the Maximum Concurrent Flow Problem and Proof of the Max-Concurrent-Flow/Max-Elongation Duality Theorem

David W. Matula

March 10, 1986

For the graph $G = \langle V,E \rangle$, let $C:E \to R^+$ denote a capacity function on the edges of G and $D:V \times V \to R^+$, with $D(i,i) = 0$, $D(i,j) = D(j,i)$, denote a concurrent demand function on the set of all vertex pairs of G.

Maximum Concurrent Flow Problem. [1],[2] Denote by P_e the set of all paths containing the edge e, by P_{ij} the set of all paths between distinct end vertices i, j, and by P the set of all non trivial paths in G. A concurrent flow of throughput z and capacity utilization u in G is a function $f:P \to R^+$ such that:

$$\sum_{p \in P_{ij}} f(p) = D(i,j)z \qquad \text{for all distinct } i,j \in V, \tag{1}$$

$$\sum_{p \in P_e} f(p) \leq C(e)u \qquad \text{for all } e \in E. \tag{2}$$

The maximum concurrent flow problem (MCFP) refers to the determination of the maximum throughput $\hat{z}$ and some maximum concurrent flow function $\hat{f}$, assuming unit capacity utilization. The minimum capacity utilization problem (MCUP) refers tp the determination of a minimum utilization u^* and concurrent flow function f^*, assuming unit throughput. □

Note that the solutions of the MCFP and MCUP differ only by scaling the resulting concurrent flow function, with $\hat{z} = 1/u^*$. We generically refer to the maximum concurrent flow problem to include both the MCFP and MCUP, with the application and/or algorithm dictating the preferred formulation.

Edge-Path Linear Program Formulation of the MCFP. In this case equations (1) and (2) with $u = 1$ are taken as the constraints of the linear program, z and $f(p)$ for all $p \in P$ are taken as non negative variables, and the objective is to maximize z. Equivalently for the MCUP take $z = 1$, and the objective to minimize u. □

Our new formulation of the MCFP introduces the notion of demand diversion for the vertex pair $\{i,j\}$ through the "bridging" vertex $k \neq i,j$. Specifically, demand between i,j is reduced by some amount, and demand between both pairs i,k and k,j is increased by the same amount. As vertex triples i,j,k are considered for all unordered pairs $i,j \in V$ and $k \in V$, $k \neq i,j$, we term this formulation the node-triple formulation.

Node-Triple Linear Program Formulation for the MCFP. Let $f_{ij}^{k} \geq 0$ denote the demand scheduled between i and j that is diverted through k. The net demand between i and j for throughput z is then given by the initial demand $D(i,j)z$ less the amount diverted off i,j to k for all k (i.e. $\sum_k f_{ij}^{k}$) plus the amount diverted onto i,j for all k (i.e. $\sum_k (f_{kj}^{i} + f_{ki}^{j})$), which must be less than the utilizable capacity $C(i,j)u$. Notationally, let $C(i,j) = C(e)$ for edge $e = (i,j)$, and $C(i,j) = 0$ when (i,j) is not an edge. Thus our linear programming formulation (unit throughput/minimize utilization form) has as variables u and f_{ij}^{k}, with the objective to minimize u subject to

$$D(i,j) - \sum_k f_{ij}^{k} + \sum_k f_{kj}^{i} + \sum_k f_{ki}^{j} \leq C(i,j)u. \qquad (3)$$ □

Observe that the edge-path formulation of the MCFP may have more than $2^{|V|}$ variables. However, since the node-triple formulation has $O(|V|^3)$ variables and $O(|V|^2)$ constraints, the MCFP is solvable in polynomially bounded time.

The problem we shall show dual to the MCFP is the following.

Distance Elongation Problem. For any distance function $d: V \times V \to R^+$, where notationally, $d(e) = d(i,j)$ for e the edge (i,j), let the elongation ℓ be the total distance of a unit of throughput, $\ell = \sum_{\{i,j\}} D(i,j)d(i,j)$, where the distance function is normalized so that $\sum_e C(e)d(e) = 1$. The distance elongation problem (DEP) refers to the determination of the maximum elongation $\hat{\ell}$ and some distance function $\hat{d}$ satisfying $\hat{\ell} = \sum_{\{i,j\}} D(i,j)\hat{d}(i,j)$ with $\sum_e C(E)\hat{d}(e) = 1$. □

The DEP may be given the following interpretations for application.

i) **Tariffs:** Letting d(e) be the tariff on edge e, with $\sum_e C(e)d(e) = 1$ representing a regulatory pricing constraint on line (edge) charges, how does a utility allocate tariffs to the different edges so as to maximize the income $\hat{\ell}$ per unit throughput?

ii) **Delay times:** Letting d(e) be the delay time on edge e, what is the worst case ratio of total delay per unit throughput to observed delay on edge capacity (i.e. what is $\max_d \sum_{\{i,j\}} D(i,j)d(i,j) \Big/ \sum_e C(e)d(e)$?)?

The DEP has a straightforward linear programming formulation with the distance function characterized by the triangle inequality rule.

Triangle Inequality Linear Program Formulation of the DEP. Let ℓ and $d_{ij} \geq 0$ for $i,j \in V$ be variables of the L.P. with $d_{ii} = 0$ for all i and $d_{ij} = d_{ji}$ for all $i,j \in V$. The objective is then to maximize ℓ subject to

$$d_{ij} + d_{jk} - d_{ik} \geq 0 \quad \text{for all } i,j,k \in V \tag{4}$$

$$\sum_e C(E)d(e) = 1. \tag{5}$$

□

Theorem(Max-Concurrent-Flow/Max-Elongation):

The maximum throughput $\hat{z}$ of concurrent flow is equal to the reciprocal of the maximum elongation $\hat{\ell}$ of distance on G.

Proof: Note from (3) that the node triple L.P. formulation of the MCFP may be written as:

minimize u subject to

$$C(i,j)u + \sum_k f_{ij}^k - \sum_k f_{kj}^i - \sum_k f_{ki}^j \geq D(i,j) \quad \text{for all } i,j \in V. \tag{6}$$

The standard linear programming dual is then

$$\text{maximize} \sum_{\{i,j\}} D(i,j)y_{ij} \tag{7}$$

subject to

$$\sum_{\{i,j\}} C(i,j)y_{ij} \leq 1, \tag{8}$$

$$y_{ij} - y_{kj} - y_{ki} \leq 0 \quad \text{for all } i,j,k \in V. \tag{9}$$

Clearly the maximum here will have $\sum_{\{i,j\}} C(i,j)y_{ij} = 1$. Note from (9) that y is a distance function on $V \times V$. The dual, as given by (7), (8), and (9), is then observed to be the DEP, with $\hat{\ell} = \max_y \sum D(i,j)y_{ij}$ the maximum objective value. Thus $\hat{\ell} = u^* = 1/z$. □

References

[1] Matula, D. W., "Concurrent Flow and Concurrent Connectivity in Graphs," in Graph Theory and its Applications to Algorithms and Computer Science, Alavi, Y., et al., ed., Wiley, New York, 1985, pp. 543-559.

[2] Matula, D. W. and Shahrokhi, F., Graph Partitioning via Sparse Cuts and Maximum Concurrent Flow, submitted for publication.