A MODELLING TECHNIQUE FOR PRODUCTION PLANNING

Jean-Claude HENNET
Laboratoire d’Automatique et d’Analyse des Systèmes du CNRS
7 avenue du colonel Roche - 31077 Toulouse Cedex

Abstract:
The purpose of a production plan is to help elaborating and combining decisions on a mean-term or a long-term basis. This objective explains why planning problems can only be solved for models generally obtained by aggregation and simplification. In the framework of an integrated decision structure, a production plan needs to be refined and specified through a detailed scheduling of operations on the various machines.
The production planning model presented in this paper is a discrete-time input-output model. It allows the use of optimal control methods. There is a correspondence between this model and the queueing network model describing the manufacturing system. This correspondence provides a direct interpretation of the decision variables of the planning problem as the reference values for the real-time release of processing orders.

1 Introduction

In manufacturing systems, the results of a planning problem are used as mean-term objectives for production units. These objectives are defined partly from the list of orders from customers and partly from the resources available for production: machines, work teams, supplies in raw materials.
In other words, planning models are built to generate reference performance requirements to the short-term monitoring level.
Therefore, a production plan is necessarily aggregated and simplified. Afterwards, it must be disaggregated and specialized to generate feasible sequences of tasks on machines.
Also, it should be robust and flexible so as not to overconstrain the scheduling problem and to allow for some real-time adjustments.
According to a hierarchical scheme similar to the one developed by S.B.Gershwin [9], the average production rates coming from the production plan will play the role of constraints for the lower hierarchical level, the scheduling one.
The proposed planning method relies on an aggregated model of the manufacturing facility. This model is drawn from an ideal model of the system, considered as a queueing network.
Products having similar processing sequences are unified into a particular type, and the demands for these products are summed up. In a similar way, operations are grouped into several types. The operations of a particular type can only be processed by several specific resources characterized by their processing capacity over a standard time-period.
The queueing network is next transformed into a planning model with continuous flows of products. This is obtained by summing up the elementary events of operation startings and completions over each period of the planning horizon. The
purpose of this transformation is to simplify the model while keeping its mean performance indices.

The planning problem can then be formulated as a problem of reachability of a reference state (or more exactly output) trajectory over a given time horizon. At each period of this horizon, some constraints on activity and inventory levels have to be satisfied. In order to solve this problem, an optimization criterion generally has to be formulated, and the corresponding optimality conditions are to be solved, often approximately.

This study focuses on the trajectory following problem, to obtain a smooth evolution of the system between the initial and the horizon time.

The multivariable linear representation of the system allows to easily deciding whether or not a target is reachable. If it is not, it can be modified so as to become reachable. A feasible production plan can then be generated under the constraints of the planning problem.

2 An ideal product-form queueing network model

Queueing networks can be used to describe many different production settings. They also provide analytical performance evaluation methods which are exact in some simple cases or which are approximate ([3], [6]). However, at first sight, they do not seem to provide appropriate models for production systems, because of the following features:

- The assumptions on which they are built do not generally seem to be realistic in this context: Poisson laws for arrivals, exponential processing times, FIFO (first in first out) priority policies
- They often require an extremely detailed description of the current state of the system: what are the available parts in front of each machine, in what order did they arrive, to what jobs do they belong...?
- They only allow to compute, at best, the system performance in its steady running state.

However, these seeming drawbacks are balanced by the following advantages:

- When the assumptions of BCMP networks [1] are satisfied, they provide a product-form solution from which the load of each machine can be evaluated in a decomposed way
- They help analyzing rather subtle running properties, such as blocking phenomena and for instance the ones due to capacity constraints.
- The evaluation that they provide are actually not very sensitive to the assumed probability distributions and priority rules. This robustness property has been the matter of many theoretical studies [18], [19]. It also sets up one of the foundations of some methods such as operational analysis [7], [5], perturbation analysis [13], mean value analysis (MVA) [17] and heavy traffic steady state approximations by continuous networks or fluid networks [16].

Consider a flexible manufacturing system made up with \( n \) different machines connected together by a parts transportation network. In an ideal open queueing net-
work, the system inputs are $R$ Poisson distributed random flows corresponding to
the different types of products to be manufactured. The mean arrival rates of each
flow during the period with index $k$, are denoted $u_r(k)$, for $r = 1, \ldots, R$, $k = 0, \ldots, N$,
where $N$ is a planning horizon.
The storage space for parts can either be local, in front of each machine, or common
to all of them. In both cases, it is supposed to be sufficiently large for the blocking
phenomena to be possibly neglected.

The process of grouping within the same job type the jobs with identical or similar
operating sequences makes up an aggregated and simplified representation of pro-
duction orders. In the case of many recurrent orders, the representation of jobs of
the same type as indistinguishable samples of the same stationary probability law,
agrees with the law of large numbers.
Similarly, the tasks to be performed on parts of different types can be grouped into
classes of similar operations. These classes define unified types of operations having
random processing times.
The manufacturing system can then be modelled as a multiclass queueing network.
The assumed queueing policy is the FIFO rule and the machine processing times
are supposed to be exponentially distributed with constant mean service rates, $\mu_i$,
$(i = 1, \ldots, n)$. Then the stationary distribution of the jobs in the network has a
product form [1].

Each job of type $r$ ($r = 1, \ldots, R$) is made of several operations which should be
processed in an order consistent with the specific precedence graph of this job type.
A job of a given type can generally be processed according to several different op-
erating sequences. Moreover, each operation can be processed on any machine of
an admissible set. One of them is to be chosen by the selected real-time assign-
ment rule. The policy for assigning sequences to jobs and machines to tasks can be
obtained as the results of an optimization program. In some previous works, two
possible techniques for implementing such policies have been studied:

- random drawings according to the optimal stationary flow distribution param-
eters [12].

An assignment policy can then be defined as the choice of a (possibly probabilistic)
sequencing rule of operations on machines for each job type. The present study is
not actually devoted to the construction of such an assignment policy, which would
describe in details the real-time piloting of the flows of products. Its main focus
is rather directed onto evaluating the average intensity of these flows. It will thus
be assumed that the problem of selecting a good assignment policy has been solved
off-line, for instance on the basis of an average load of the system.

Then, the evolution of type $r$-products in the network can be wholly described by a
probabilistic transition matrix between machines, $P^r = (P^r_{ij})$, which is supposed
to be irreducible.

In this context, the decision variables which will be considered are the rates of re-
lease of raw products at each period, $u_r(k)$, for $r \in (1, \ldots, R)$, $k \in (0, \ldots, N)$. The
system is modelled as an open multiclass queueing network. For constant rates
of release, it has a product-form and its current state is simply described by the
vector of the numbers of products (of all the types) queueing at each machine:
The total release rate for raw products in the FMS during period \( k \) is denoted \( \lambda_k \). It can be defined by:

\[
\lambda_k = \sum_{r=1}^{R} u_r(k)
\]

During this time period, the probability of release of a type \( r \)-job is:

\[
p_r(k) = \frac{u_r(k)}{\lambda_k}.
\]

The first operation of a job of type \( r \) is processed at machine \( i \) with probability \( t_{ir} \). Therefore, a job initially released in the FMS during period \( k \) belongs to type \( r \) and has its first operation processed at machine \( i \) with probability:

\[
\theta_{ir}(k) = p_r(k)t_{ir}.
\]

The average release rate of type \( r \)-jobs can also be written:

\[
u_r(k) = \frac{\lambda_k}{n} \sum_{i=1}^{n} \theta_{ir}(k)\]

\( u(k) \) denotes the vector of average release rates during period \( k \). It is defined by

\[
u(k) = (u_1(k), ..., u_R(k)).
\]

After processing at machine \( i \), type \( r \)-jobs are routed to machine \( j \in (1, .., n) \), with probability \( P_{r_{ij}} \) such that:

\[0 \leq P_{r_{ij}} \leq 1 \text{ for } j \in (1, .., n) ; \sum_{j=1}^{n} P_{r_{ij}} \leq 1.\]

A type \( r \) manufactured product is obtained after an operation at machine \( i \) with probability: \( Q^r_i = 1 - \sum_{j=1}^{n} P_{r_{ij}} \).

Let \( e_{ir}(k) \) be the relative arrival rate of jobs of type \( r \) at machine \( i \).

The traffic balance equation for each machine \( i \) at the equilibrium can be written as follows:

\[
e_{ir}(k) = \theta_{ir}(k) + \sum_{j=1}^{n} P_{r_{ij}} e_{jr}(k)
\]

The conditions for the network to be open relatively to all the types of products are:

\[
\sum_{i=1}^{n} Q^r_i > 0 \text{ for } r \in (1, ..R)
\]

They imply that matrix \( I - P^r \) is non-singular and that the set of equations (3) has a unique solution, which can be written as the following row-vector:

\[
e_r(k) = (e_{1r}(k), ..., e_{nr}(k)) = \theta_r(k)(I - P^r)^{-1}
\]

with, by definition,

\[
\theta_r(k) = (\theta_{1r}(k), ..., \theta_{nr}(k)) = p_r(k)t_r \text{ and } t_r = (t_{1r}, ..., t_{nr})
\]
Equation (5) can also be written:

\[ e_r(k) = p_r(k)t_r(I - P^r)^{-1} \]  (7)

It is worth noting that vector \( \eta_r = (\eta_{1r}, \ldots, \eta_{nr}) = t_r(I - P^r)^{-1} \) does not depend on the selected release policy during period \( k \). If the release policy satisfies the network stability requirement, the value of the mean arrival rate of type \( r \)-jobs at machine \( i \) during period \( k \) is, in quasi-static running conditions:

\[ \alpha_{ir}(k) = u_r(k)\eta_{ir} \]  (8)

The utilisation factor of machine \( i \) is:

\[ \rho_i(k) = \frac{\sum_{r=1}^{R} u_r(k)\eta_{ir}}{\mu_i} \]  (9)

A necessary condition for the network stability is:

\[ 0 \leq \rho_i(k) < 1 \]  (10)

Under the assumption of quasi-static running conditions, the product form of the equilibrium probability distribution during period \( k \) can be written:

\[ p^k(x(k)) = p^k_1(x_1(k)) \ldots p^k_n(x_n(k)) \]  (11)

Functions \( p^k_i(m) \) with \( i = 1, \ldots, n \) and \( m \in N \) give the equilibrium probabilities of the number of jobs waiting at each node (machine + input queue) of the network. They can be written:

\[ p^k_i(m) = \frac{(\rho_i(k))^m}{\sum_{l=0}^{\infty} (\rho_i(k))^l} = (1 - \rho_i(k))(\rho_i(k))^m \]  (12)

The average number of jobs waiting at machine \( i \) during period \( k \) is:

\[ E[x_i(k)] = \frac{\rho_i(k)}{1 - \rho_i(k)} \]  (13)

And, from Little’s formula [14], the mean sojourn time at machine \( i \) is:

\[ E[T_i(k)] = \frac{1}{\mu_i(1 - \rho_i(k))} \]  (14)

Under the release rates of period \( k \), the total mean sojourn time in the system for a type \( r \)-job can be written:

\[ E[T^r(k)] = \sum_{i=1}^{n} \eta_{ir} E[T_i(k)] \]  (15)

Thus, from relations (7), (14), the mean flow time of a type \( r \)-product can be expressed as follows:

\[ E[T^r(k)] = \sum_{i=1}^{n} \frac{\eta_{ir}}{\mu_i - \sum_{c=1}^{R} u_c(k)\eta_{ic}} \]  (16)
3 The model with continuous flows

The analysis of the last section was restricted to a single running period. The duration of such a period was supposed to be long enough, compared with the dynamics of the elementary processes. In this case, the equilibrium running conditions provide an adequate approximation of the real behaviour.

But in order to simply model the production process as a discrete-time system with delays, it now becomes necessary to relax this assumption. The system lags for the various types of products are now supposed to be of the same magnitude order than the selected sampling period. The previous assumption of two different time scales is then clearly violated.

However, we can still assume that the load variations of the system are sufficiently smooth for the quasi-static flow approximation to remain valid. In agreement with this new assumption, and so as to correctly distribute the work load over space and time, one of the objectives of the production planning problem will be to smooth the production over all the periods of the planning horizon.

Consider now the system dynamics over the periods of the planning horizon. Let $T$ be the duration of each of these periods. The size of $T$ may vary from one hour to one week, depending on the types of products and on the size of production lots.

The release of type $r$-jobs with the mean rate $u_r(k)$ during period $k$ generates a mean production rate with the required quality, $\beta_r u_r(k)$ after a mean time delay given by relation (16). $\beta_r$ is a parameter measuring the average ratio of type $r$-products which meet the specifications. Poor quality products are supposed to be cleared out, not recycled.

The distribution of the delay between the instants of release and of completion of a type $r$-job is denoted $T_r(k)$. It describes the system response to a step input of size $u_r(1)$ at period 1, while the release policy related to the other job types is supposed to be fixed. For a release policy $\pi$, the mean production rate of type $r$-products during period $k$ can then be written as a discrete-time convolution product:

$$E_\pi[y_r(k)] = \eta_r \sum_{t=0}^{\infty} u_r(k-t)\text{Prob}_\pi\{t \leq T_r(k-t) < t+1\}$$ (17)

Formally, the relation above shows that for a linear-type input-output behaviour, the impulse response of the system is described by the coefficients: $\text{Prob}_\pi\{t \leq T_r(k-t) < t+1\}$. But it should be noticed that these coefficients are functions of vectors $u_c(l)$ for $c \in (1, .., R)$ and $l \leq k$. In this model, the correlations between the various production flows, and the non-linear properties of the response appear through these conditional probabilities.

But if we assume that these coefficients have slow variations, the system behaviour can be approximately described by the following discrete-time linear dynamics:

$$Y_k = B(q^{-1})U_k + \epsilon(k)$$ (18)

$T$ is the basic planning period,

$y'(k) = (y_1(k), ..., y_R(k))$ the row-vector of mean production rates,

$Y_k \in \mathbb{R}^R$, $Y'_k = (Y_1(k), ..., Y_R(k)) = T y'(k)$.

$U_k \in \mathbb{R}^R$, $U'_k = (U_1(k), ..., U_R(k)) = T u'(k)$.

$\epsilon'(k) = (\epsilon_1(k), ..., \epsilon_R(k))$ is a vector of white noises such that, by assumption,
\[ \xi_k = (1 - q^{-1}) \epsilon_k \] is also a vector of white noises. 

\[ q^{-1} \] is the delay operator and \( B(q^{-1}) \) a polynomial matrix of this operator:

\[ B(q^{-1}) = B_0 + q^{-1}B_1 + ... + q^{-p}B_p \]

\( B_0, B_1, ..., B_p \) are scalar matrices in \( \mathbb{R}^{R \times R} \).

In this model, the selected outputs are the quantities produced at each period. Other interesting data are the inventory levels at the end of each period, \( Z_r(k) \) for \( r = 1, ..., R; \ k = 1, ... \). They are defined by

\[ Z_r(k) = \max \{0, Z_r(k-1) + Y_r(k) - D_r(k)\} \]

\( D_r(k) \) is the demand for product \( r \) at period \( k \).

If we now assume that demands can always be totally satisfied, the new input-output equation can be re-written as follows:

\[ Z_k = Z_{k-1} + B(q^{-1})U_k - D_k + \epsilon(k) \] (19)

with \( Z'_k = (Z_1(k), ..., Z_R(k)) \) and \( D'_k = (D_1(k), ..., D_R(k)) \).

The coefficients of matrix \( B(q^{-1}) \) are functions of the loading levels of the network. The evolution of these levels is supposed to be slow. And the moments of the probability laws are supposed to have a bounded evolution; that is to say that the system is only considered in its stability region defined by inequalities \( \rho_i(k) < 1 \ \forall i, \ \forall k \).

The coefficients of matrix \( B(q^{-1}) \) have slow time-variations. They can be estimated from the average performance of the system which can be evaluated from the queueing network model described in the previous section. The parameters of model (18) can also be recursively estimated from direct measurements on the system.

By this scheme, a continuous-time detailed model with discrete inputs and outputs has been transformed into an integrated and simplified discrete-time model with continuous inputs and outputs. The correspondence between these two models rests on the mean values of flows at the equilibrium and on the mean delays between the release of a raw item in the system and its completion as a corresponding manufactured product. The second model is much simpler than the first one. It is possible to apply on it many different techniques for performance evaluation, optimization and control. In particular, this model is specially well fitted to the solving of these problems by classical techniques.

Because of these properties, it is an interesting mathematical tool for the production planning problem. From such a representation of the production process, it is easy to build an optimal release policy of the various product-types, under the choice of a suitable performance criterion.

A basic difference between the proposed model and the hierarchical planning models ([15], [8]) can be noticed. Here, the use of time integration is combined with the aggregation of product flows. Feasibility and robustness problems are made more complex by this change of time scale.
4 An optimal control approach to the planning problem

On the basis of the model with continuous flows defined in the preceding section, the planning problem can be formulated as the problem of following an optimal output trajectory over a given time-horizon. At each period, the outputs are the quantities of currently completed products, while feasibility and stability constraints have to be satisfied.

An efficient technique for solving this trajectory following problem is the algorithm of generalized predictive control (GPC) [4], extended to the multivariable case.

The criterion to be minimized over the planning horizon is quadratic. It contains two terms:

- a penalty term on the differences between the vectors of future outputs, \( Y_k \)
  and the vectors of reference outputs, \( Y_k^M \)
- a weighted function of the differences between successive controls.

Such a generalized predictive criterion can be written:

\[
J = E[\sum_{l=0}^{N} \{(Y_{k+l} - Y_{k+l}^M)'Q_l(Y_{k+l} - Y_{k+l}^M) + (U_{k+l} - U_{k+l-1})'R_l(U_{k+l} - U_{k+l-1})\}] \tag{20}
\]

with matrices \( Q_l \in \mathbb{R}^{n \times n} \) and \( R_l \in \mathbb{R}^{n \times n} \) non-negative definite.

This criterion can be re-written as follows:

\[
J = E[(\tilde{Y}_k - \tilde{Y}_k^M)'Q(\tilde{Y}_k - \tilde{Y}_k^M) + (\Delta U_k)'R(\Delta U_k)] \tag{21}
\]

with \( Q = \text{Diag}(Q_1, ..., Q_N) \), \( R = \text{Diag}(R_1, ..., R_N) \),

\( \Delta U_k' = (U_k - U_{k-1}, ..., U_{k+N} - U_{k+N-1}) \), \( \tilde{Y}_k' = (Y_k, ..., Y_{k+N}) \), \( \tilde{Y}_k^M' = (Y_k^M, ..., Y_{k+N}^M) \).

Using the input-output model (18) for the process, the generalized predictive control law corresponding to this criterion can be written:

\[
\Delta U_k = (G'QG + R)^{-1}G'Q(\tilde{Y}_k^M - F_k) \tag{22}
\]

with, for \( p < N \),

\[
G = \begin{bmatrix}
B_0 & 0 & 0 & \cdots & 0 \\
B_1 + B_0 & B_0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
B_p + \ldots + B_0 & B_1 + B_0 & B_0 & \ldots & B_0 \\
0 & 0 & \ldots & 0 & B_p + \ldots + B_0
\end{bmatrix}
\]

and \( F_k = \begin{bmatrix}
B_0 + \ldots + B_p & \ldots & \ldots & \ldots & B_0 + B_1 \\
0 & B_0 + \ldots + B_p & \ldots & \ldots & B_0 + B_1 + B_2 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \ldots & 0 & B_0 + \ldots + B_p \\
0 & 0 & \ldots & 0 & 0
\end{bmatrix}\begin{bmatrix}
\Delta U_{k-p} \\
\Delta U_{k-1}
\end{bmatrix} \]
Furthermore, it has been shown [2] that it is possible to guarantee that constraints on the control vector are satisfied by introducing in the criterion some penalty terms. These terms are iteratively updated to satisfy the associated Kuhn-Tucker conditions.

The constraints related to model (18) are obtained from the detailed model through aggregation and integration. Thus, they are necessary but not sufficient for generating feasible schedules. It can be noticed however that under the probabilistic assumptions of the detailed model, it is not possible to strictly satisfy any deterministic constraint on due dates.

Consider the case of an aggregated plan which is not totally feasible for the detailed model. Then, feasibility may be obtained by moving some operations from one or several periods to the following ones. And this would lead to a worse value of the criterion. But in any case, the real job release policy for products or lots should satisfy the local operational constraints of non-overlapping and precedence.

5 Conclusion

The main purpose of this paper was to describe a dynamical model well fitted to the production planning problem. Such a model was built from a detailed model of the FMS in working conditions. The detailed model is a queueing network representation. It can be used for constructing real-time scheduling policies, based for instance on the evolution toward an optimal steady-state [12], or on some priority rule using a comparison between dynamic allocation indices [10], [11].

It is clear that consistency of the various models used for decision making is a basic condition for the setting up of an integrated management system. Therefore, the construction of such a planning model can be seen as a prime stage in the integration of production decisions.

The generalized predictive control algorithm presented in section 4 is purely illustrative. It should certainly not be seen as a general planning technique. But it has some advantages in terms of simplicity, of efficiency for prediction and trajectory following and of flexibility in the choice of the preponderant terms in the criterion. However, it is specially well-fitted to the case of relatively regular productions. And, on the other hand, it could hardly be used, for instance, in the case of release policies with large lot sizes.

References


