Optimization Problem

February 19, 2014

1 Problem Statement

Our problem is to optimize metric, M, in the following scenario, we look at a few choices for M below. Consider a data-center network abstracted as a complete bipartite graph with left vertices denoted by the set A and right vertices denoted by B, each having n elements. Flows arrive based on a schedule S where flows are characterized by start time, size (s_f), and $(a_i, b_i), i \in 1 \dots n$, as the source and destination nodes, determine the schedule for flow processing that minimizes the metric M.

Examples of metric M can be:

1. Average flow completion time (FCT)
2. Average slowdown (Normalize the FCT by the time it would take if it was the only flow in the network)

2 Tradeoff between algorithms

2.1 SRPT* - Extension of SRPT to our case

SRPT scheduling is provably optimal for minimizing completion times on a single server. A natural extension of SRPT to this scenario is as follows. Consider a flow of size $size_f$ that arrives at a destined to b.

if both ports are idle; schedule f

else if only one of them is busy but s_f is smaller than the flow on the busy port, schedule f by pre-empting the other flow and check if any other flow can now be scheduled.

else if both ports are busy and size(f) is smaller, then schedule f and preempt the other two flows; see if more flows can be scheduled.

2.2 Maximal Matching (MAXMAT)

A maximal matching results in highest utilization possible.

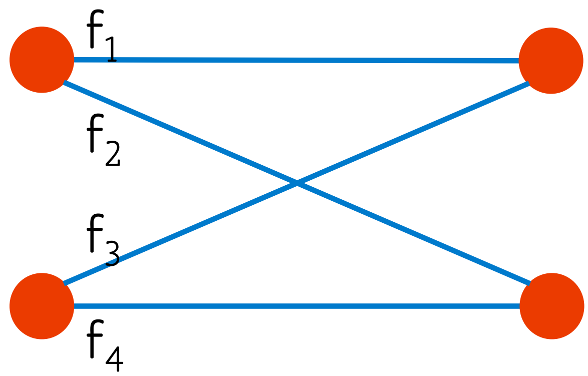

Figure 1: A simple example

2.3 SRPT* vs MAXMAT

One can easily construct examples in which one is better than the other. Consider Figure 1 for a simple example. Assume at full line rate; a unit flow takes 1 unit of time to finish.

2.3.1 SRPT* is better

$t = 0$: f_1, f_2, f_4 come. $s_{f_1} = s_{f_4} = 10$, $s_{f_2} = 1$
If SRPT* is used, f_2 gets scheduled first and then f_1 and f_4:
FCTs: $f_1 = 11$, $f_2 = 1$, $f_4 = 11$; Average = 23/3

If MAXMAT is used: f_1 and f_4 get scheduled first and then f_2:
FCTs: $f_1 = 10$, $f_4 = 10$, $f_2 = 11$; Average = 31/3

2.3.2 MAXMAT is better

$t = 0$: f_1, f_4 come. $s_{f_1} = s_{f_4} = 2 + e$ (e is very small)
$t = 1$: f_2 comes in with $s_{f_2} = 1$
$t = 2$: f_3 comes in with $s_{f_3} = 1$
If SRPT* is used, f_2 preempts f_1 and f_4, then f_3 finishes and then f_1 and f_4
FCTs: $f_1 = f_4 = 4 + e$, $f_2 = 1$, $f_3 = 1$; Average $= (10 + e)/3$
If MAXMAT is used used: f_2 and f_3 dont preempt f_1 and f_4
FCTs: $f_1 = f_4 = 2 + e$, $f_2 = 2 + e$, $f_3 = 1 + e$; Average $= (7 + e)/3$

3 Optimality of Weighted Maximal Matching

We consider the algorithm which schedules flows as per the weighted maximal matching where the weight of a flow, $w(f) = \frac{1}{s_f}$, i.e., reciprocal of flow's size. The weight of a matching is denoted by the sum of weight of the flows.

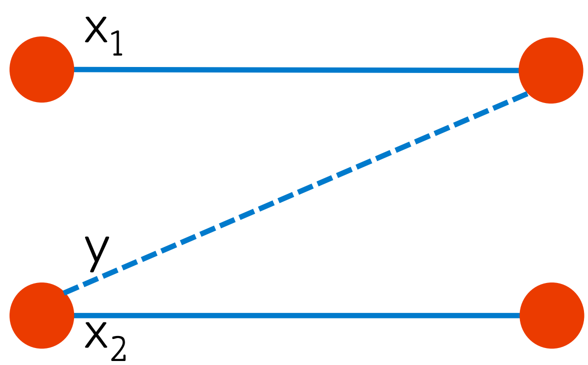

Figure 2: Optimality property of weighted maximal matching

We are able to prove the following optimality in a local setting. Consider the scenario in Figure 2. Let $x_1 < x_2$ without loss of generality. We consider the following two schedules:

- A: Schedule x_1 and x_2 first. When x_1 finishes, check if y would now pre-empt x_2.
- B: Schedule y first and then x_1 and x_2

The sum of FCTs in A is $x_1 + (y + x_2) + (x_1 + y)$ if y pre-empts x_2 once x_1 finishes, which happens if $x_2 - x_1 > y$; otherwise it is $x_1 + x_2 + (x_2 + y)$.

The sum of FCTs in B is $y + (y + x_1) + (y + x_2)$.

We show the following: If $A \succ B$ then $w(A) > w(B)$. (The reverse side is straight forward which actually establishes equivalence).

Case 1: $x_2 - x_1 > y$: $A \succ B \equiv x_1 < y$. Then $w(A) = \frac{1}{x_1} + \frac{1}{x_2} > \frac{1}{x_1} > \frac{1}{y} = w(B)$

Case 2: $x_2 - x_1 < y$: $A \succ B \equiv x_2 < 2y$. Then $w(A) = \frac{1}{x_1} + \frac{1}{x_2} > 2\frac{1}{x_2} > 2\frac{1}{2y} = w(B)$

H.P.