The problem $1|| \sum w_j C_j$ and its variants

Due date: March 27, 2009
Do at least six of the following problems.


2. Consider single-machine min-sum chapter, problem 4.13, and determine whether or not there is a preference order on sequences for $1||L_{\text{max}}$ (depending on the precise definition of a preference order).

3. Give an efficient procedure that, given $(C_1, \ldots, C_n)$ either finds a subset $S$ such that
   \[ \sum_{j \in S} p_j C_j < f(S) \]
   (where $f(S) = (p^2(S) + p(S)^2)/2$) or else correctly concludes that no such set exists.

4. Give a 3-approximation algorithm for $1|r_j, \text{ prec}||\sum w_j C_j$ by considering the linear relaxation: minimize $\sum w_j C_j$ subject to $\sum_{j \in S} p_j C_j \geq f(S)$ for each $S \subset N$, $\sum_{j \in N} p_j C_j \geq f(N)$, $C_k \geq C_j + p_k$ for each $j \prec k$, and $C_j \geq p_j + r_j$ for each $j \in N$.

5. Consider a (non-preemptive) time-indexed formulation for $1|\text{ prec}||\sum w_j C_j$ ($x_{jt}$ variables). The interpretation for these variables is that $x_{jt} = 1$ if job $j$ completes at time $t$.
   (a) Give a precise integer programming formulation in these variables. You should include constraints that (1) express that for each time $t = 1, \ldots, p(N)$, there is at most one job that is being processed in the time interval $[t-1, t)$; (2) for each job $j$ it is scheduled to complete at one time in the interval $[p_j, p(N))$ - that is, it is also not scheduled to complete earlier than $p_j$; (3) and constraints that enforce the precedence constraints - here you are suggested to use the following way to derive very strong constraints - for each time $t$, and each precedence constrained jobs $j \rightarrow k$, if job $k$ completes by time $t$ (that is, it completes at some time among $p_k, \ldots, t$) then job $j$ completes by time $t - p_k$ (that is, it completes at some time among $p_j, \ldots, t - p_k$).
   (b) Consider the following $\alpha$-point algorithm based on this formulation. Given $x_{jt}$, a feasible solution for this formulation, let the $\alpha$-point of job $j$ be the minimum time $t$ such that $\sum_{s=1}^{t} x_{js} \geq \alpha$. Then, schedule the jobs in the order in which they reach their $\alpha$-points. Show that the schedule produced is feasible for each $\alpha \in (0, 1]$.
(c) Prove that the 1/2-point algorithm finds a schedule of objective function value within a factor of 4 of optimal.

(d) (*) Show that by choosing $\alpha$ at random from the correct distribution, the randomized $\alpha$-point algorithm finds a schedule for which the expectation of the objective function value is within a factor of 2 of optimal.

6. This problem is to show how to use the max-flow min-cut theorem to compute the initial set $S$ that maximizes $w(S)/p(S)$. First recognize that it is sufficient to give an algorithm to decide if there exists a set $S$ for which $w(S)/p(S) \geq \lambda$ (or equivalently, $w(S) - \lambda p(S) \geq 0$). To solve this decision problem, you want an $s, t$ min-cut problem problem in which you take the precedence graph, add a source $s$ (and an edge to each node in the precedence graph) and a sink $t$ (with an edge from each node in the precedence graph), impose (infinite) capacities (appropriately) to ensure that all finite capacity cuts correspond to initial sets, and other capacities equal to $w_j$ and $\lambda p_j$ (appropriately).

7. Suppose that one takes an input to $1|\text{prec}| \sum w_j C_j$, and creates a new one by interchanging the weights and processings times (that is, $w'_j = p_j$ and $p'_j = w_j$, $j = 1, \ldots, n$), and reversing the direction of the precedence constraints (that is, $j$ precedes $i$ in the new precedence relation if and only if $i$ precedes $j$ in the original); use a 2-dimensional Gantt chart to explain the relationship between these two inputs.

8. (*) Suppose that you are given, for any value of $n$, a bipartite graph $G = (L, R, E)$, where $|L| = |R| = n$, each vertex in $R$ has degree $n^{3/4}$, each vertex in $L$ has degree at most $3n^{3/4}$, and each subset of $n^{3/4}$ vertices in $R$ has at least $n - n^{3/4}$ distinct neighbors in $L$. Use this graph to define inputs for the linear ordering formulation of $1|\text{prec}| \sum w_j C_j$ ($\delta_{jk}$ variables) with the following properties:

- There are $2n$ jobs, and for each $j = 1, \ldots, n$, $(p_j, w_j)$ is either (0,1) or (1,0);
- There is a feasible fractional solution (in fact, one in which each variable is either 0,1, or 1/2) of objective function value at most $(n^2 + n^{7/4})/2$;
- For any feasible schedule, there are at least $n - n^{3/4}$ jobs of weight 1 that finish later than at least $n - n^{3/4}$ jobs of processing requirement 1.

Show that the linear ordering formulation has inputs for which the integrality gap, the ratio between the values of the optimal integer and fractional solutions, is asymptotically close to 2. (You may simply assume that the claimed bipartite graphs exist; however, if you have seen the probabilistic method before, you might also try to prove the existence of such graphs.)
9. (**) Give a 2-approximation algorithm for $1|r_j, prec, pmtn| \sum w_jC_j$.

(a) It will be convenient to assume that for all $j, k$ with $j < k$, $r_k \geq r_j + p_j$. Why can this assumption be made without loss of generality?

(b) Consider the linear programming relaxation in $C_j$ variables used to derive a 2-approximation algorithm for $1|prec| \sum w_jC_j$; give a simple set of $n$ inequalities that reflect that there are release date constraints. Explain why this linear program is also a relaxation for the preemptive version of the problem.

(c) The linear programming relaxation in the previous part is weak in the following way: consider a set of job $S$ in which all of the jobs have large release dates; if one considers the constraint

$$\sum_{j \in S} p_jC_j \geq f(S),$$

where $f(S) = (p^2(S) + p(S)^2)/2$, it should be clear that $f(S)$ now provides a weak lower bound. Use this insight to derive a new function $g(S)$ that provides a stronger lower bound that incorporates these release dates.

(d) Use the stronger linear relaxation to derive a 2-approximation algorithm. *Hint*: since the algorithm may produce a preemptive schedule, one can take advantage of idle gaps in the schedule when adding in the “next” job.

10. (Research question) The best known performance guarantee for $1|prec, r_j| \sum w_jC_j$ is $e$. There is every reason to think that a factor of 2 should be achievable, and might even be achieved by a relatively simple approach.

11. (Research question) Goemans and Williamson gave a primal-dual proof of Lawler’s algorithm for $1|sepa| \sum w_jC_j$ based on the $C_j$-style LP formulation. We have seen that the weak linear ordering formulation is also integral for series-parallel inputs. Can one give an alternate proof of this result that also yields a proof of correctness for Lawler’s algorithm?