A Continuous Relaxation for the Cumulative Constraint

J. N. Hooker
Carnegie Mellon University
Hong Yan
Hong Kong Polytechnic

October 2001

The Cumulative Constraint

For resource-constrained scheduling

$$\text{cumulative}(t, d, r, L)$$

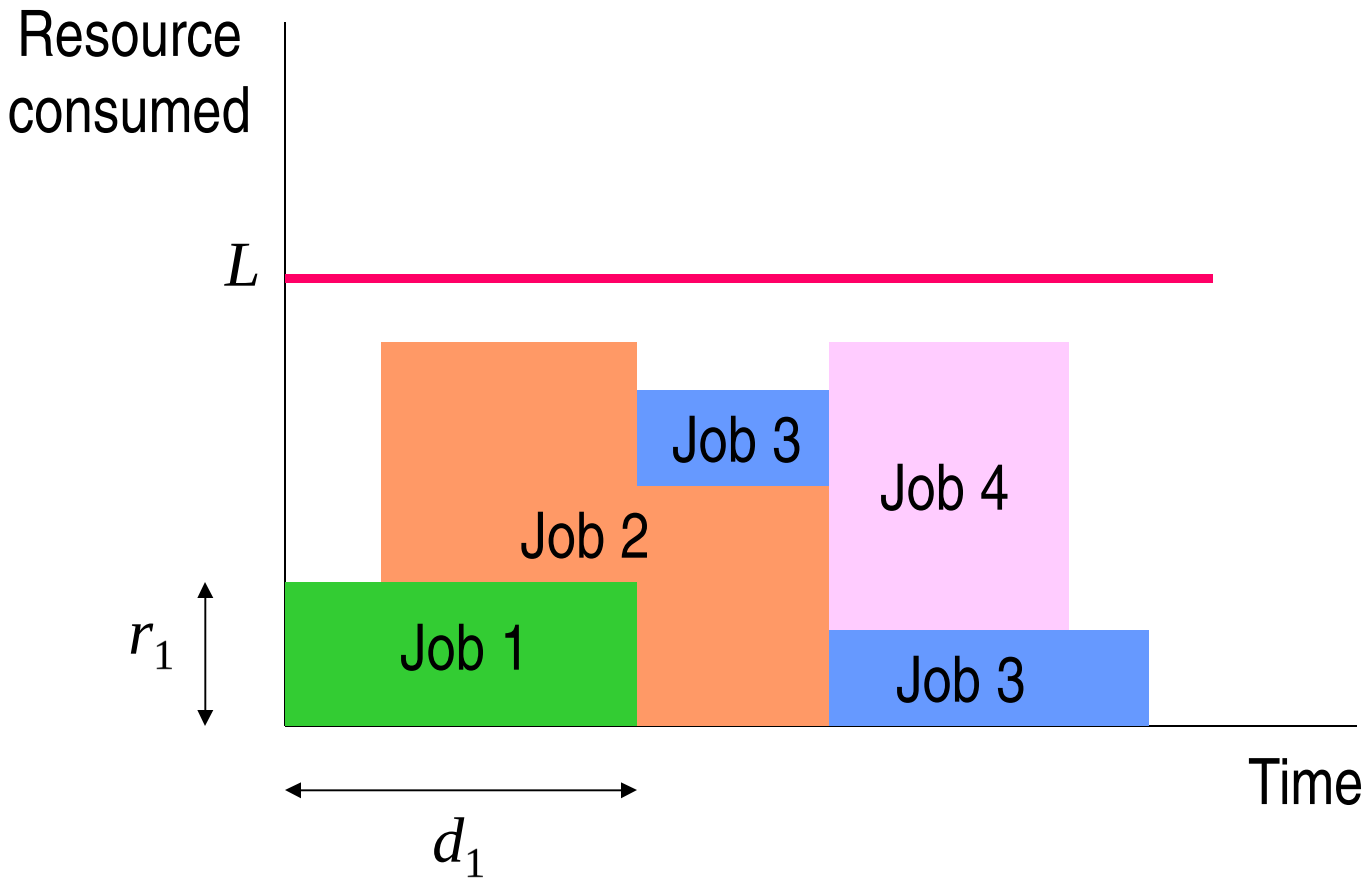
Start times $t = (t_1,\ldots,t_n)$

Durations $d = (d_1,\ldots,d_n)$

Resource consumption rates $r = (r_1,\ldots,r_n)$

Bound on total resource consumption

- Schedule jobs 1,…,*n* so that the total resource consumption at any one time is at most *L*.
- Earliest and latest start times are given by the initial *domains* (a_j, b_j) of each t_j.


Resource consumed
L
Job 3
Job 4
Job 2
r_1
Job 1
Job 3
Time
d_1

Domain Reduction for Cumulative

$$\text{cumulative}(t, d, r, L)$$

- One or more *filtering* algorithms deduce that the some of the start times cannot be feasible. (Edge-finding.)
- The algorithms exploit the structure of this *global constraint* (much as cutting plane algorithms exploit structure).
- The domain of t_j is reduced from (a_j, b_j) to (a_j', b_j').
- This has proved a powerful technique for scheduling in a constraint programming context.

Continuous Relaxation for Cumulative

- We wish to find a continuous relaxation for Cumulative, expressed solely in terms of the variables t_j (no integer variables).
- Constraint programming traditionally does not use relaxations, but they are very useful in hybrid CP/OR methods:
  - Branch-and-bound methods (known as *branch and relax* in CP), in which one can bound the optimal value by solving a relaxation of global constraints.
  - Decomposition methods, in which the master problem can be strengthened by adding a relaxation of global constraints in the subproblem.

Example: Shop Scheduling by Decomposition

- Each of n jobs must be processed in one of m shops. Job j costs c_{ij} to process in shop i, requires time d_{ij}, and uses resources at the rate r_{ij}. Shop i has a max of L_i resources.

- Master problem (can be solved as MILP):

$$\begin{aligned} \min \quad & \sum_j c_{y_j j} \\ \text{s.t.} \quad & a \le t \le b \\ & y_j \in \{1, \ldots, m\}, \text{ all } j \end{aligned}$$

Shop assigned machine j

- Subproblem:

$$\text{cumulative}\big((t_j \mid y_j = i), (d_{ij} \mid y_j = i), (r_{ij} \mid y_j = i), L_i\big), \text{ all } i$$

- Solve by logic-based Benders decomposition.
- An infeasible subproblem generates a “Benders cut”; e.g., the jobs currently assigned to shop *i* cannot all be assigned to shop *i*.
- Jain and Grossmann (2001) applied this method to a special case in which each shop processes one job at a time.
- Obtained factor of 1000 speedup relative to MILP (CPLEX) and CP (ILOG scheduler called by OPL).
- Thorsteinsson (2001) pointed out that key to success was the presence of a relaxation of *cumulative* in the master problem.
- Jain & Grossmann used a very simple relaxation for the special case.

The Relaxation Problem

- We wish to relax

$$\text{cumulative}(t, d, r, L)$$

$$a \le t \le b$$

using inequalities of the form

$$t_{j_1} + \cdots + t_{j_k} \ge h$$

$$t_{j_1} + \cdots + t_{j_k} \le h$$

Derived from "lower problem," in which $b_j = \infty$

Derived from "upper problem," in which $a_j = -\infty$

- We will focus on lower problem.

We can find a valid RHS of $t_{j_1} + \cdots + t_{j_k} \geq h$

By solving a relaxation of the *cut problem*

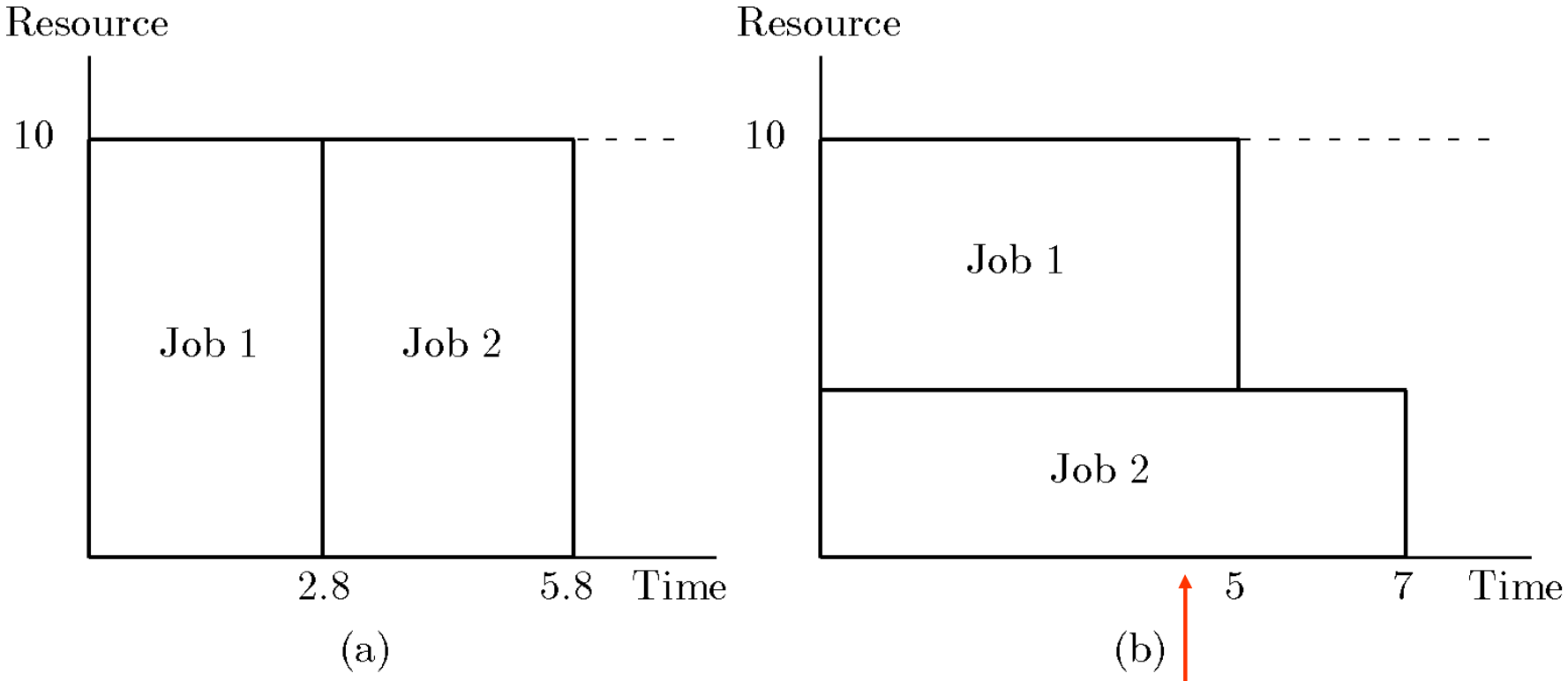
$$\begin{aligned} \min \quad & t_{j_1} + \cdots + t_{j_k} \\ \text{s.t.} \quad & \text{cumulative}(t, d, r, L) \\ & t \geq a \end{aligned}$$

We will:

- Use a bin packing relaxation of the above to obtain some facet-defining cuts.
  - When the cuts are facet-defining, the bin-packing relaxation can be solved in closed form.
- Solve a continuous relaxation of the bin packing problem to obtain other cuts.
  - The continuous relaxation can be solved by a fast greedy heuristic.

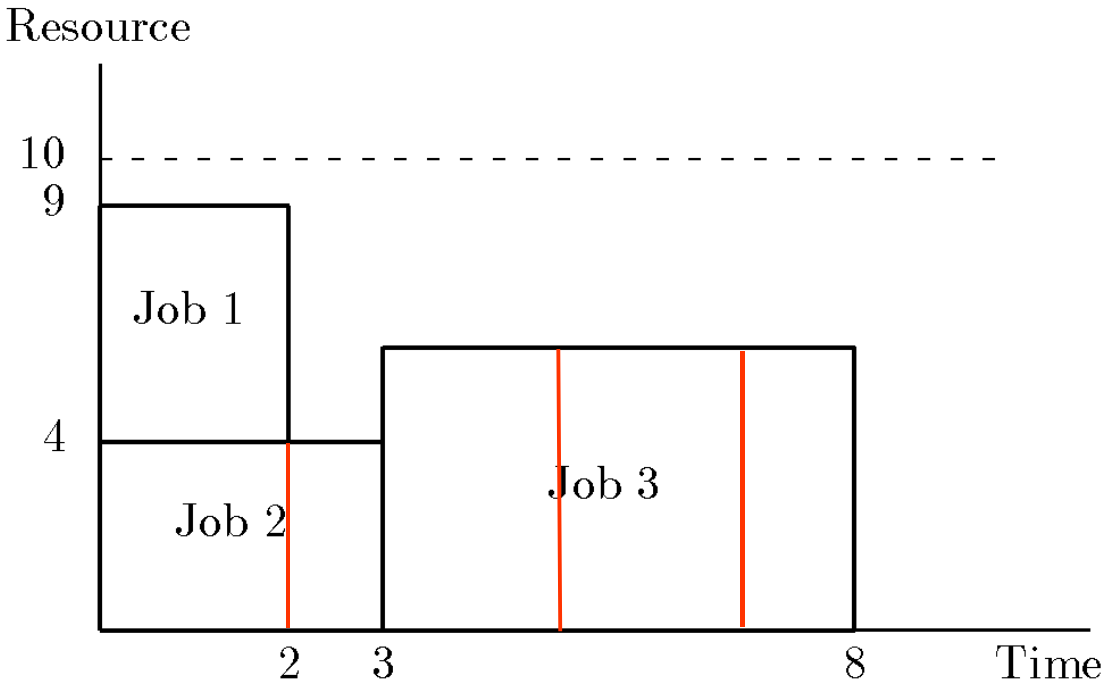
A continuous relaxation is not as easy as one might think…

- Schedule 2 jobs with $(d_1, d_2) = (7,5)$ and $(r_1, r_2) = (4,6)$.
- "Compress" each job into a job of the same area that consumes resources at the maximum rate.



$$t_1 + t_2 \geq 2.8$$ ← Cut is invalid, because this is the optimal solution of the cut problem; $t_1 + t_2 = 0$..

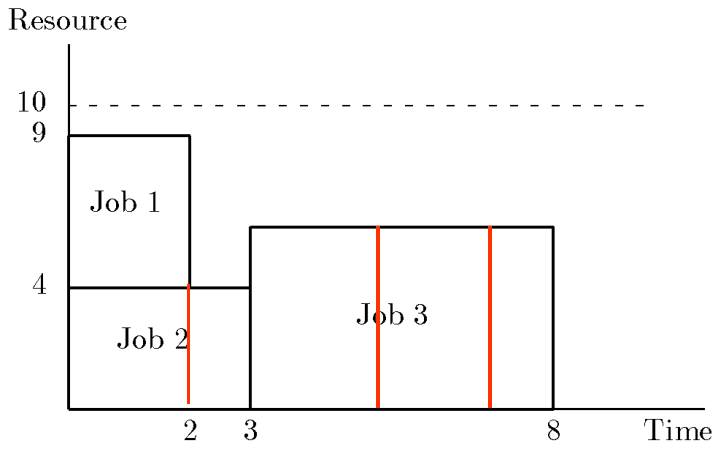
A Bin Packing Relaxation

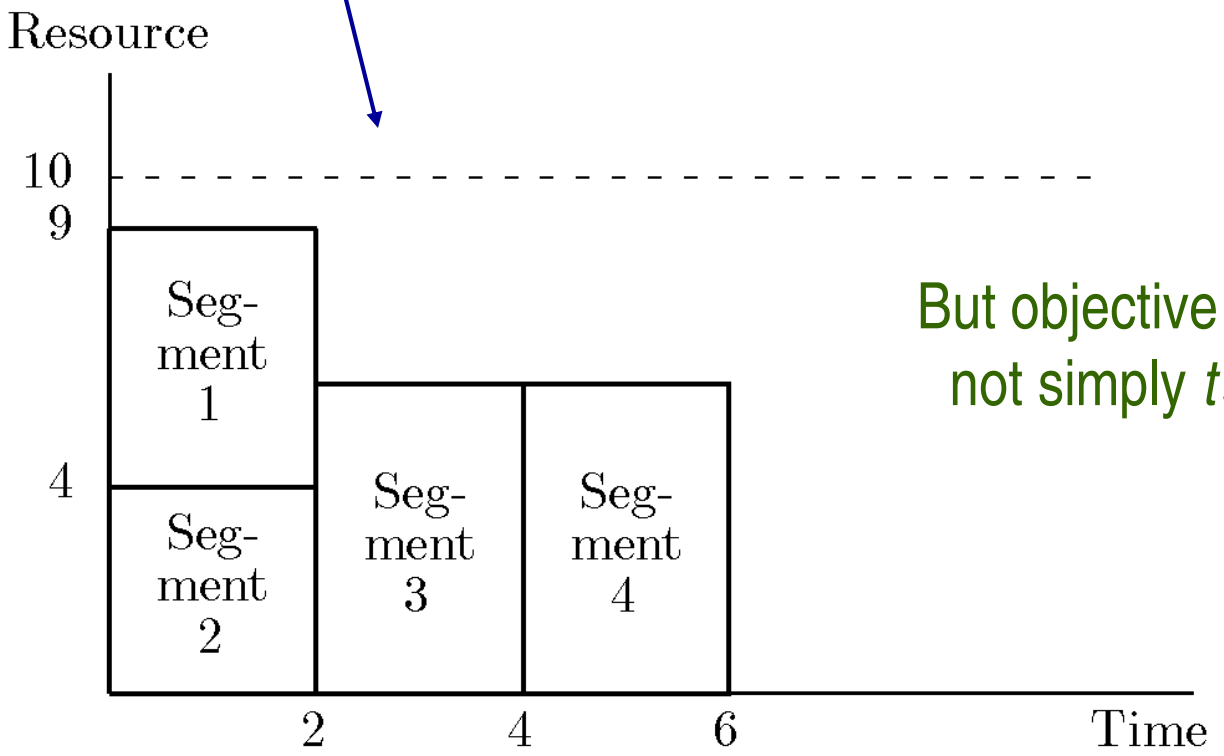


Optimal solution of a cut problem with 3 jobs.

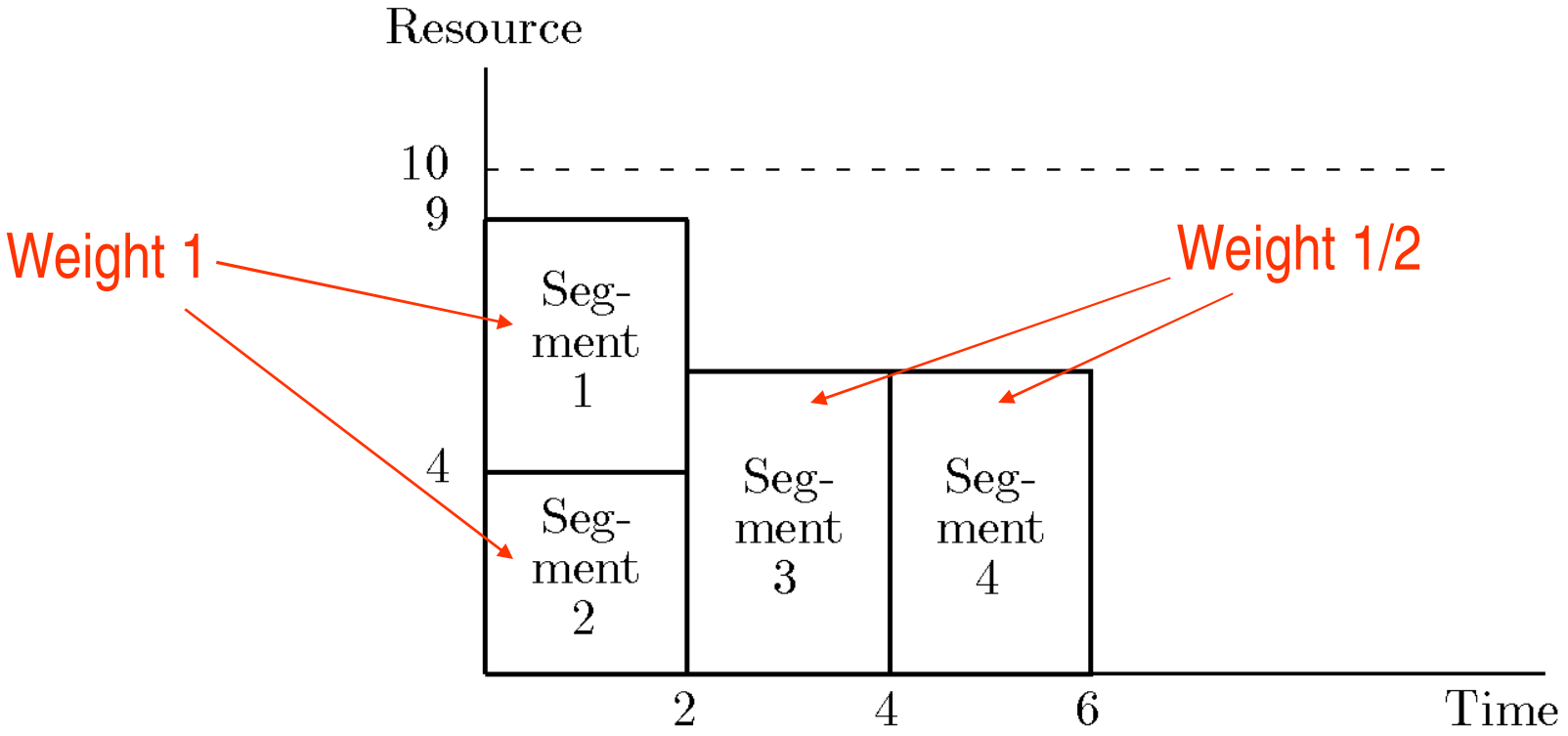
- We want to formulate a bin packing relaxation, using bins of width 2.
- Split jobs 3 into pieces of width $\Delta d = 2$ (discarding leftovers)

Optimal solution of the bin packing relaxation





But objective function is not simply $t_1 + t_2 + t_3$.



Objective function weights segments and corrects for excess...

$$t_1 + t_2 + t_3 \geq 1 \cdot 0 + 1 \cdot 0 + \frac{1}{2} \cdot 2 + \frac{1}{2} \cdot 4 - \frac{1}{2} \cdot 2 = 2$$

Weights

Start times

Correction for excess

Correction for excess is computed

$$E = \sum_{i=1}^{k} E_{j_i}$$

where

$$E_{j_i} = \frac{1}{2}\left(n_{j_i} - 1\right)\Delta d$$

Number of segments into which job j_i is split

In certain cases, the bin packing problem can be solved in closed form and yields a facet-defining cut.

Theorem. If jobs $j_1, \ldots, j_k$ are identical (have the same duration d_0 and release time a_0, and consume resources at the same rate r_0), then the following defines a facet of the convex hull of *cumulative*:

$$t_{j_1} + \cdots + t_{j_k} \geq (P+1)a_0 + \tfrac{1}{2}P[2k - (P+1)Q]d_0$$

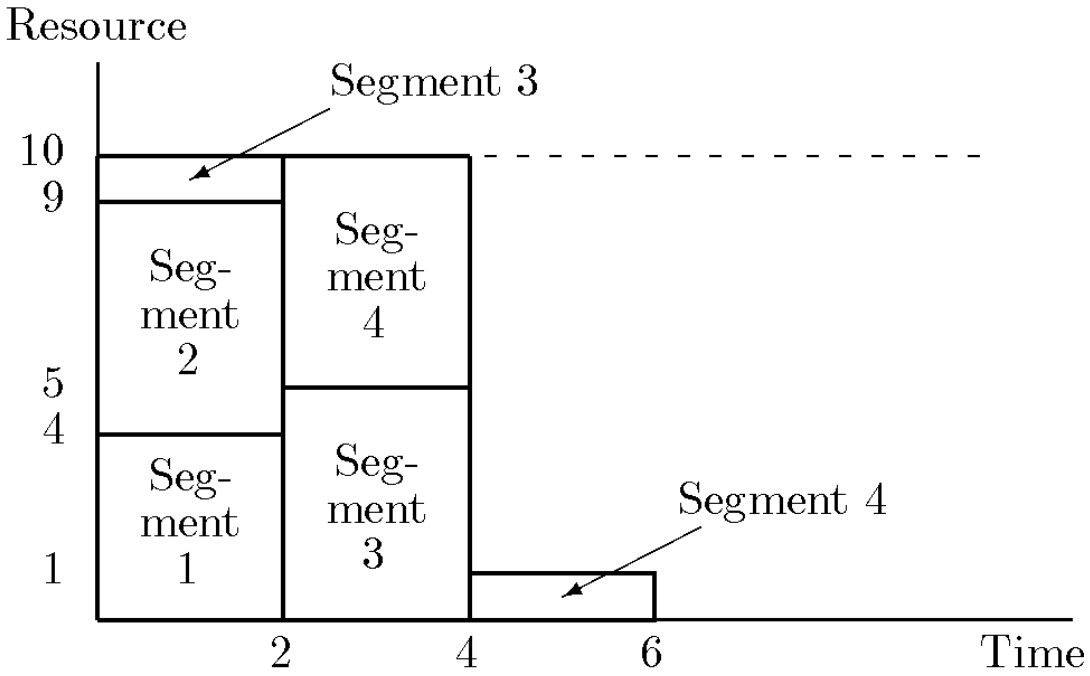
where

$$Q = \lfloor L / r_0 \rfloor$$

$$P = \lceil k / Q \rceil - 1$$

Continuous Relaxation of the Bin Packing Problem

To obtain more general cuts one can use a greedy algorithm to solve a continuous relaxation of the bin packing problem.



$$t_1 + t_2 + t_3 \geq 1$$

The relaxation can be written as an LP:

$$\begin{aligned}
\min \quad & \sum_{j=1}^{k'} \sum_{p=1}^{P'} w_j (p-1) y_{jp} \Delta d - E \\
\text{s.t.} \quad & \sum_{j=1}^{k'} r'_j y_{jp} \leq L, \; p = 1, \ldots, P' \qquad (\lambda_p) \\
& \sum_{p=1}^{P'} y_{jp} \geq 1, \quad j = 1, \ldots, k' \qquad (\mu_j) \\
& y_{jp} \leq 1, \quad \text{all } j, p \qquad (\nu_{jp}) \\
& y_{jp} \geq 0, \quad \text{all } j, p
\end{aligned}$$

Theorem. The greedy algorithm solves the continuous relaxation.

Proof. Use the primal and dual LP solutions

$$y_{jp} = \begin{cases} 1 & \text{if } R_{j-1} \geq (p-1)L \text{ and } R_j \leq pL \\ \dfrac{R_j - (p_j - 1)L}{r_{p_j}} & \text{if } R_{j-1} < (p-1)L \text{ and } R_j > (p-1)L \\ \dfrac{(p_j - 1)L - R_{j-1}}{r_{p_j}} & \text{if } R_{j-1} < pL \text{ and } R_j > pL \\ 0 & \text{otherwise} \end{cases}$$

$$\lambda_p = \left(\frac{w_{j(p)}}{r'_{j(p)}} + \sum_{\ell=j(p)+1}^{k} (p_\ell - p_{\ell-1}) \frac{w_\ell}{r'_\ell} \right) \Delta d, \quad p = 1, \ldots, P'$$

$$\mu_j = \left(w_j (p_j - 1) + r'_j \sum_{\ell=j+1}^{k'} (p_\ell - p_{\ell-1}) \frac{w_\ell}{r'_\ell} \right) \Delta d, \quad j = 1, \ldots, k'$$

$$\nu_{jp} = 0, \quad \text{all } j, p$$

Choice of Bin Size

Theorem. If the bin size $\Delta d \rightarrow 0$, the cut resulting from the continuous relaxation of the bin packing problem becomes

$$t_{j_1} + \cdots + t_{j_k} \geq \sum_{q=1}^{k} \left((k - q + \tfrac{1}{2}) \frac{r_q}{L} - \tfrac{1}{2} \right) d_q$$

Any of the following can deliver the strongest cut:

- Set $\Delta d = 0$.
- Set Δd = duration of shortest job.
- Set Δd somewhere in between.

Strongest cut at $\Delta d = 2$, smallest job duration

RHS of cut

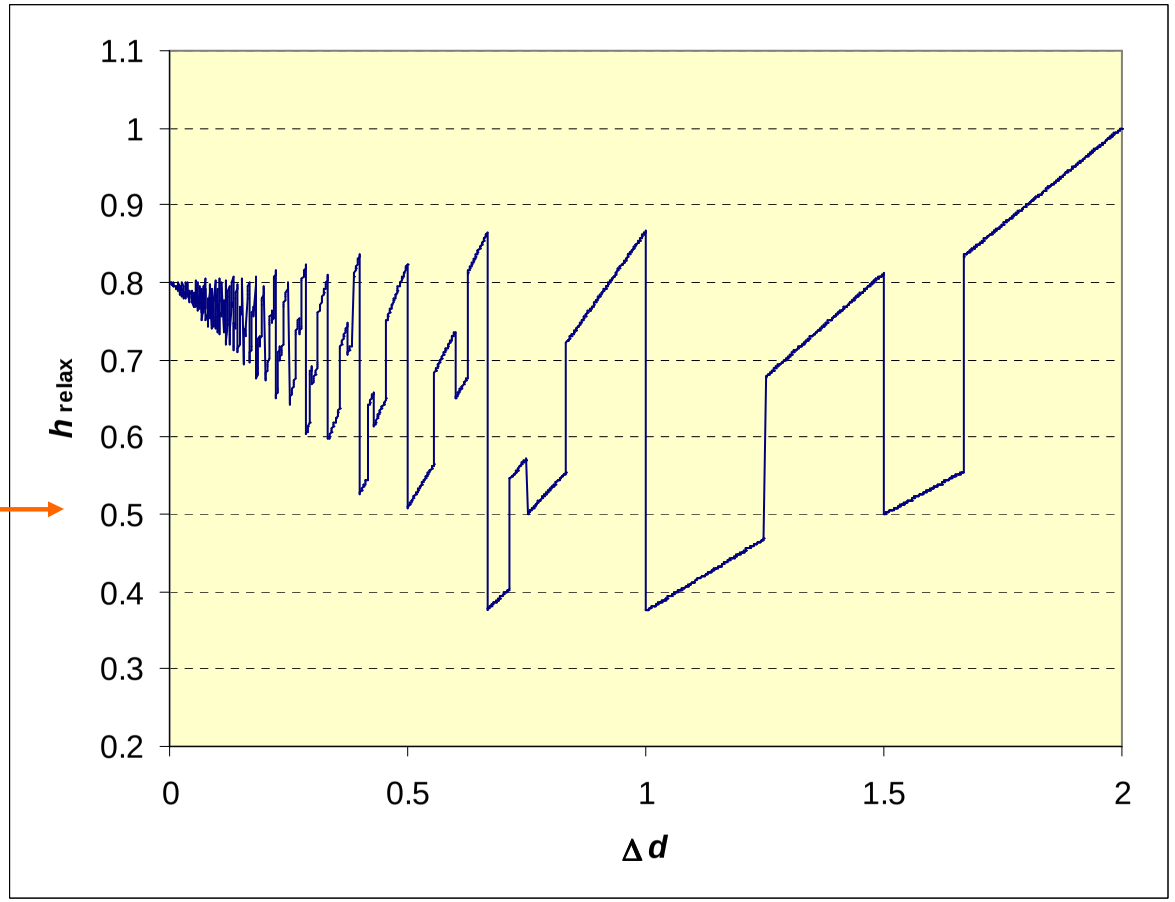


Figure 5: *Plot of* h_{relax} *versus* Δd *for example (23), a 3-job problem in which* $d = (2, 3, 5)$, $r = (5, 4, 6)$ *and* $L = 10$. *Here* $\Delta d = \min_i\{d_{j_i}\} = 2$ *yields the strongest cut, with* $h_{\text{relax}} = 1$. *The asymptotic value of* h_{relax} *as* $\Delta d \to 0$ *is* 0.8.

Strongest cut at $\Delta d = 0$

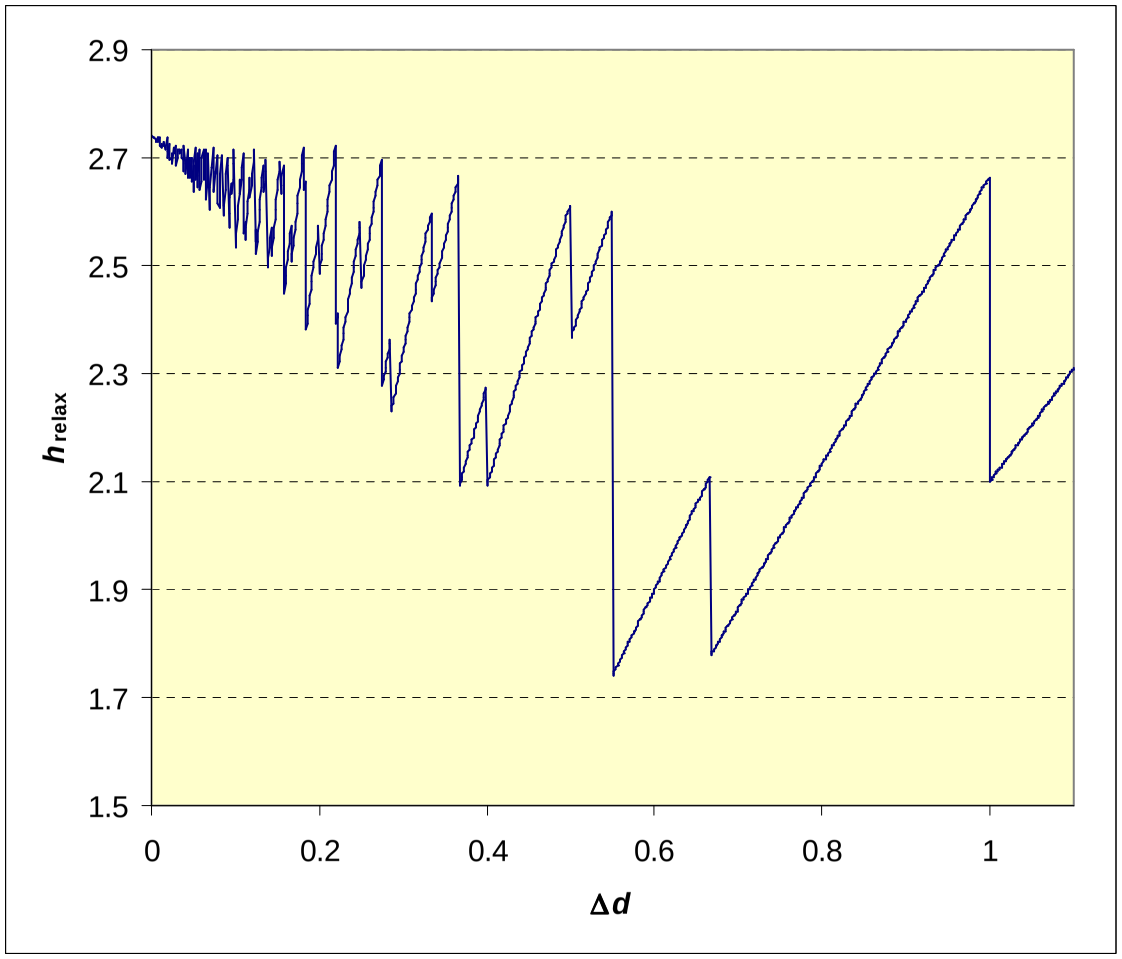


Figure 6: *Plot of* h_{relax} *versus* Δd *for a 3-job problem in which* $d = (1.1, 2, 2)$, $r = (5, 4, 6)$, *and* $L = 6$. *Here* $\Delta d = 0$ *yields the strongest cut, with* $h_{\text{relax}} = 2.742$.

Strongest cut at $\Delta d = 1$, between 0 and smallest job duration (1.1)

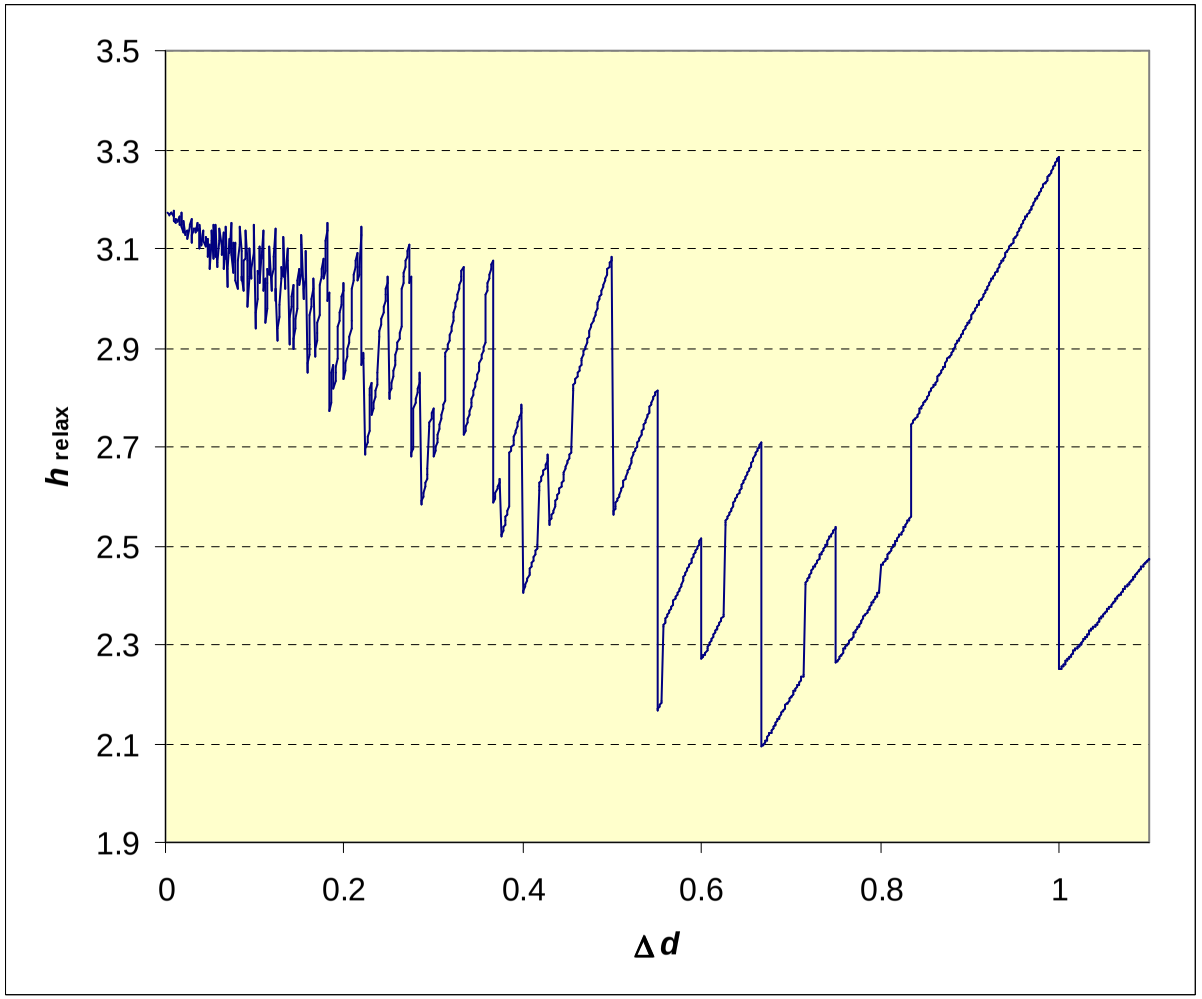


Figure 7: *Plot of* h_{relax} *versus* Δd *for a 5-job problem in which* $d = (1.1, 2, 3, 4, 5)$, $r = (5, 5, 4, 4, 6)$, *and* $L = 12$. *Here* $\Delta d = 1$ *yields the strongest cut, with* $h_{\text{relax}} = 3.287$. *The asymptotic value of* h_{relax} *as* $\Delta d \to 0$ *is* 3.179.