Production Planning and Scheduling of Parallel Continuous Processes with Product Families

Name: Aditya Biyani

Guide: Prof. Chrysanthos Gounaris

Course: 06-635 Production and Supply Chain Optimization

School: Carnegie Mellon University

Index

Contents

1. Description of the paper

The objective of the paper is to develop an approach to simultaneously perform production planning and scheduling of continuous parallel units that can produce a large number of products. The final products can thus be classified into several different families. The paper describes an efficient mixed integer programming formulation for the production planning and scheduling with sequence-dependent switchover times and costs for product families and sequence-independent switchover times and costs for product families in the same period. (Kopanos, Puigjaner, & Maravelias, 2011)
The optimization goal is to minimize the total inventory, backlog, changeover (within and across periods), and setup and operating costs.

The proposed model in the paper is motivated from the industry itself. Such problems quite often arise in process industries, packaging industries (like food and beverage industry). It was designed to be able to address the real-world production facilities and in some cases, also yield better results that are currently available through different tools. (Kopanos, Puigjaner, & Maravelias, 2011)

Aim of production planning is to determine production amounts and inventory levels which will allow us to fulfill the customer demands at the least cost possible. This cost is subject to some capacity constraints and also includes processing, holding and backlog and switchover costs).
Whereas, scheduling is optimally allocating the limited resources at hand (like manpower, utilities and equipment units). The competing tasks with production lot size, changeovers times and tasks on units are subject to some production constraints.
Thus, both production planning and scheduling are interdependent tasks as solution of production planning is input to scheduling and the production capacity constraints restrict the scheduling model. (Kopanos, Puigjaner, & Maravelias, 2011)

There have been many standalone papers on production planning and scheduling separately using mixed-integer programming. (Méndez, Cerdá, Grossmann, Harjunkoski, & Fahl, 2006) provide an overview process scheduling approaches. (Kallrath, 2003) and (Shah) provide an overview of the integrated solutions to scheduling and production planning.
The major focus of this paper was to have an independency of production and scheduling with parallel units. They are also called single-stage processes. For such problems, a number of methods in the literature have been reported. (Chen) studies the medium-term planning problem of a single-stage single-unit continuous multiproduct polymer plant. A slot-based MIP model on hybrid discrete/continuous time representation, where the production planning horizon was divided into several discrete weeks, and each week

was formulated with a continuous time representation. (Liu) further improved the work of Chen by presenting a MIP formulation based on the classic traveling salesman problem (TSP) formulation. Their proposed model, without time slots, was computationally more effective. (Erdirik-Dogan, Grossmann) proposed a multiperiod slot-based MIP model for the simultaneous planning and scheduling of single-stage single-unit multiproduct continuous plants. A bilevel decomposition algorithm in which the original problem is decomposed into an upper level planning and a lower level scheduling problem was also developed in order to deal with complex problems.
This paper studies a more general case by considering product families, short planning periods that may lead to idle units for entire periods and changeovers spanning multiple periods, and maintenance activities. The grouping into families can be based on various criteria, including product similarities, processing similarities, or changeover considerations. The goal of the aforementioned grouping is to lead to computationally tractable optimization models without compromising the quality of solution. (Kopanos, Puigjaner, & Maravelias, 2011)

The method of devising the model is novel in the sense that it uses three different approaches. First, the discrete-time approach for inventory and backlog cost calculation for production planning. Second, a continuous-time approach with sequencing using immediate precedence variables for scheduling of families. Third, lot sizing type of capacity constraint for scheduling of products.

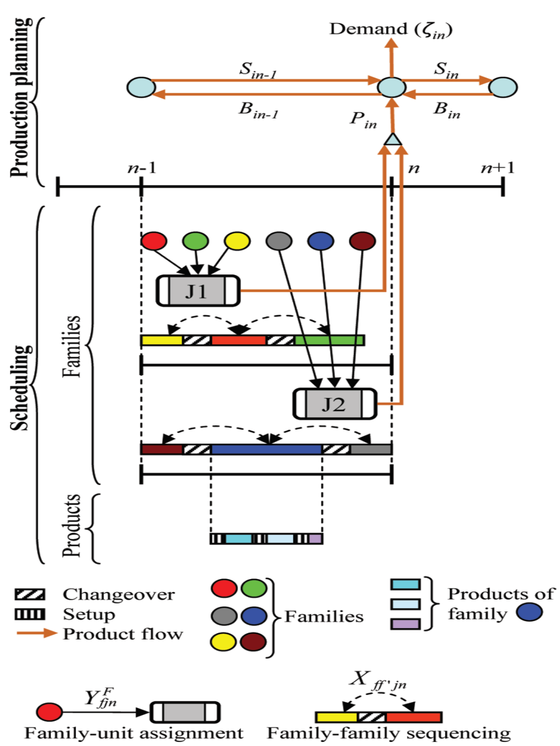

Figure 1: Proposed modeling approach

The above diagram explains the three distinct approaches. The production planning constraint is defined by the material balance over the production of each product in each period *n* belongs to {1, 2, 3…., N}. The mass balance also includes the inventory storage (S_{in}) and backlog (B_{in}).

Scheduling involves two levels. Allotting a particular family (F) to a particular unit (j) in a particular period (n) that is controlled by a binary variable (Y^F_{jn}) and also allotting a particular product from each family to that unit. There are binary variables that control the sequencing of the each product in the family and each family in the period. The constraints also confine the sequencing of these families and products in each unit. Sizing of the lots are also carried out. The unique features of this model are also that it incorporates the changeover of families and the fractional time they require to process in that unit when they crossover from one period to another. There are also constraints put on product processing time and rate. The model also takes of idle units (I_f) which basically act as dummy products, having no setup time and cost and their maximum processing time is equivalent to available production time (ω_{ij}). The model also takes into account the maintenance that is to be scheduled for each unit. The maintenance can be equal to the production period or less than that and the model takes care of thast as well. The main idea of how the model was designed can be understood from figure 1.

The dataset used in the paper is a simple example which is a small subset of a complex- real world continuous beer bottling process. It has 15 products (I01-I15) divided into 5 different families numerically. F01: {I01, I02, I03}, F02: {I04, I05, I06} and so on. The production process happens over a period of 4 days in 24 hour periods. Maintenance is scheduled on J01, J02, J03 units in periods N2, N3, N4. The values for setup time, setup cost, production rate, minimum and maximum processing time, product demands, changeover times.

2. Description of own effort

2.1. Implementation Details

The model is a mixed integer programming problem. It was solved on Lenovo ThinkPad 8 GB RAM using CPLEX 12 via GAMS 27.3 interface. The average time to run the MIP problem is about 43 CPU s. This time is basically a representation of amount of time practitioners will spend on getting an optimal solution in the industry. The above time is a very small data set and thus, scaling this would mean a non-linear increase in the time required to reach the optimal solution. This practice is prevalent in industry as we need to test different scenarios for dispatchment and rescheduling in a routinely fashion.

The model has constraints for product mass balance, inventory capacity, family sequencing in the same period and consecutive periods, determining the sequencing of families in these units, avoiding sub-cycle sequence, family changeovers, unit production time, unit production and family and product processing times. These constraints are defined for the optimization goal of minimizing the total cost. This is cost is over inventory, backlog, changeovers, setup and operating costs.

Following were the challenges faced while understanding and replicating the data. First, the dataset was missing some values like the value of available production time (ω_{ij}) was not mentioned anywhere. Thus, we had to assume it to be 24 hours. The initial backlog and inventory (S_{in}, B_{in}) i.e. at n = 0 was also not mentioned. That was also taken to be '0'. The minimum production rate (ρ^{min}_{ij}) was also missing and thus, assumed to be 20 % of the maximum value of production rate. The extension for maintenance was unclear if the binary crossover have to be fixed to zero for just 'n' or even include 'n-1' period. The figure 7 in the paper has some typo error. The summation over $\bar{X}$ from period 1 to 2 is zero, whereas when there is maintenance in period 2, we cannot have changeover between families in that period and thus it should have been sum over X. The maintenance affects the X and $\bar{X}$ binary variables. Thus, the initialization of WF_{fjn} and WL_{fjn} are also affected by maintenance. This was prevented from giving errors by excluding the maintenance set from being implemented in the model for the above three constraints.

The smaller data still had values for all the parameters, which was not the case for the lager dataset in the industrial case-study. Without any source for the data, it was difficult to replicate the data for the same. The extensions to the above model provided in this report is the use of different MIP solvers (CPLEX, GUROBI and XPRESS) with different MIP options in each model. Important point to note was the optimal gap for model solver needs to be set to '0' i.e. *option optcr = 0.*

2.2. Results, Implementation and Comparison

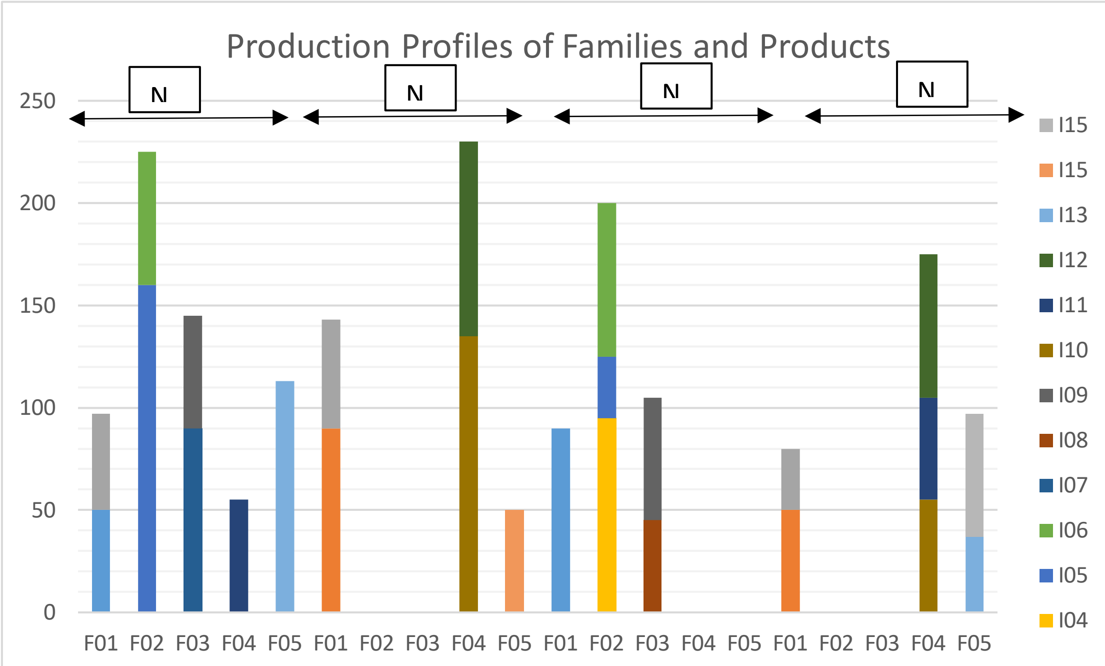

Figure 2: Production quantities for each family F in period N

Following is the production profile for each family in each period. This plot is same as the one reported in the paper. The optimal cost value reported in the paper is $ 2,630 and my result is $ 2,629.50 which is very close. The inventory, production and processing times are very similar to the ones reported in the paper. High inventory for N1 for I05 (90 kg), 107 (50 kg) and I13 (40 kg). High backlogs are seen in second day for products I08, I04 and in the third day for products I10, I11 and I12. Figure 3 shows a gannt chart of all the processing times of units produced in each period. This chart has the same values as the one reported in the paper but does not account for changeover and scheduling times in the graph.

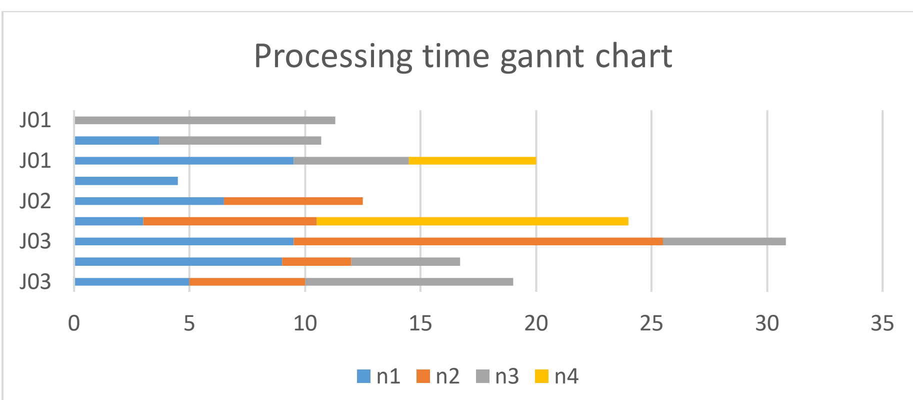

Figure 3: Processing time gannt chart

The model in the paper takes about 30 CPUs but my model took about 45 CPUs. There are various factors affecting the computing time like the RAM space while execution, CPU power, etc.

Table 1: Result comparison between reported and calculated observations

Comparison	Constraints	Binary var.	Relative gap (%)	Obj val ($)	CPU s
Paper	1147	510	0	2630	30
My answer	1151	510	0	2629.5	45

3. Presentation of extension of results

There are several ways in which the model could be extended. Some of them would be to introduce a larger dataset and check if the model behaves the same way as it did for smaller dataset. Or to define more slack variables to improve the model or test different solvers and the different options they provide to see of any parameter has a significant change on the run time or execution of the model.

The third idea is my extension of the result. Since, the model is a MIP. I used CPLEX and XPRESS solvers with different MIP options in each to address the tractability of the model.

The MIP options used in CPLEX were as follows:

- BendersStratergy – It is a method of formulation where CPLEX can decompose the model into a single master and multiple sub-problems. This option specifies CPLEX on how to implement the Bender algorithm as a strategy and make use of the annotations that use supplies to the model. It has 3 levels. We choose level 3, where it automatically decomposes the model, ignoring any annotation that use supplies.
- Disjcuts – disjunctive cuts generation determines whether or not to generate disjunctive cuts during optimization. We chose level 2 – generate cuts aggressively.
- Solvefinal – It switches to solve the problem with fixed discrete variables. We turn it off to avoid solving the problem with fixed discrete variables and avoid long run-time.

The MIP options used in XPRESS were as follows:

- cutStratergy – specifies the cut stratergy. The more aggressive the cut, the lesser nodes explored but increased time cost in generating the cuts. We chose a moderate cut strategy.
- defaultAlg – sets the default LP algorithm. Here we choose dual simplex / Newton Barrier.
- barAlg – determines which barrier algorithm to use. We start with homogeneous self-dual barrier algorithm and then optionally switch to infeasible-start barrier algorithm.

Result from Xpress: Resource usage time = 103 CPU s with optimal gap = 0 and 26 CPU s with gap = 45.6. Note: that the solution is fast with barrier algorithm but also takes sufficient time when cut strategy is introduced. It takes 760 different variables as compared to 1220 variables in the normal CPLEX function.

Result from CPLEX: It takes 83 CPU s to reach the optimal value. It has 1220 variables and uses dynamic MIP search.

From the comparison of both the results, we can say that if we want to reach an approximate value within an acceptable tolerance, we can use XPRESS solver whereas for an exact solution within limited cuts, CPLEX seems to be a better option.

4. References:

Kopanos, G. M., Puigjaner, L., & Maravelias, C. T. (2011). Production Planning and Scheduling of Parallel Continuous Processes with Product Families. *Industrial & Engineering Chemistry Research*, *50*(3), 1369–1378. doi: 10.1021/ie100790t

Méndez, C. A., Cerdá, J., Grossmann, I. E., Harjunkoski, I., & Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers & Chemical Engineering*, *30*(6-7), 913–946. doi: 10.1016/j.compchemeng.2006.02.008

Kallrath, J. (2003). Planning and scheduling in the process industry. *Advanced Planning and Scheduling Solutions in Process Industry*, 11–42. doi: 10.1007/978-3-662-05607-3_2

Shah, N. (n.d.). Production Scheduling. *Computer Aided Process and Product Engineering*, 481–516. doi: 10.1002/9783527619856.ch15