Optimal Control of Production Systems with Unreliable Machines and Finite Buffers*

Fabio Balduzzi (†), Giuseppe Menga (†), Alessandro Giua (‡)

(†) Dip. di Automatica ed Informatica, Politecnico di Torino, Italy
Email: {balduzzi,menga}@polito.it

(‡) Dip. di Ingegneria Elettrica ed Elettronica, Università di Cagliari, Italy
Email: giua@diee.unica.it

Abstract

In this paper we present a novel formulation for the optimal control of discrete event dynamic processes which represent production systems with unreliable machines and buffers of finite capacity. We derive an optimum control strategy that is critically based on the fact that the discrete event dynamic behavior of the system is approximatively represented with a hybrid model. We introduce a deterministic fluid network model where the average flow rates through the machines are the control variables and with an original approach we show that the dynamics of the system easily translate into a discrete–time, time–varying state variable model, where the optimum machine production rates can be easily obtained by solving a sequence of linear programming problems.

1 Introduction

In this paper we consider manufacturing systems consisting of unreliable machines and buffers of finite capacity in the most general multi–class multi–machine setting. The process allows the simultaneous production of several part types (classes) with general service time distributions and routing policies. Each part has to perform its own orderly sequence of operations (production cycle), different for each class, in order to be completed. The production cycle specifies for each part the sequence of machines it must visit and the operation performed by them. The same machine can perform operations on different part classes (sequencing), eventually with different service times. The same operation can be performed on alternative machines (routing).

Machine breakdowns, planned and unplanned maintenance, operator unavailability, setup times, etc., make the manufacturing environment stochastic. Therefore the general problem setting is

*Published as: F. Balduzzi, G. Menga, A. Giua, "Optimal Control of Production Systems with Unreliable Machines and Finite Buffers," *Proc. 1999 IEEE Int. Conf. on Robotics & Automation* (Detroit, Michigan), pp. 1462-1468, May 1999.

a stochastic discrete event dynamic process. In this paper we present a preliminary result where we assume deterministic fail and repair events, such as in the presence of planned maintenance programs, and we assume that the uncertainty of the service times is small enough so that it does not affect the sequence of a limited number of events that we call *macro–events*.

We derive an optimum control strategy that is critically based on the fact that the discrete event dynamic behavior of the production system is approximatively represented with a hybrid model. This model is characterized by two layers of aggregation. At the lower layer the behavior of the system is approximated by the average flow of each part through each machine. At the higher layer a discrete event model will represent the transitions of the system through a sequence of operational states, that we call *macro–states*, at the occurrence of the macro–events (machine starvation, blockage, breakdown and repair, material release times and due dates). The control is composed of two parts: a dispatcher and a planner. During the permanence into each macro–state, between the occurrence of two consecutive macro–events, the dispatcher will take care of the sequencing of parts waiting in the buffers and of the routing of them at the exit of the machines according to their class. This will be done in order to accomplish average flows according to a planner which implements an optimum control strategy over a finite time horizon that will be discussed in this paper.

By this approximation approach, described in [2] and briefly outlined in the next section, the discrete event dynamics of the system easily translate into a discrete–time, time–varying state variable model, where the average flows of parts through the machines enter the model as non–linear control variables.

The optimum control problem of maximizing productivity over a finite time horizon while guaranteeing at the final time a desired production mix, has a necessary condition which lends itself to the evaluation of the optimum machine production rates for each sample period through a sequence of linear programming problems. This result is particularly interesting because it formally confirms the intuition resulting from the myopic approach described in [2] where each sample period is treated independently from the others. Moreover the resulting control policy easily allows the computation of the sensitivity functions of the optimum control system with respect to the design parameters, i.e., buffer capacity, maintenance period and machine productivity, to be used during the design of the manufacturing system.

A multi-class discrete production is usually seen as the problem of scheduling parts to the machines subject to release times and due dates for a set of lots of different classes while optimizing a certain performance index. If the production model is approximated by a fluid network model, individual parts in a lot do not exist anymore, and the average flow rates through the machines for the different lots become the decision variables. A discrete time optimum control problem can be formalized where release times and due dates can be easily accounted for by introducing appropriate constraints in the state variable model at the occurrence of certain macro–events. In this paper we present the technique for the special case of constraining the production mix at the final time.

1.1 Previous Work

Optimal control formulations have been developed and applied to a number of automated manufacturing systems. Gershwin [6] developed a general model suggesting a hierarchical approach for scheduling and planning. Sethi and Zhou [9] established a graph–theoretic framework for a dynamic jobshop that describes the system dynamics along with state and control constraints. In [8] Presman *et al.* considered the problem of choosing the production rates of an N–machine flowshop by formulating a stochastic dynamic programming problem. Akella and Kumar [1] proposed an optimal control of a continuous–time system with jump Markov disturbances and with an infinite horizon discounted cost criterion for an unreliable manufacturing system producing a single class of product that has to meet a given demand. Yao *et al.* [5] considered the problem of scheduling manufacturing systems based on a deterministic fluid network model. In a recent work Balduzzi and Menga [2] developed a discrete–time, time–varying linear stochastic state variable model for the fluid approximation of flexible manufacturing systems. Then, by using perturbation analysis techniques they obtained average values and variances of both performance measures and their gradients with respect to the system parameters to perform optimal design of the system configuration. The results presented in this paper follow the model developed in [2].

2 Description of the Model

The production process considered in this work consists of a set of n single–server stations, denoted by M_i, for $i = 1, \ldots, n$, serving ℓ classes of products, indexed by $r = 1, \ldots, \ell$. Parts of different classes move from machine M_i to M_j according to their production cycle and are queued in buffers, one for each machine, with the initial one (input buffer) acting as an unlimited supply of parts and the final buffer acting as a limited storage area for collecting finished products, thus representing the production target. The buffers have finite capacity C_i and the machines are unreliable.

We consider operation-dependent failures and we define for each machine the *production volumes before a machine fails* and the *repair times*, denoted by the sequences $w_i = (w_{i,1}, w_{i,2}, \ldots)$ and $f_i = (f_{i,1}, f_{i,2}, \ldots)$, for $i = 1, \ldots, n$, respectively. In a general problem setting these sequences have been considered as stochastic processes [2]. In this paper a preliminary result is approached and these sequences are assumed deterministic values as in the case of planned maintenance programs. Machine service times are assumed independent random variables with identical distribution with finite mean and variance. The maximum average production rate for parts of class r of machine M_i is denoted V_i^r.

The evolution in time of the production process is discussed within a framework that distinguishes two levels of aggregation. The lower layer represents the microscopic behavior of arrivals and departures of parts to/from each machine (micro–events). It will be modeled in an aggregate view by using first order fluid approximations [7]. At the higher layer a discrete event model will represent the transitions of the process through a sequence of macro–states, at the occurrence of the macro–events.

In [3] the authors have shown for the single–class multi–machine case that it is possible to make an efficient analysis of this hybrid system by defining a suitable *discrete linear inclusion* that describes the transitions of the associated finite automata through a sequence of admissible macro–states, and thus studying the stability of the production process.

2.1 The Microscopic Layer: Fluid Model

Let $\tau_k = [t_k, t_{k+1})$, for $k = 0, 1, 2, \ldots$, be the interval of time between the occurrence of consecutive macro–events at time t_k and t_{k+1}, that we call *macro–period*, and let $v^r_{i,j}(k)$ be the constant average flow rates of parts of class r from machines M_i to M_j.

The microscopic behavior of a production system during a macro–period can be approximated by the following three processes defined for each machine M_i:
(I) the buffer levels for parts of class r

$$x^r_i(t) = x^r_i(t_k) + [v^r_{in,i}(k) - v^r_{out,i}(k)](t - t_k), \tag{1}$$

(II) the production volume processed by the machine since the last repair (used to evaluate the machine breaking time)

$$\chi_i(t) = \chi_i(t_k) + v_{out,i}(k)(t - t_k), \tag{2}$$

(III) the time spent by the machine under repair since the last failure

$$s_i(t) = s_i(t_k) + (t - t_k) \tag{3}$$

where $t \in [t_k, t_{k+1})$ for all these processes, and

$$\begin{array}{ll} v^r_{in,i}(k) = \sum_h v^r_{h,i}(k), & v_{in,i}(k) = \sum_r v^r_{in,i}(k), \\ v^r_{out,i}(k) = \sum_j v^r_{i,j}(k), & v_{out,i}(k) = \sum_r v^r_{out,i}(k) \end{array}$$

are the inflow and outflow rates of parts of each machine. We also define the process $p^r_i(t)$ that represents the cumulative production of parts of class r at the exit of each machine M_i as

$$p^r_i(t) = p^r_i(t_k) + v^r_{out,i}(k)(t - t_k), \quad t \in [t_k, t_{k+1}). \tag{4}$$

Note that Equations (2) and (4) are similar but while $\chi_i(t)$ will be reset to 0 after each failure, $p^r_i(t)$ will keep on accounting for the cumulative volume currently processed by the machine.

2.2 The Macroscopic Layer: State Variable Model

At the macroscopic level the evolution in time of the system through a sequence of macro–states can be described by a finite automata with states given by a finite set of admissible configurations of machines status (*operational* or *down*) and buffer status (*full*, *not full–not empty*, *empty for parts of class* r) and with transitions represented by the macro–events (*failure*, *repair*, *buffer full* and *buffer empty for parts of class* r).

The transitions of the finite automata define the interlacing of Equations (1)–(3) and drive the evolution of the system whose macro–behavior can be described by the following discrete–time, time–varying state variable model:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\big(k, \mathbf{u}(k)\big)\mathbf{x}(k) + \mathbf{b}\big(k, \mathbf{u}(k)\big)d(k+1) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \tag{5}$$

whose samples are the occurrence of the macro events at time t_k, for $k = 0, 1, 2, \ldots$. The state variable vector is

$$\mathbf{x}(k) = [\ldots, x_i^r(t_k), \chi_i(t_k), s_i(t_k), \ldots]^T \tag{6}$$

with entries given by the values of the processes (1)–(3) at the occurrence of the macro–events. The control input vector is

$$\mathbf{u}(k) = [\ldots, v_{i,j}^r(k), \ldots]^T \tag{7}$$

with entries given by the constant average flow rates. The disturbance input sequence $d(k)$ is a deterministic sequence of values 0, C_i, $w_{i,\alpha} \in w_i$, $f_{i,\beta} \in f_i$, for $i = 1, \ldots, n$, and α, $\beta \in \mathbb{N}^+$. With this model any macro–event happens when an appropriate component of the state vector reaches a specified value given by the sequence $d(k)$. The detailed derivation of this state variable model can be found in Appendix A.

Note that each macro–state defines the feasible region, denoted by $\mathcal{U}(k)$, for the average machine production rates $v_{i,j}^r(k)$ that enter model (5) as non–linear control variables.

We define an auxiliary system linked with model (5) that will be used in the optimization problem introduced in the next section:

$$\begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{B}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big) \\ \mathbf{p}(0) = \mathbf{p}_0 \\ \mathbf{y}(k) = \mathbf{C}\mathbf{p}(k) \end{cases} \tag{8}$$

where $\mathbf{p}(k) = [\ldots, p_i^r(t_k), \ldots]^T$ describes the cumulative production of parts, i.e., its entries are given by the values of (4) at the occurrence of the macro–events. Matrices $\mathbf{B}$ and $\mathbf{C}$ are of appropriate dimensions and have entries 0 or 1. The scalar function $\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big) = \big(t_{k+1} - t_k\big)$ is defined by Eq. (5) (see Appendix A for details) and $\mathbf{y}(k)$ is a vector of dimension ℓ whose components represent the level of each part class in the output buffer, i.e., the cumulative production of finished parts.

We indicate with $I_o(k)$ and $I_d(k)$ the sets of indices of operational and down machines, $I_f(k)$ and $I_e(k)$ the sets of indices of full and empty buffers for parts of class $1, \ldots, r \leq \ell$, during the k–th macro–period, respectively.

Definition 2.1. A control $\mathbf{u}(k) \in \mathcal{U}(k)$ is *admissible* if it is a feasible solution of the following set of linear inequalities:

$$\begin{cases} (a) & 0 \leq \sum_j v_{i,j}^r(k) \leq V_i^r, \quad \forall i \in I_o(k) \\ (b) & \sum_r \sum_j v_{i,j}^r(k) = 0, \quad \forall i \in I_d(k) \\ (c) & \sum_r \sum_h v_{h,i}^r(k) \leq \sum_r \sum_j v_{i,j}^r(k), \\ & \forall i \in I_f(k) \\ (d) & \begin{cases} \sum_j v_{i,j}^1(k) \leq \sum_h v_{h,i}^1(k) \\ \ldots \\ \sum_j v_{i,j}^r(k) \leq \sum_h v_{h,i}^r(k) \end{cases} \\ & \forall i \in I_e(k) \\ v_{i,j}^r \geq 0 & \end{cases} \tag{9}$$

The *consistency constraint set* (CCS) (9) will be denoted $\mathbf{g}(k, \mathbf{u}(k)) \leq \mathbf{0}$. ■

Constraints of the form (9.a) bound the machine production rates at their maximum values and apply for all operational machines. Constraints of the form (9.b) apply for all machines under repairing. Constraints of the form (9.c) have to be satisfied for all machines with full buffers and constraints (9.d) for all machines whose buffer level for parts of class $1, \ldots, r \leq \ell$ is 0.

The region $\mathcal{U}(k)$ defined by $\mathbf{g}(k, \mathbf{u}(k)) \leq 0$ is a convex polyhedron whose vertices are basic solutions of any linear programming problem with objective function of the form $J = \mathbf{a}^T(k)\mathbf{u}(k)$ and subject to the CCS. Any admissible control policy $\mathbf{u}(k)$ corresponds to a point within the feasible region $\mathcal{U}(k)$ and the boundary represents all those control policies aimed at maximizing a given linear objective function. Thus the optimum solution denoted $\mathbf{u}^o(k)$ will always lay on the boundary of the feasible region.

3 The Dynamic Control Problem

The dynamic control policy adopted in this work provides the machine production rates $\mathbf{u}(k)$ as the solution of an optimal control problem aimed at maximizing the productivity of the system while guaranteeing a given production mix at the final event.

Let N be the final event, $T = \cup_{k=0}^{N-1}[t_k, t_{k+1}]$ the finite time horizon, and

$$y_{tot}(k) = \mathbf{1}^T \mathbf{C}\mathbf{p}(k) = \sum_{r=1}^{\ell} y_r(k) \tag{10}$$

the cumulative production of finished parts over all classes up to the time horizon $[0, t_k]$.

We now introduce our definition of *production mix.*

Definition 3.1. The production $\mathbf{y}(k)$ satisfies the production mix $\mathbf{m} = [m^1, \ldots, m^\ell]^T$ if:

$$\begin{cases} y_1(k) = m^1 y_{tot}(k) \\ \cdots \\ y_\ell(k) = m^\ell y_{tot}(k) \end{cases} \Leftrightarrow \ (\mathbf{M} - \mathbf{I})\mathbf{C}\mathbf{p}(k) = \mathbf{0} \tag{11}$$

and $\sum_{r=1}^{\ell} m^r = 1$. ■

Here m^r denotes the mix factor of parts of class r and $\mathbf{M} = \mathbf{m} \cdot \mathbf{1}^T$ is a real matrix of dimension $\ell \times \ell$.

The optimum control problem is to determine the control sequence $u^o = \big(\mathbf{u}^o(0), \ldots, \mathbf{u}^o(N-1)\big)$, for $\mathbf{u}^o(k) \in \mathcal{U}(k)$, which maximizes the performance functional $J(u) = y_{tot}(N)$ over the finite time horizon T. This problem can be formulated as follows:

$$\begin{aligned} &\max_u \ y_{tot}(N) = \min_u \left[-\mathbf{1}^T \mathbf{C}\mathbf{p}(N)\right] \qquad \text{subject to:} \\ &(a) \begin{cases} \mathbf{p}(k+1) = \mathbf{p}(k) + \mathbf{B}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big) \\ \mathbf{p}(0) = \mathbf{p}_0 \end{cases} \\ &(b) \begin{cases} \mathbf{x}(k+1) = \mathbf{D}(k)\big[\mathbf{x}(k) + \mathbf{R}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big)\big] \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \\ &(c) \ \ \mathbf{u}(k) \in \mathcal{U}(k), \quad \forall k = 0, \ldots, N-1 \\ &(d) \ \ (\mathbf{M} - \mathbf{I})\mathbf{C}\mathbf{p}(N) = \mathbf{0} \end{aligned} \tag{12}$$

Equations (12.a) and (12.b) represent the dynamics of the systems (8) and (5). Equation (12.c) is the control constraints (9) and Equation (12.d) is the terminal condition of an assigned production mix at the final event. Note that we do not aim at guaranteeing the assigned mix at each macro–event.

To solve this problem, we introduce the *Hamiltonian* sequence

$$\begin{aligned} H_k = &\boldsymbol{\lambda}^T(k+1)\big[\mathbf{p}(k) + \mathbf{B}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big)\big] \\ &+ \boldsymbol{\mu}^T(k+1)\big\{\mathbf{D}(k)\big[\mathbf{x}(k) + \mathbf{R}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}(k)\big)\big]\big\} \end{aligned}$$

where $\boldsymbol{\lambda}(k)$ and $\boldsymbol{\mu}(k)$ are the costate vectors, and the scalar function

$$\Phi = \boldsymbol{\nu}^T(\mathbf{M} - \mathbf{I})\mathbf{C}\mathbf{p}(N) - \mathbf{1}^T\mathbf{C}\mathbf{p}(N)$$

with a set of ℓ multipliers $\boldsymbol{\nu}^T = (\nu_1, \dots, \nu_\ell)$. Let us assume the existence of optimal trajectories $p^o = \big(\mathbf{p}^o(0), \dots, \mathbf{p}^o(N-1)\big)$ and $x^o = \big(\mathbf{x}^o(0), \dots, \mathbf{x}^o(N-1)\big)$ and corresponding control sequence u^o, that satisfy a number of technical assumptions. Then, for the *maximum principle* [4], in order that u^o be optimal, there exists a costate vector $\boldsymbol{\lambda}(k)$, such that $\boldsymbol{\lambda}(k)$ and $\mathbf{p}^o(k)$ are a solution of the system

$$\begin{cases} \mathbf{p}^o(k+1) = \mathbf{p}^o(k) + \mathbf{B}\,\mathbf{u}^o(k)\Delta\big(\mathbf{u}^o(k), \mathbf{x}^o(k)\big) \\ \boldsymbol{\lambda}^T(k) = \boldsymbol{\lambda}^T(k+1) \quad \Rightarrow \quad \boldsymbol{\lambda}^T(k) = \boldsymbol{\lambda}^T \end{cases}$$

with boundary conditions

$$\begin{cases} \boldsymbol{\lambda}^T(N) = \dfrac{\partial \Phi}{\partial \mathbf{p}^o(N)} = \big[\boldsymbol{\nu}^T(\mathbf{M} - \mathbf{I}) - \mathbf{1}^T\big]\mathbf{C} \\ \mathbf{p}^o(0) = \mathbf{p}_0 \end{cases}$$

and there exists a costate vector $\boldsymbol{\mu}(k)$, such that $\boldsymbol{\mu}(k)$ and $\mathbf{x}^o(k)$ are a solution of the system

$$\begin{cases} \mathbf{x}^o(k+1) = \mathbf{D}(k)\big[\mathbf{x}^o(k) + \mathbf{R}\,\mathbf{u}^o(k)\Delta\big(\mathbf{u}^o(k), \mathbf{x}^o(k)\big)\big] \\ \boldsymbol{\mu}^T(k) = \boldsymbol{\mu}^T(k+1)\mathbf{D}(k)\big[\mathbf{I} + \mathbf{R}\,\mathbf{u}^o(k) \\ \qquad\qquad \cdot \nabla_{\mathbf{x}^o(k)}[\Delta(\mathbf{u}^o(k), \mathbf{x}^o(k))]\big] \end{cases}$$

with boundary conditions

$$\begin{cases} \boldsymbol{\mu}^T(N) = \dfrac{\partial \Phi}{\partial \mathbf{x}^o(N)} = \mathbf{0} \\ \mathbf{p}^o(0) = \mathbf{p}_0 \end{cases}$$

and such that H_k is minimized, i.e.,

$$\begin{aligned} H_k^o = &\min_{\mathbf{u}(k)}\big\{\boldsymbol{\lambda}^T\left[\mathbf{p}^o(k) + \mathbf{B}\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}^o(k)\big)\right] \\ &+ \boldsymbol{\mu}^T(k)\mathbf{D}(k)\big[\mathbf{x}^o(k) + \mathbf{R}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k), \mathbf{x}^o(k)\big)\big]\big\} \\ &\text{subject to:} \qquad \mathbf{g}(\mathbf{u}(k)) \le \mathbf{0} \end{aligned}$$

for $k = 0, \dots, N-1$. Note that this problem is a non–linear programming problem with linear constraints, which is difficult to solve analytically even if there exists several tools for dealing with this kind of problem, e.g., active sets, primal, and gradient projection methods. In the next section we will consider a linearized version of this problem.

This class of optimal control formulation has been applied to a number of flexible manufacturing systems, e.g., Akella and Kumar [1]. The major difficulty is that there is no available technique to solve this control problem analytically. However, as it will be clear in the following sections, a general finding is that the optimal control sequence u^o can be seen as a switching policy over the control variables. Control takes place when the macro–events do occur. Each solution $\mathbf{u}^o(k)$ corresponds to an extreme point of the CCS, i.e., the desirable operating point, and the controller always attempts to drive the system there and keep it.

4 The Linear Optimization Problem

Since problem (12) is analytically untractable, we consider a linearized version. Let us assume the existence of an optimum solution: p^o and x^o are the optimal trajectories and u^o is the corresponding control sequence for the problem (12).

We consider perturbations from the optimal trajectories p^o and x^o produced by admissible infinitesimal perturbations $\delta\mathbf{u}(k)$ in the optimal control sequence that, in turn, gives rise to the perturbations $\delta\mathbf{p}(k)$ and $\delta\mathbf{x}(k)$ obtained by linearizing (8) and (5) around their optimal trajectories. Furthermore we assume that the perturbation $\delta\mathbf{u}(k)$ is small enough so that is does not change the sequence of the macro–events. Thus we can write the two linear perturbed system dynamics as follows:

$$\begin{cases} \tilde{\mathbf{p}}(k+1) = \tilde{\mathbf{p}}(k) + \mathbf{A}_{px}(k)\tilde{\mathbf{x}}(k) + \mathbf{B}_p(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_p(k) \\ \tilde{\mathbf{p}}(0) = \mathbf{p}_0 \\ \tilde{\mathbf{y}}(k) = \mathbf{C}\tilde{\mathbf{p}}(k) \end{cases}$$

and

$$\begin{cases} \tilde{\mathbf{p}}(k+1) = \mathbf{A}_{xx}(k)\tilde{\mathbf{x}}(k) + \mathbf{B}_x(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_x(k) \\ \tilde{\mathbf{p}}(0) = \mathbf{x}_0 \end{cases}$$

where $\tilde{\mathbf{p}}(k) = \mathbf{p}^o(k) + \delta\mathbf{p}(k)$, $\tilde{\mathbf{x}}(k) = \mathbf{x}^o(k) + \delta\mathbf{x}(k)$ and $\tilde{\mathbf{u}}(k) = \mathbf{u}^o(k) + \delta\mathbf{u}(k)$. The complete derivation of these models can be found in Appendix B.

Now the optimal control problem is to find the control sequence $\tilde{u}^o = \big(\tilde{\mathbf{u}}^o(0), \ldots, \tilde{\mathbf{u}}^o(N-1)\big)$, for $\tilde{\mathbf{u}}^o(k) \in \mathcal{U}(k)$, which maximizes the performance functional $J(\tilde{u}) = \tilde{y}_{tot}(N)$ over the finite time horizon T. This optimization problem can be formulated as follows:

$$\begin{aligned} &\max_{\tilde{u}} \tilde{y}_{tot}(N) = \min_{\tilde{u}} \left[-\mathbf{1}^T\mathbf{C}\tilde{\mathbf{p}}(N)\right] \qquad \text{subject to:} \\ &\begin{cases} \tilde{\mathbf{p}}(k+1) = \tilde{\mathbf{p}}(k) + \mathbf{A}_{px}(k)\tilde{\mathbf{p}}(k) + \mathbf{B}_p(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_p(k) \\ \tilde{\mathbf{p}}(0) = \mathbf{p}_0 \end{cases} \\ &\begin{cases} \tilde{\mathbf{p}}(k+1) = \mathbf{A}_{xx}(k)\tilde{\mathbf{x}}(k) + \mathbf{B}_x(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_x(k) \\ \tilde{\mathbf{p}}(0) = \mathbf{x}_0 \end{cases} \\ &\tilde{\mathbf{u}}(k) \in \mathcal{U}(k), \quad \forall k = 0, \ldots, N-1 \\ &(\mathbf{M} - \mathbf{I})\mathbf{C}\tilde{\mathbf{p}}(N) = \mathbf{0} \end{aligned} \tag{13}$$

We observe that the objective function and the constraints are linear in the states and control variables. Thus we are dealing with a linear programming problem and the optimum solution, if it exists, will always require the control variables to be laying on the boundary of the feasible region. This approach can be seen as an extension of the *bang–bang principle* [4] to the multi–dimensional case.

4.1 Optimum Control: a Linear Programming Approach

An optimum solution of the problem (13) can be obtained by solving a sequence of linear programming problems, one for each sample period. By the same developments as in Section 3 we introduce the *Hamiltonian* sequence

$$H_k = \boldsymbol{\lambda}^T(k)\big[\mathbf{A}_\lambda(k)\mathbf{z}(k) + \mathbf{B}_\lambda(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_\lambda(k)\big] \tag{14}$$

where we have defined

$$\mathbf{z}(k) = \begin{bmatrix} \tilde{\mathbf{p}}(k) \\ \tilde{\mathbf{p}}(k) \end{bmatrix}, \quad \mathbf{A}_\lambda(k) = \begin{bmatrix} \mathbf{I} & \mathbf{A}_{px}(k) \\ \mathbf{0} & \mathbf{A}_{xx}(k) \end{bmatrix},$$
$$\mathbf{B}_\lambda(k) = \begin{bmatrix} \mathbf{B}_p(k) \\ \mathbf{B}_x(k) \end{bmatrix}, \quad \mathbf{G}_\lambda(k) = \begin{bmatrix} \mathbf{G}_p(k) \\ \mathbf{G}_x(k) \end{bmatrix},$$

and the scalar function

$$\Phi = -\mathbf{1}^T\mathbf{C}\tilde{\mathbf{p}}(N) + \boldsymbol{\nu}^T\big(\mathbf{M} - \mathbf{I}\big)\mathbf{C}\tilde{\mathbf{p}}(N). \tag{15}$$

The costate dynamics is given by

$$\begin{cases} \boldsymbol{\lambda}^T(k) = \boldsymbol{\lambda}^T(k+1)\mathbf{A}_\lambda(k) \\ \boldsymbol{\lambda}^T(N) = \big[\boldsymbol{\nu}^T(\mathbf{M} - \mathbf{I}) - \mathbf{1}^T\big]\mathbf{C} \end{cases} \tag{16}$$

which yields

$$\boldsymbol{\lambda}(k) = \mathbf{A}_\lambda^T(k)\cdots\mathbf{A}_\lambda^T(N-1)\big[\boldsymbol{\nu}^T(\mathbf{M} - \mathbf{I}) - \mathbf{1}^T\big]\mathbf{C}.$$

The optimal solution of problem (13) can be obtained by computing the optimal values H_k^o, for $k = 0, \ldots, N-1$, as follows:

$$\begin{aligned} H_k^o =& \min_{\tilde{\mathbf{u}}(k)}\big\{\boldsymbol{\lambda}^T(k)\big[\mathbf{A}_\lambda(k)\mathbf{z}(k) + \mathbf{B}_\lambda(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_\lambda(k)\big]\big\} \\ & \text{subject to:} \qquad \mathbf{g}(\tilde{\mathbf{u}}(k)) \leq \mathbf{0} \end{aligned} \tag{17}$$

We observe that, since the optimal trajectories are assumed to be known, matrices $\mathbf{A}_\lambda(k)$ are also known. This fact allows us to compute $\boldsymbol{\lambda}(k)$ and thus solve problem (17) iteratively. Finally, we note that the Hamiltonian sequence is linear in $\tilde{\mathbf{u}}(k)$, thus problem (13) reduces to the solution of a sequence of linear programming problems of the following form:

$$\begin{aligned} & \min_{\tilde{\mathbf{u}}(k)} \big\{\mathbf{a}^T(k)\,\tilde{\mathbf{u}}(k)\big\} \qquad \text{subject to:} \\ & \mathbf{g}(\tilde{\mathbf{u}}(k)) \leq \mathbf{0} \end{aligned} \tag{18}$$

for $k = 0, \ldots, N-1$, where the cost coefficient vectors are defined as $\mathbf{a}^T(k) = \boldsymbol{\lambda}^T(k)\mathbf{B}_\lambda(k)$.

With the following theorem we demonstrate that the optimum solution of the linear problem (13) is also a solution for the non–linear problem (12).

Theorem 4.1. *Consider the following constrained non–linear programming problem*

$$\begin{aligned} & \min_{\mathbf{u}}\ J(\mathbf{u}) \quad \textit{subject to:} \\ & \mathbf{g}(\mathbf{u}) \leq \mathbf{0} \end{aligned} \tag{19}$$

and the following constrained linear programming problem

$$\begin{aligned} &\min_{\mathbf{u}} \ \mathbf{c}^T\mathbf{u} \quad \textit{subject to:} \\ &\mathbf{g}(\mathbf{u}) \leq \mathbf{0} \end{aligned} \tag{20}$$

Let $\mathcal{U}$ be the feasible region for both problems, $\mathbf{u}^o$ an optimal solution for the problem (20) and suppose $\mathbf{u}^o$ is a regular point of the constraints $\mathbf{g}(\mathbf{u}) \leq \mathbf{0}$. If

$$\nabla_{\mathbf{u}} J(\mathbf{u}^o) \neq \mathbf{0}, \qquad \forall\, \mathbf{u} \in \mathcal{U} \tag{21}$$

$$\nabla_{\mathbf{u}} J(\mathbf{u}^o) = \mathbf{c}^T \tag{22}$$

then $\mathbf{u}^o$ satisfies the necessary conditions for the solution of the problem (19).

Proof. If $\mathbf{u}^o$ satisfies the necessary condition for the solution of problem (19), then there exists a vector $\tilde{\boldsymbol{\mu}} \geq \mathbf{0}$ such that (Kuhn-Tucker conditions)

$$\begin{aligned} &\nabla_{\mathbf{u}} J + \tilde{\boldsymbol{\mu}}^T \nabla_{\mathbf{u}}\, \mathbf{g}(\mathbf{u}^o) = \mathbf{0} \\ &\tilde{\boldsymbol{\mu}}^T \mathbf{g}(\mathbf{u}^o) = \mathbf{0}. \end{aligned} \tag{23}$$

However by hypothesis $\mathbf{u}^o$ is a solution for the problem (20), then it must satisfy for a certain $\boldsymbol{\mu} \geq \mathbf{0}$

$$\begin{aligned} &\mathbf{c}^T + \boldsymbol{\mu}^T \nabla_{\mathbf{u}}\, \mathbf{g}(\mathbf{u}^o) = \mathbf{0} \\ &\boldsymbol{\mu}^T \mathbf{g}(\mathbf{u}^o) = \mathbf{0}. \end{aligned} \tag{24}$$

From condition (22) it is immediate to verify that (23) is equal to (24) with $\tilde{\boldsymbol{\mu}} = \boldsymbol{\mu}$.
This completes the proof. ■

The previous theorem provides a necessary condition for the optimum solution of problem (12), i.e., if $\mathbf{u}^o$ is an optimum solution of problem (13) then it is also a solution of problem (12). In fact if the linear problem admits a unique optimum solution $\mathbf{u}^o$, and condition (21) of the previous theorem allows us to avoid degenerate solutions, then $\mathbf{u}^o$ must be a stationary point for the problem (12).

Finally, to obtain the optimum solution for the problem (12) we may choose any starting values for the multiplier sequence $\boldsymbol{\nu}^T$ and solve the sequence of linear programming problems. Then we can tune $\boldsymbol{\nu}^T$ such as to satisfy the terminal conditions and the optimum solution obtained by solving problem (13), if it is unique, for Theorem 4.1 is also the optimum solution of problem (12).

5 Summary and Conclusions

In this paper we have approached the control problem of determining optimal machines production rates so as to maximize the total production while guaranteeing a given production mix over a finite planning horizon. We have shown that this problem has a necessary condition which lends itself to the evaluation of the optimum machine production rates for each sample period through a sequence of linear programming problems. A discrete time optimum control problem has been formalized and more general constraints, such as release times and due dates, can be easily accounted for. Our main future goal will be to extend the results obtained for the control of deterministic production system to the stochastic case.

Appendix A

Here we briefly develop the discrete–time, time–varying state variable model which describes the evolution in time of the hybrid system introduced in Section 2.2 and that can be expressed in matrix notation as:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{D}(k)\big[\mathbf{x}(k) + \mathbf{R}\,\mathbf{u}(k)\Delta\big(\mathbf{u}(k),\mathbf{x}(k)\big)\big] \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases} \tag{25}$$

where:

- $\Delta\big(\mathbf{u}(k),\mathbf{x}(k)\big) = (t_{k+1} - t_k)$ is the length of the k–th macro–period,
- $\mathbf{x}(k) = [\ldots, x_i^r(t_k), \chi_i(t_k), s_i(t_k), \ldots]^T$ is the state vector, as given in (6),
- $\mathbf{u}(k)$ is the average machine production rates vector,
- $\mathbf{D}(k)$ is a diagonal matrix with entries 0 and 1,
- $\mathbf{R}$ is the weight matrix for the machine outflow rates with entries 0, 1, -1.

With this model any macro–event happens when an appropriate component of the state vector reaches a specified value given by the sequence $d(k)$. Precisely when a machine fails or gets repaired then it must result $\chi_i(t_{k+1}) = w_{i,\alpha}$ or $s_i(t_{k+1}) = f_{i,\beta}$. When a buffer gets full or empty for parts of class r then the condition $\sum_r x_i^r(t_{k+1}) = C_i$ or $x_i^r(t_{k+1}) = 0$ will be satisfied.
Let us define $\mathbf{h}(k) = \mathbf{e}_j^T(k)\mathbf{D}(k)$, $q(k) = \mathbf{e}_j^T(k)\mathbf{D}(k)\mathbf{R}\,\mathbf{u}(k)$ and $K(k) = \frac{1}{q(k)}$. Then we have:

$$\begin{aligned} d(k+1) &= \mathbf{e}_j^T(k)\mathbf{x}(k+1) \\ &= \mathbf{h}(k)\mathbf{x}(k) + q(k)\Delta\big(\mathbf{u}(k),\mathbf{x}(k)\big) \end{aligned} \tag{26}$$

and the length of the k–th macro–period is given by

$$\Delta\big(\mathbf{u}(k),\mathbf{x}(k)\big) = K(k)\big[d(k+1) - \mathbf{h}(k)\mathbf{x}(k)\big] \tag{27}$$

where $\mathbf{e}_j^T(k)$ is a unit vector which selects the state variable within $\mathbf{x}(k)$ that has generated the macro–state transition. Equation (25) can be rewritten as:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\big(k,\mathbf{u}(k)\big)\mathbf{x}(k) + \mathbf{b}\big(k,\mathbf{u}(k)\big)d(k+1) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases}$$

where $\mathbf{A}\big(k,\mathbf{u}(k)\big) = \mathbf{D}(k)\big[\mathbf{I} - K\big(k,\mathbf{u}(k)\big)\mathbf{R}\,\mathbf{u}(k)\mathbf{h}(k)\big]$ and $\mathbf{b}\big(k,\mathbf{u}(k)\big) = \mathbf{D}(k)K\big(k,\mathbf{u}(k)\big)\mathbf{R}\,\mathbf{u}(k)$.

Appendix B

We consider perturbations from the optimal trajectories p^o and x^o produced by admissible infinitesimal perturbations $\delta\mathbf{u}(k)$ in the optimal control sequence u^o.
By linearizing (8) around p^o we have:

$$\begin{cases} \delta\mathbf{p}(k+1) = \delta\mathbf{p}(k) + \mathbf{B}\,\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\,\delta\mathbf{u}(k) \\ \qquad\qquad + \mathbf{B}\,\mathbf{u}^o(k)\big[\nabla_{\mathbf{u}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\,\delta\mathbf{u}(k) \\ \qquad\qquad + \nabla_{\mathbf{x}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\,\delta\mathbf{x}(k)\big] \end{cases}$$

where the gradients in the last expression can be easily obtained from the results presented in Appendix A. If we let

$$\begin{aligned}\mathbf{A}_{px}(k) &= \mathbf{B}\,\mathbf{u}^o(k)\,\nabla_{\mathbf{x}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\\ \mathbf{B}_p(k) &= \mathbf{B}\big[\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\,\mathbf{I}\\ &\qquad + \mathbf{u}^o(k)\,\nabla_{\mathbf{u}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\big]\end{aligned}$$

then we simply obtain

$$\delta\mathbf{p}(k+1) = \delta\mathbf{p}(k) + \mathbf{A}_{px}(k)\,\delta\mathbf{x}(k) + \mathbf{B}_p(k)\,\delta\mathbf{u}(k).$$

Let us define $\tilde{\mathbf{p}}(k) = \mathbf{p}^o(k) + \delta\mathbf{p}(k)$, $\tilde{\mathbf{x}}(k) = \mathbf{x}^o(k) + \delta\mathbf{x}(k)$ and $\tilde{\mathbf{u}}(k) = \mathbf{u}^o(k) + \delta\mathbf{u}(k)$. The linear perturbed system dynamics is given by

$$\begin{cases}\tilde{\mathbf{p}}(k+1) = \tilde{\mathbf{p}}(k) + \mathbf{A}_{px}(k)\tilde{\mathbf{p}}(k)\\ \qquad\qquad + \mathbf{B}_p(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_p(k)\\ \tilde{\mathbf{p}}(0) = \mathbf{p}_0\\ \tilde{\mathbf{y}}(k) = \mathbf{C}\tilde{\mathbf{p}}(k)\end{cases}\tag{28}$$

where $\mathbf{G}_p(k) = \big[\mathbf{B}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big) - \mathbf{B}_p(k)\big]\mathbf{u}^o(k) - \mathbf{A}_{px}(k)\mathbf{x}^o$.

By linearizing (25) or (5) around the optimal trajectory x^o we have:

$$\begin{cases}\delta\mathbf{p}(k+1) = \mathbf{D}(k)\delta\mathbf{x}(k) + \mathbf{D}(k)\mathbf{R}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\\ \quad\cdot\,\delta\mathbf{u}(k) + \mathbf{D}(k)\mathbf{R}\mathbf{u}^o(k)\big[\nabla_{\mathbf{u}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\\ \quad\cdot\,\delta\mathbf{u}(k) + \nabla_{\mathbf{x}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\,\delta\mathbf{x}(k)\big]\end{cases}$$

and if we let

$$\begin{aligned}\mathbf{A}_{xx}(k) &= \mathbf{D}(k)\big[\mathbf{I} + \mathbf{R}\mathbf{u}^o(k)\nabla_{\mathbf{x}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\big]\\ \mathbf{B}_x(k) &= \mathbf{D}(k)\mathbf{R}\big[\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\\ &\qquad + \mathbf{u}^o(k)\nabla_{\mathbf{u}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\big]\end{aligned}$$

then we simply obtain

$$\delta\mathbf{x}(k+1) = \mathbf{A}_{xx}(k)\delta\mathbf{x}(k) + \mathbf{B}_x(k)\,\delta\mathbf{u}(k).$$

The linear perturbed system dynamics is given by

$$\begin{cases}\tilde{\mathbf{p}}(k+1) = \mathbf{A}_{xx}(k)\tilde{\mathbf{p}}(k)\\ \qquad\qquad + \mathbf{B}_x(k)\tilde{\mathbf{u}}(k) + \mathbf{G}_x(k)\\ \tilde{\mathbf{p}}(0) = \mathbf{x}_0\end{cases}\tag{29}$$

where

$$\begin{aligned}\mathbf{G}_x(k) = &\big[\mathbf{D}(k)\mathbf{R}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big) - \mathbf{B}_x(k)\big]\mathbf{u}^o(k)\\ &- \mathbf{D}(k)\mathbf{R}\mathbf{u}^o\nabla_{\mathbf{x}^o(k)}\Delta\big(\mathbf{u}^o(k),\mathbf{x}^o(k)\big)\mathbf{x}^o.\end{aligned}$$

References

[1] R. Akella, P.R. Kumar, "Optimal Control of Prduction Rate in a Failure Prone Manufacturing System," *IEEE Trans. Automatic Control*, Vol. 31, No. 2, pp. 116–126, 1986.

[2] F. Balduzzi, G. Menga, “A State Variable Model for the Fluid Approximation of Flexible Manufacturing Systems,” *IEEE International Conference on Robotics and Automation* (Leuven, Belgium), May 1998, pp. 1172–1178.

[3] F. Balduzzi, A. Giua, G. Menga, “Hybrid Analysis of Automated Manufacturing Systems Using Discrete Linear Inclusions,” *IEEE Conference on Decision and Control* (Tampa, USA), December 1998, pp. 1710–1715.

[4] A.E. Bryson, Y.C. Ho, *Applied Optimal Control*, Ginn and Company, 1969.

[5] D. Connors, G. Feigin, D. Yao, “Scheduling Semiconductor Lines Using a Fluid Network Model,” *IEEE Trans. Robotics and Automation*, Vol. 10, No. 2, pp. 88–98, 1994.

[6] S.B. Gershwin, *Manufacturing Systems Engineering*, Prentice Hall, Inc., Englewood Cliffs, NJ, 1994.

[7] A. Mandelbaum, H. Chen, “Discrete Flow Networks: Bottleneck Analysis and Fluid Approximations,” *Math. Opns. Res.*, Vol. 16, pp. 408–446, 1991.

[8] E. Presman, S.P. Sethi, Q. Zhang, “Optimal Feedback Production Planning in a Stochastic N–machine Flowshop,” *Automatica*, Vol. 31, No. 9, pp. 1325–1332, 1995.

[9] S.P. Sethi, X.Y. Zhou, “Stochastic Dynamic Jobshops and Hierarchical Production Planning,” *IEEE Trans. Automatic Control*, Vol. 39, No. 10, pp. 2061–2076, 1994.