A Benders decomposition based heuristic for the hierarchical production planning problem

Karen AARDAL * and Torbjörn LARSSON
Department of Mathematics, Linköping Institute of Technology, S-581 83 Linköping, Sweden

Abstract: In this paper a heuristic procedure for determining good feasible solutions to a multi-item dynamic production planning problem is described and evaluated. The problem, which is modelled as a mixed-integer linear program, has a hierarchical structure where production items are aggregated into families, and families into product types.

A decomposition of the original problem creates a trivial subproblem providing feasible solutions on the type level, and a master problem which separates into one uncapacitated lot-sizing problem for each family. The latter problems can be solved by dynamic programming with very little computational effort. Besides the nice structure of both the sub- and master problem, the dual information from the subproblem can be given interesting economic interpretations. As distinguished from Hax and Meal’s hierarchical framework for production planning and also they hybrid approach suggested by Graves in 1982, there exists a direct information flow from the family level problem to the type level problem in form of a production plan.

The heuristic is essentially an application of Benders decomposition, the only difference being that the Benders cuts are relaxed using Lagrangean multipliers. A subgradient procedure is used to update the Lagrangean multipliers. One iteration of this heuristic requires less computational effort compared to Graves’ algorithm. This is due to the fact that the Benders subproblem can be solved by inspection, while Graves’ algorithm involves solving an LP-problem on the type level. For thirtysix medium-size test problems the average deviation from optimum is 2.34%, and the deviation ranges between 0.00% and 5.95%.

Keywords: Hierarchical production planning, Benders decomposition, Lagrangean relaxation, subgradient optimization

1. Introduction

Production planning involves many complex decisions where the decision maker is confronted with conflicting objectives under the presence of scarce resources. The field has received much attention in the operations research literature where many different models and solution methods have been discussed and developed.

Several attempts have been made to formulate the production planning and scheduling problem as a large mixed-integer linear program, the so-called monotonic approach, see e.g. Manne [18], Duziński and Gomory [8], and Lasdon and Terpung [17]. Mixed-integer programs however offer great difficulties to solve to optimality as they are stated except for very small instances. The monotonic programs treated by [18,8,17] are all solved as linear programs by means of various large-scale programming techniques, the solution is then rounded to a feasible integer solution.

Another approach is hierarchical production planning, see e.g. Hax and Meal [14], where the objective function coefficients in both subproblems. The algorithm is very efficient and gives near-optimal solutions.

In this paper another hybrid approach is suggested that is applied to the same mixed-integer model as the one considered by Graves. Instead of using Lagrangean relaxation with respect to a certain group of constraints in the original model, the model is first partitioned according to Benders decomposition [4]. The Benders cuts are then priced out by a set of Lagrangean multipliers which are updated by subgradient optimization. The only subproblem that needs to be solved in this approach is an uncapacitated lot-sizing problem, while Graves’ algorithm also involves solving an LP-problem on the upper level. The algorithm works very well and results in an information flow with direct interaction between the subproblems on the different hierarchical levels. It should be noted that the presented solution principle is also applicable to more complex hierarchical production planning models than the one studied in this paper.

The outline of the paper is as follows. In Section 2 a formulation of the production planning and scheduling problem as a mixed-integer linear program is given. In Section 3 the technical details of the solution procedure are presented. The economic interpretations of the information flow between the problems generated by the algorithm are given in Section 4. Section 5 presents the computational results and finally, in Section 6, the method and results are discussed and some possible extensions are suggested.

2. A production planning and scheduling model

Production planning and scheduling have mainly two objectives. The planning function should determine the resource requirements and at what points in time they occur, in order to satisfy aggregate demand over the planning horizon. Also, the available resources should be allocated to specific products in order to satisfy customer requirements at a minimum cost over the scheduling period.

For planning purposes, the products to be manufactured can in certain applications be grouped into two levels of aggregation. The final products to be delivered to the customers, are
aggregated into families. The items in a family share a setup and therefore need to be considered jointly when preparing a production schedule. A type includes items that have the same demand pattern and all items belonging to the same type share a unit inventory holding cost. All items belonging to a family also belong to the same type and therefore a type can be viewed as an aggregation of families. This aggregation scheme has been proposed by Hax and Meal [14], and is reflected well in the model considered in this paper, in the sense that setup costs are accounted for by families and overtime and inventory holding costs are accounted for by types.

We now give a description of the model. The problem is to decide on a production schedule on the family level which in turn determines the inventory on the type level at the end of each period. This decision should be made in order to minimize the sum of overtime cost, inventory holding cost and setup cost subject to capacity restrictions and demand requirements, but as distinguished from the traditional hierarchical approach, both family and type decisions are included in one overall problem formulation. The problem under consideration is stated mathematically below.

Let

\[ T \]  

- the number of time periods,
\[ c_i \]  

cost of one hour overtime in period \( t \),
\[ h_j \]  

inventory holding unit cost for type \( j \) during type period \( t \),
\[ s_p \]  

setup cost for family \( f \) in time period \( t \),
\[ d_p(f_j) \]  

demand for type \( j \) in family \( f \) in time period \( t \),
\[ r_j \]  

regular production duration available during time period \( t \),
\[ k_j \]  

production time required per unit for type \( j \),
\[ T(f) \]  

set of families belonging to type \( j \),
\[ m_j \]  

an upper bound on the production quantity for family \( f \) during period \( t \),
\[ m_{j,t} \]  

\[ \sum_{i:t} d_p(f_j) \] .

The decision variables of the model are:

- \( O_t \)  

hours of overtime production during time period \( t \),
\[ I_{t}(F) \]  

inventory of type \( i \) (family \( f \)) during time period \( t \),
\[ \Pi_p(F_p) \]  

production quantity of type \( i \) (family \( f \)) during time period \( t \),
\[ X_{j} \]  

zero-one variable to indicate setup of family \( j \) in time period \( t \).
block-angular linear programming problem with common resource constraints is solved by the two respective methods. See e.g. [19] for a thorough discussion on this subject.

In a resource direct method, the subproblems are allocated a portion of the common resources provided by the master problem, while in price directive decomposition prices are put on the joint constraints which are placed in the objective function of the subproblem. The master problem then calculates optimal prices on the shared resources.

The present algorithm, which is based on Benders decomposition, resembles a price directive method in the sense that the dual prices play a central role. The dual variables $w_r$ and $w_s$, that are needed as input to the objective function of (LRPM) can be interpreted as variable production and inventory holding costs for families belonging to a certain product type. These interpretations also support the choice to assign a highest value as possible to the dual variables, as mentioned in the previous section. The alternative of letting $w_r$ and $w_s$ take on the value zero whenever possible, intuitively yields a more realistic problem considering these interpretations.

As pointed out in the introduction, a drawback of the traditional hierarchical approach is the lack of a feedback process from the lower-level to the higher-level subproblem. Graves' method involves feedback but not a direct interaction between the family and type decision problems. The decomposition on which our algorithm is based leads to an information flow that, from a hierarchical point of view, can be interpreted as if the inventory holding cost on the type level is divided into an inventory holding cost and a variable production cost that in turn are allocated down to the family level. The problem on the family level then suggests a new production plan based on these costs.

In Figure 1 the information flows in Graves' algorithm and the algorithm presented in this paper are illustrated.

5. Computational study

The computational study includes thirty-six test problems divided into four sets. Each problem set contains nine test problems. All problem data are generated according to Graves [13]. For the first two problem sets, product structures containing twenty families are studied. These families are aggregated into three types with five families in the first two types and ten families in the third type. The third and fourth set of problems contain product structures with forty families where the first two types each consist of ten families and the third type of twenty families. The number of time periods is twelve for all test problems. For all problem sets, the inventory holding cost and overtime cost are equal and take on the following values:

$$h_t = 1.0, \quad h_s = 1.75, \quad h_r = 1.5$$

for all time periods, $c_r = 5.0$ for all time periods.

Within each set of problems, both setup cost and resource coverage, i.e. the available regular time, are varied. The setup cost can assume three different levels: low, medium, and high. The low level is one fifth of the medium and the high level is five times the medium. The medium level is set. The available regular time can cover 80%, 100% and 120% of total demand over the planning horizon. The medium setup cost for a family $f$ belonging to type $i$ is generated randomly for each time period from the uniform distribution with range $(S_i, S_i)$ given in Table 1.

For problem set $K$, the available regular time in time period $t$ when the resource coverage is 100% is calculated as

$$r^K_i(t) = \frac{1}{12} \sum_{i=1}^{12} k_i d^K_i,$$

where $k_i$ is 1.0, 2.0 and 1.5 respectively for the three types and $d^K_i$ is obtained by aggregating family demand. The demand for family $f$ in time period $t$ is given by

$$d^K_{f,t} = f^K_{i,t} u^f, \quad f \in T(i),$$

where $f^K_{i,t}$ is the seasonality factor for type $i$ in time period $t$ in problem set $K$. No seasonal variation of demand is considered for problems in the first and third set (i.e. $f^K_{i,t} = 1$, all $i$, $t$ and $K = 1, 3$). For the problems in the second and fourth set the seasonal factors in Table 2 are used.

The family demand $u^f$ is generated by a random drawing from the uniform distribution over the ranges $(Q_f, D_f)$. The ranges for families 1–20 are given in Table 2 and these ranges are the same for families 21–40.

Finally, the initial inventories for the respective families are determined by a random drawing from the uniform distribution with ranges $(0, Q_f)$ where $Q_f$ is the economic order quantity for family $f$.

All test problems were solved on a DEC-20 using a code made by the authors. The average CPU-time was 72 seconds for the problems in the first and second problem set and 184 seconds for the problems in the third and fourth set. The algorithm was terminated after four main iterations. At the first three main iterations, the Lagrangean multipliers were updated five times and at the last iteration 105 updates were made. The step-length constant in the subgradient proce-

<table>
<thead>
<tr>
<th>Table 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
</tr>
<tr>
<td>$S$</td>
</tr>
<tr>
<td>$S$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time period</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Type 1</td>
</tr>
<tr>
<td>Type 2</td>
</tr>
<tr>
<td>Type 3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Families</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>$D_1$</td>
</tr>
<tr>
<td>$D_2$</td>
</tr>
</tbody>
</table>
mulated as a mixed-integer linear programming problem, has been developed and evaluated. The algorithm, which is based on Benders decomposition, brings the problem to separate into two subproblems that correspond to the aggregate and disaggregate level in Hax and Mehl’s [14] hierarchical framework.

The contribution of this paper is threefold. Firstly, the algorithm itself is conceptually interesting and similar methods have been investigated in the literature only to a small extent. Hoang Hai Hoc [16] applied generalized Benders decomposition to a class of network topological optimization problems formulated as nonlinear mixed integer models. He used Lagrangean relaxation to solve the master problem and could report encouraging results. Another application of this type of method is due to Ferland and Florian [9] who studied a large-scale 0–1 programming problem. The problem was formulated as a Benders master problem. They tried to use the Lagrangean relaxation approach but were not able to move optimally in the feasible region of the Lagrangean dual. However, they used the optimal value of the dual to get a lower bound on the optimal value of the original problem. When applying Lagrangean relaxation to a Benders master problem it is in each iteration of the subgradient procedure necessary to make a projection onto a unit simplex in the dual space in order to invoke dual feasibility, see [6]. In our case this projection is trivial, but in general it involves the solution of a quadratic knapsack problem. This problem can, however, be solved very efficiently by utilizing an algorithm by Brucker [5].

Secondly, the algorithm performs very well and the results are comparable with those reported by Graves [13] with respect to tightness between upper and lower bounds. The average percentage deviation from optimum was only 2.34% and the maximum deviation was 5.95%. For three problems the optimal solution was found. Nine out of eighteen test problems having 480 0–1 variables gave solutions that were within 2.0% of the lower bound.

The average CPU-time for the problems in set 1 and 2 was 72 seconds and for the problems in set 3 and 4, 184 seconds. The computational effort required at each iteration of our algorithm versus Graves’ algorithm is comparable apart from the solution of a type level LP subproblem of the latter method. According to Graves, the setup and solution of the LP-problem accounts for much of the required computational time. In this context it could be mentioned that the authors tried to solve a five-family, seven-period problem (35 0–1 and 105 continuous variable) as it was stated, by using a commercial mixed-integer code. But, the calculations were interrupted after ten CPU-hours on a DEC-20 computer without having been able to reach optimality.

Finally, we consider the third contribution of this paper to be that the algorithm provides an information flow that is interesting both in terms of economic interpretations and in terms of a feedback process in the hierarchical planning context.

For further research it would be interesting to investigate how the algorithm could be modified if there were additional production resource constraints and decision variables added to the model. The existing restriction on family production just implies that there is no point in producing more in the present time period than the total remaining demand. If a restriction is added to the model that requires the sum of production on the family level in each time period to be less than or equal to some relatively tight upper bound, the Lagrangean subproblems will get a lot simpler and the problem and the Wagner–Whitin algorithm can no longer be applied. Recently Barany [2] presented a way of reformulating the capacitated lot-sizing problem using valid inequalities for generating the convex hull of the feasible solutions to the uncapacitated case that might be exploited in a modified version of the algorithm. Possible extensions of the model and the algorithm concerning additional variables are e.g. workforce smoothing and capacity expansion decisions.

Furthermore, in order to consider the ideas behind hierarchical planning in a larger extent. Graves [13], suggested that it would be more realistic to use a shorter scheduling horizon for the product families. For instance, the families could be scheduled by weeks for a few months. Some inventory consistency constraints then need to be added to the model.

Acknowledgement
The authors wish to thank Prof. Jan Karel Lenstra, Dr. Kaj Rosling, Prof. Jørgen Tind and an anonymous referee for their helpful comments and suggestions which improved the presentation. The research leading to this paper was supported by the Swedish Transport Research Board (TFB) and the Swedish National Board for Technical Development (STU).

References


