Logical Benders for Scheduling

John E. Mitchell

Department of Mathematical Sciences
RPI, Troy, NY 12180 USA

April 2017
A scheduling problem

Some classes of scheduling problems can be solved by a type of logical Benders decomposition popularized by Hooker [2, 3, 1].

Consider a problem where we have to assign $J$ jobs to $m$ machines and then schedule the jobs on the machines.

Each job $j$ has a release time $a_j$ and a due date $b_j > a_j$. Work cannot start on job $j$ until time $a_j$ and it must be finished by time $b_j$.

It also has a processing time depending on the machine $i$ chosen, denoted $p_{ij}$.

We have a cost $c_{ij}$ for processing job $j$ on machine $i$ and the objective is to minimize the sum of these costs.
Logical Benders Decomposition

A Benders-type of approach to this problem:

*construct a Master Problem that assigns jobs to machines*

and then

*have separate subproblems for each machine that schedule the jobs assigned to that machine.*

If the subproblem is infeasible then a constraint is fed back to the Master Problem and the Master Problem is solved again.
Logical Benders Decomposition

A Benders-type of approach to this problem:

construct a Master Problem that assigns jobs to machines

and then

have separate subproblems for each machine that schedule the jobs assigned to that machine.

If the subproblem is infeasible then a constraint is fed back to the Master Problem and the Master Problem is solved again.
Master Problem

We can write the Master Problem as

\[
\begin{align*}
\min_x & \quad \sum_{i=1}^m \sum_{j=1}^J c_{ij} x_{ij} \\
\text{subject to} & \quad \sum_{i=1}^m x_{ij} = 1 \\
& \quad \text{constraints to prevent invalid assignments to machine } i \\
& \quad x \text{ binary}
\end{align*}
\]

(1)
Subproblems

Given a solution to the Master Problem (1), the subproblem for machine \( i \) has to determine start times for each job \( j \) assigned to it in order to meet the time window constraints.

If the subproblem is infeasible then there is a time interval \([a, b]\) which is too short for the jobs assigned to the machine.
More specifically, let \( J(i) \) be the jobs assigned to machine \( i \) and let

\[
J^b_a(i) := \{ i \in J(i) : a \leq a_j \text{ and } b_j \leq b \}. \tag{2}
\]

If

\[
\sum_{j \in J^b_a(i)} p_{ij} > b - a \tag{3}
\]

then this set of jobs cannot all be assigned to machine \( i \).
Master Problem

Denote this set of jobs by \( \hat{J} \). A valid constraint for the Master Problem is then

\[
\sum_{\hat{j}} (1 - x_{ij}) \geq 1. \tag{4}
\]

This forces \( x_{ij} = 0 \) for at least one of the jobs in \( \hat{J} \), thus preventing the assignment of this complete set of jobs to machine \( i \).

In order to speed up the algorithm, it is desirable to \textit{sparsify} the set \( \hat{J} \), that is, include as few jobs in the set as possible.

\textit{Like} minimal \textit{covers}.
Master Problem

The structure of this Benders algorithm is different from the others we have considered in that the subproblem is an integer program.

Thus, we cannot exploit LP duality in the construction of cuts for the Master Problem.

Instead, we have to use logical arguments to construct valid constraints for the Master Problem.

Hooker refers to the arguments as infeasibility proofs; he generalizes the idea of an LP dual to an inference dual.
Master Problem


Master Problem

J. N. Hooker.
An integrated method for planning and scheduling to minimize tardiness.

J. N. Hooker.
Planning and scheduling by logic-based benders decomposition.

J. N. Hooker and G. Ottosson.
Logic-based Benders decomposition.
Master Problem

J. N. Hooker.
An integrated method for planning and scheduling to minimize tardiness.

J. N. Hooker.
Planning and scheduling by logic-based benders decomposition.

J. N. Hooker and G. Ottosson.
Logic-based Benders decomposition.