A Generalized Stochastic Petri net Model for Performance Analysis and Control of Capacitated Re-entrant Lines

Jin Young Choi and Spyros A. Reveliotis*

Abstract— **The basic definition of the re-entrant line, which constitutes the typical abstraction for the formal modelling and analysis of the fab scheduling problem, considers only the job contest for the finite processing capacity of the system workstations, ignoring completely the effects and complications arising from additional operational issues like the finite buffering capacity of the system workstations / production units. Yet, as the semiconductor industry moves to more extensively automated operational modes, the explicit characterization and control of these additional operational features is of paramount importance for the robust and stable operation of the entire system. Moreover, the operational policies developed to control these logical aspects of the system behavior introduce additional constraints to the fab scheduling problem, that complicate it even further and, more importantly, invalidate prior characterizations of its optimal solutions. Motivated by these remarks, the work presented in this paper develops a novel analytical framework for the modelling, analysis and control of capacitated, flexibly automated re-entrant lines, based on the class of Generalized Stochastic Petri nets (GSPN's). The proposed framework (i) allows the seamless integration of the logical/structural and the timed-based aspects of the system behavior, (ii) provides an analytical formulation for the underlying scheduling problem, and (iii) leads to an interesting qualitative characterization of the structure of the optimal scheduling policy. Hence, it provides the analytical basis for addressing the re-entrant line scheduling problem in its contemporary, more complex operational context, and it constitutes the starting point for the development of new scheduling tools and policies for it.**

Keywords— **Capacitated Re-entrant Lines, Performance Modelling and Control, Scheduling, Timed Petri nets**

I. Introduction

Currently, the *re-entrant (production) line* is the most typical abstraction for the formal modelling and analysis of the fab scheduling problem. In its basic characterization [1], such a line supports the production of a single item through m workstations, W_1, $W_2, \ldots, W_m$. Each workstation W_i, $i = 1, \ldots, m$, possesses S_i identical servers, and the production of each unit occurs in n stages, J_1, $J_2, \ldots, J_n$, with stage J_j, $j = 1 \ldots, n$, being supported by one of the system workstations, to be denoted by $W(J_j)$. The re-entrant nature of the line is expressed by the fact that there exists at least one workstation W_k such that $|\{j : W(J_j) = W_k\}| \geq 2$, and raises the problem of determining how to allocate the workstation processing capacity to the job stages competing for it, in order to optimize some pre-specified performance objective(s).[1] The resulting scheduling problem has been investigated extensively in the last decade, and many of the developed results are analytically strong and of high mathematical sophistication. A representative and insightful exposition of these results is provided in the recent survey paper of [2].

Yet, as it is evident from the above description, the basic re-entrant line model considers that each workstation possesses infinite buffering capacity, a feature that in the past has been justified by the presence of the human operator in the fab shop-floor, that handily addressed any potential overflow problems. Currently, the migration of modern fabs to highly automated modes of operation, through the advent of 300mm production technology, necessitates the development of explicit real-time control logic that will establish the logically correct and consistent operation of the fab shop-floor, including the orderly allocation of limited resources like the buffering capacity of the system workstations and the interconnecting material handling equipment. The corresponding set of real-time control problems is collectively known as the fab logical or structural control problem, and it is treated in [3]. As it is argued in [3], the explicit modelling of these additional operational aspects and the control policies developed to address the fab logical control problem, introduce additional constraints to the complementary performance control problem, which, therefore, must be re-investigated in this new operational context. Indeed, a preliminary study on the problem of scheduling structurally controlled re-entrant lines has indicated that the introduction of the finite buffering capacity and the corresponding structural control logic into the fab operational model, leads to additional material flow dynamics, that negate in a strong qualitative sense prior analytical results, obtained through the study of the basic re-entrant line model outlined above [4].

Motivated from the above remarks, the work presented in this paper proposes a novel formal framework for analysis and control of the re-entrant line modelling the emerging flexibly automated fab, based on the broader class of *Generalized Stochastic Petri nets (GSPN)* [5], [6]. More specifically, first it is shown that the GSPN modelling framework provides a systematic integrated representation of the timed and the logical/behavioral dynamics of the structurally controlled, capacitated re-entrant line, which when analyzed through standard GSPN performance evaluation techniques, leads to an analytical characterization of the underlying scheduling problem, in the form of a Mathematical Programming (MP) formulation. This formulation is subsequently shown to be *exactly* solvable through enumerative techniques for a variety of (steady-state) performance objectives, thanks to some important properties of the structure of the optimal scheduling policy. Finally, the application of the developed results to the detailed analysis of a small capacitated re-entrant line exemplifies the presented theory, but more importantly, it reveals the fundamental structural difference between the optimal scheduling policies for capacitated and uncapacitated re-entrant lines, even in the more stochastic operational context presumed by the GSPN modelling framework, and concurs the results presented in [4], which were developed under a more deterministic set of assumptions regarding the timing of the system operations.

Due to space limitations, the subsequent development assumes that the reader is familiar with the GSPN modelling framework and the relevant theory. An excellent introduction to it can be found in [5], [6]. Also, it is noted, for completeness, that an extensive coverage of the use of the broader class of timed PN models for manufacturing system modelling and analysis until the middle 1990's, can be found in [7], [8], while some more recent applications of the timed PN theory to the modeling and performance evaluation of manufacturing – including semiconductor – systems are the works presented in [9], [10] and [11].

Finally, before proceeding with the detailed presentation of the paper results, we want to emphasize that the nature of the intended contribution is rather qualitative, i.e., providing detailed analytical characterizations of the capacitated re-entrant line scheduling problem, the structure of the optimal solution, and its differentiation from past results on uncapacitated re-entrant lines. As it is demonstrated by the presented example, the implementation of the proposed methodology to actual fab environments will be severely limited by the very high (super-polynomial) complexity of the approach. Yet, the presented

* corresponding author, School of Industrial & Systems Engineering, Georgia Institute of Technology, 765 Ferst Drive, Atlanta, GA 30332, phone: (404) 894-6608, fax: (404) 894-2301, e-mail: *spyros@isye.gatech.edu*

[1] Due to the very high capital cost of modern fabs, and the market forces driving the fab economics, the major performance objective addressed by the re-entrant line scheduling problem is the maximization of the system throughput.

characterizations are intended to provide the analytical insights and benchmarking cases[2] for any effort towards the eventual development of pertinent approximations to the underlying optimization problem. In fact, the development of such pertinent approximations to the scheduling problem of the structurally controlled fab, is part of our current investigations.

II. The capacitated re-entrant line and its GSPN model

The capacitated re-entrant line considered in this work refines the basic re-entrant line model, presented in the introductory section, through the explicit modeling of (i) the workstation buffering capacity and its internal material flow, and (ii) the interconnecting material handling system. More specifically, it is assumed that each workstation W_i, $i = 1, \ldots, m$, consists of C_i buffer slots and S_i identical servers. Each part visiting the workstation for the execution of some processing stage is allocated one unit of buffering capacity, which it holds exclusively during its entire sojourn in the station. Once in the station local buffer, the part competes for one of the station servers for the execution of the requested stage. Under the current model definition, it can be assumed either that the part is mounted into the server for its processing and then it is returned to its designated slot, or that the server processes the part by visiting the corresponding buffer. A part having finished the processing of its current stage at a certain station, waits in its allocated buffer for transfer to the next requested station. This transfer is facilitated by the central (automated) material handling system, and it is authorized by a supervisory control policy ensuring that (i) the destination workstation has available buffering capacity, and (ii) the transfer is *safe*, i.e., it is still physically possible from the resulting state to process all running jobs to completion. In the subsequent analysis, the central material handling system can be considered to be either a centrally located robotic manipulator, or a single-loop AGV system; in the former case, the re-entrant line is the modeling abstraction for what is known as a cluster tool, while in the latter case, the resulting model represents the dynamics of a modern fab bay, where the various process tools possess a local stocker of limited buffering capacity.

Following the typical practice, the main scheduling objective considered in the undertaken analysis is the maximization of the long-run system throughput, and therefore, it is assumed that there exists an infinite amount of raw material waiting for processing at the line's Input/Output (I/O) station. Furthermore, in order to facilitate the GSPN-based modeling and analysis, it is also assumed that all stage processing and transfer times are exponentially distributed. In particular, the processing time of stage J_j, $j = 1 \ldots, n$, is assumed to follow an exponential distribution with finite non-zero mean $m_j = 1/\mu_j$, while job transfer times are assumed to be exponentially distributed with a mean $d = 1/\lambda$, that applies uniformly across all the transferring operations. This presumed uniformity of the mean transfer times is introduced in order to simplify the computations involved in the presented example, and it also allows the analytical investigation of the limiting case where the transfer times are negligible with respect to the processing times involved, by taking $\lambda \to \infty$ in the derived expressions. Finally, we notice that the rather unrealistic assumption of exponentially distributed processing and transfer times can be eventually relaxed in the resulting GSPN model, by substituting each timed transition in that net with a GSPN subnet, modeling a phase-type distribution that approximates, to any desired degree of accuracy, the original/empirical distribution of the corresponding event timing. We refer the reader to [12] for a detailed treatment of phase-type distributions and the relevant approximation theory.

[2]thanks to the exact solvability of the presented formulation for small system configurations

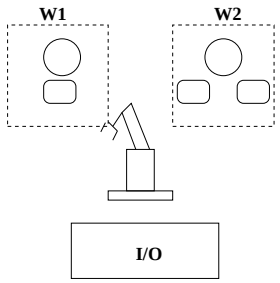

Fig. 1. Example: The capacitated re-entrant line

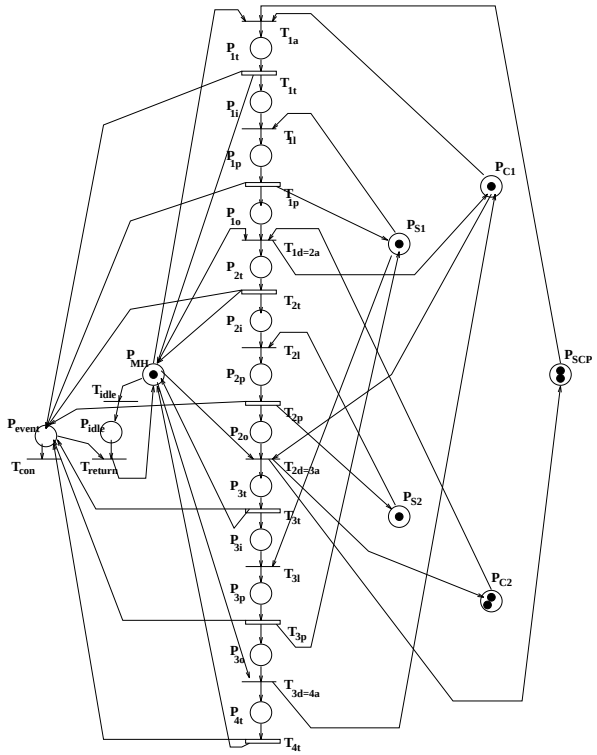

Fig. 2. Example: The GSPN model

Example: The above general description of the capacitated re-entrant line is exemplified by the small system presented in Figure 1. The depicted configuration possesses two stations, W_1 and W_2, with $S_1 = S_2 = 1$ and $C_1 = 1$; $C_2 = 2$. Furthermore, the supported production sequence is $J =< J_1, J_2, J_3 >$, with $W(J_1) = W(J_3) = W_1$ and $W(J_2) = W_2$. Finally, stage processing times are exponentially distributed with means $m_j = 1/\mu_j > 0$, $j = 1, 2, 3$, and so are the involved transfer times, with a uniform mean $d = 1/\lambda$. For this small configuration, it is easy to see that, under the operational assumptions outlined above, the system material flow will remain deadlock-free, as long as

$$|J_1| + |J_2| \leq C_1 + C_2 - 1 = 2 \tag{1}$$

where $|J_j|$, $j = 1, 2, 3$ denotes the number of job instances in $W(J_j)$ executing stage J_j. □

The GSPN modeling the behavior of the capacitated re-entrant line of Figure 1, under the control of the maximally permissive structural control policy (SCP) of Equation 1, is

depicted in Figure 2. Specifically, in the GSPN of Figure 2, the part flow dynamics associated with each processing stage J_j, $j = 1, 2, 3$, are modeled by the corresponding net path $< T_{ja}, P_{jt}, T_{jt}, P_{ji}, T_{jl}, P_{jp}, T_{jp}, P_{jo}, T_{jd} >$, while it also holds $T_{jd} \equiv T_{j+1,a}$, with $j = 4$ denoting the last unloading step. A token in place P_{jt} represents a part in transit to the buffer of workstation $W(J_j)$; a token in place P_{ji} represents a part in the buffer of $W(J_j)$ waiting the allocation of one of the buffer servers; a token in place P_{jp} represents a part in processing of stage J_j; finally, a token in place P_{jo} represents a part having finished processing of stage J_j, and waiting for transfer to the next requested workstation or, in case that J_j is the last processing stage, to the I/O station. On the other hand, places P_{MH}, P_{S_i}, P_{C_i}, $i = 1, 2$, and P_{SCP} model respectively the availability of the system transporter, workstation servers and buffers, and the logic of the applied SCP, according to the standard, by now, modeling practice of resource-process nets [13]. It is important to notice that transitions T_{ja}, T_{jl} and T_{jd}, that are associated with the various decisions regarding the allocation of the system buffering, processing and/or transport capacity, are untimed / immediate transitions, while the delays experienced from the processing and/or transfer times involved with the execution of these decisions, are modeled by the timed transitions T_{jt} and T_{jp}. As mentioned above, this separation of the net components modeling the timings of the various system events from the net structure modeling the underlying resource allocation and the associated decision making, enables the modeling of timing distributions other than exponential through the (local) substitution of the corresponding timed transitions by GSPN subnets modeling the approximating phase-type distributions. It also allows, as it is shown below, the modeling of the required scheduling logic through a set of *dynamic random switches*, that resolve the conflicts among the immediate transitions that are simultaneously enabled at the net reachable vanishing markings. Finally, some explanation is necessary about the role of places P_{idle}, P_{event} and their associated transitions T_{idle}, T_{return} and T_{con}. This subnet essentially establishes a GSPN-compatible mechanism for representing some deliberate idleness in the underlying scheduling logic, since, in the considered operational context, the optimal scheduling policy is not necessarily non-idling. Hence, the triggering of transition T_{idle} consumes the transporter-modeling token, which remains in place P_{idle}, until the immediate transition T_{return} is enabled through the presence of a token in place P_{event}. P_{event} is marked every time that one of the system timed transitions fires, signaling the completion of some event. Notice that T_{return} will always be in conflict with transition T_{con}, but it is assumed to have priority over the latter, which is technically imposed by setting the corresponding (static) random switch to $\{\xi_{T_{return}} = 1,\ \xi_{T_{con}} = 0\}$. Finally, T_{con} is a sink transition that "consumes" event completion signaling tokens, in case that the transporter is not (deliberately) idling.

III. GSPN-based performance evaluation and the capacitated re-entrant line scheduling problem

According to the general GSPN theory [6], the marking process of a GSPN net, $\mathcal{N}$, is a semi-Markov process with a discrete state space, $\mathcal{S}$, given by the net reachability space $R(N, M_0)$. $\mathcal{S}$ is partitioned to *vanishing* states / markings, $\mathcal{V}$, which enable at least one immediate transition of $\mathcal{N}$, and therefore, they have zero sojourn time, and *tangible* markings, $\mathcal{T}$, which enable only timed transitions, and therefore, they present positive sojourn times. Furthermore, the untimed system dynamics, defined by its transitional patterns among the various states of its reachable state space, are characterized by the, so called, *Embedded Markov Chain (EMC)*, whose branching probabilities, $Q = [q_{kl}]$ are determined by the specified *(dynamic) random switches*, in case of vanishing markings, and the enabled event *exponential race*, in case of tangible markings. If this EMC is finite-state, homogeneous, and irreducible, it possesses a steady-state distribution $\mathbf{y} = [y_k]$, determined through the following system of equations:

$$\mathbf{y} = \mathbf{y}Q \ ; \ \sum_{s_k \in \mathcal{S}} y_k = 1 \tag{2}$$

Furthermore, the steady-state probabilities, $\pi = [\pi_k]$, for the underlying continuous-time stochastic process, are obtained through the following formula:

$$\pi_k = \begin{cases} 0, & s_k \in \mathcal{V} \\ y_k E[s_k] / \sum_{s_l \in \mathcal{T}} y_l E[s_l] & s_k \in \mathcal{T} \end{cases} \tag{3}$$

In Equation 3, $E[s_k]$ denotes the expected sojourn time for tangible marking $s_k \in \mathcal{T}$, and it is computed by:

$$E[s_k] = 1 \ / \sum_{T_j \text{ enabled in } s_k} r_j \tag{4}$$

where r_j denotes the (firing) rate of (timed) transition T_j. Once the steady-state probability vector π has been obtained, various performance measures of interest can be defined as appropriate functions of π and the other system parameters.

In the case of GSPN's modelling the behavior of capacitated re-entrant lines, the underlying EMC is finite-state and homogeneous, but it might contain absorbing states due to the presence of transition T_{idle}. Specifically, if T_{idle} fires while no other event is in process, the token representing the system transporter will be permanently stuck in place P_{idle}. This problem can be addressed by disabling these problematic firings of T_{idle} through appropriate setting of the corresponding dynamic random switches. The resulting modified EMC has the property that from every pair of states s_i and s_j in it, there exists a deterministic scheduling policy that renders s_j accessible from s_i.[3] This property subsequently guarantees the existence of an optimal pricing of the random switching probabilities, ξ_l, appearing in the modified EMC, that leads to a controlled system behavior that is modelled by a *unichain* Markov chain, i.e., a Markov chain consisting of a single communicating class and possibly a set of transient states (c.f. [14], Section 8.3); in the following, the scheduling policies resulting from such pricings will be referred to as *unichain* policies, and their set will be denoted by UP. Constraining the search for an optimal scheduling policy in set UP, and letting $\overline{Q}(\xi)$ denote the transition probability matrix (TPM) of the aforementioned modified EMC, resulting from the removal of all the absorbing states, we obtain the following MP formulation for the problem of throughput maximization for a capacitated re-entrant line:

$$\max_{\xi \in \mathbf{UP}} TH(\xi) \equiv \sum_{(k,j)} \pi_k r_j I \ \begin{matrix} \{\text{Transition } T_j \text{ enabled in } s_k \\ \wedge \text{ cor. to an unloading event}\} \end{matrix} \tag{5}$$

s.t.

$$\forall l, \ \xi_l \geq 0.0 \tag{6}$$

$$\forall \text{ random switch } \Xi_u, \ \sum_{l:\xi_l \in \Xi_u} \xi_l = 1.0 \tag{7}$$

and

Equations 2 and 3

applied over the modified EMC.

[3] This is the effect - in fact, the objective - of the applied structural control policy.

TABLE I
EXAMPLE:THE EMC MARKINGS

s_k	$P_{1t}P_{1i}P_{1p}P_{1o}$	$P_{2t}P_{2i}P_{2p}P_{2o}$	$P_{3t}P_{3i}P_{3p}P_{3o}P_{4t}$	$P_{MH}P_{idle}P_{event}$	$P_{S_1}P_{S_2}$	$P_{C_1}P_{C_2}P_{SCP}$
0	0 0 0 0	0 0 0 0	0 0 0 0 0	1 0 0	1 1	1 2 2
1	1 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0	1 1	0 2 1
2	0 0 0 0	0 0 0 0	0 0 0 0 0	0 1 0	1 1	1 2 2
3	0 1 0 0	0 0 0 0	0 0 0 0 0	1 0 1	1 1	0 2 1
4	0 0 1 0	0 0 0 0	0 0 0 0 0	0 1 0	0 1	0 2 1
5	0 0 0 1	0 0 0 0	0 0 0 0 0	0 1 1	1 1	0 2 1
6	0 0 0 1	0 0 0 0	0 0 0 0 0	1 0 0	1 1	0 2 1
7	0 0 0 1	0 0 0 0	0 0 0 0 0	0 1 0	1 1	0 2 1
8	0 0 0 0	1 0 0 0	0 0 0 0 0	0 0 0	1 1	1 1 1
9	0 0 0 0	0 1 0 0	0 0 0 0 0	1 0 1	1 1	1 1 1
10	0 0 0 0	0 0 1 0	0 0 0 0 0	1 0 0	1 0	1 1 1
11	0 0 0 0	0 0 1 0	0 0 0 0 0	0 1 0	1 0	1 1 1
12	1 0 0 0	0 0 1 0	0 0 0 0 0	0 0 0	1 0	0 1 0
13	0 0 0 0	0 0 0 1	0 0 0 0 0	0 1 1	1 1	1 1 1
14	0 0 0 0	0 0 0 1	0 0 0 0 0	1 0 0	1 1	1 1 1
15	0 0 0 0	0 0 0 1	0 0 0 0 0	0 1 0	1 1	1 1 1
16	0 0 0 0	0 0 0 0	1 0 0 0 0	0 0 0	1 1	0 2 2
17	1 0 0 0	0 0 0 1	0 0 0 0 0	0 0 0	1 1	0 1 0
18	0 0 0 0	0 0 0 0	0 1 0 0 0	1 0 1	1 1	0 2 2
19	0 0 0 0	0 0 0 0	0 0 1 0 0	0 1 0	0 1	0 2 2
20	0 0 0 0	0 0 0 0	0 0 0 1 0	0 1 1	1 1	0 2 2
21	0 0 0 0	0 0 0 0	0 0 0 1 0	1 0 0	1 1	0 2 2
22	0 0 0 0	0 0 0 0	0 0 0 1 0	0 1 0	1 1	0 2 2
23	0 0 0 0	0 0 0 0	0 0 0 0 1	0 0 0	1 1	1 2 2
24	0 0 0 0	0 0 0 0	0 0 0 0 0	1 0 1	1 1	1 2 2
25	0 1 0 0	0 0 1 0	0 0 0 0 0	1 0 1	1 0	0 1 0
26	1 0 0 0	0 0 0 1	0 0 0 0 0	0 0 1	1 1	0 1 0
27	0 0 1 0	0 0 1 0	0 0 0 0 0	0 1 0	0 0	0 1 0
28	0 0 0 1	0 0 1 0	0 0 0 0 0	0 1 1	1 0	0 1 0
29	0 0 1 0	0 0 0 1	0 0 0 0 0	0 1 1	0 1	0 1 0
30	0 0 0 1	0 0 1 0	0 0 0 0 0	1 0 0	1 0	0 1 0
31	0 0 0 1	0 0 1 0	0 0 0 0 0	0 1 0	1 0	0 1 0
32	0 0 0 0	1 0 1 0	0 0 0 0 0	0 0 0	1 0	1 0 0
33	0 0 0 1	0 0 0 1	0 0 0 0 0	0 1 1	1 1	0 1 0
34	0 0 0 1	0 0 0 1	0 0 0 0 0	1 0 0	1 1	0 1 0
35	0 0 0 0	1 0 0 1	0 0 0 0 0	0 0 0	1 1	1 0 0
36	0 0 0 1	0 0 0 1	0 0 0 0 0	0 1 0	1 1	0 1 0
37	0 0 0 0	0 1 0 1	0 0 0 0 0	1 0 1	1 1	1 0 0
38	0 0 0 0	0 0 1 1	0 0 0 0 0	0 1 0	1 0	1 0 0
39	0 0 0 0	0 0 0 2	0 0 0 0 0	0 1 1	1 1	1 0 0
40	0 0 0 0	0 0 0 2	0 0 0 0 0	1 0 0	1 1	1 0 0
41	0 0 0 0	0 0 0 2	0 0 0 0 0	0 1 0	1 1	1 0 0
42	0 0 0 0	0 0 0 1	1 0 0 0 0	1 0 0	1 1	0 1 1
43	0 0 0 0	0 0 0 1	0 1 0 0 0	1 0 1	1 1	0 1 1
44	0 0 0 0	0 0 0 1	0 0 1 0 0	0 1 0	0 1	0 1 1
45	0 0 0 0	0 0 0 1	0 0 0 1 0	0 1 1	1 1	0 1 1
46	0 0 0 0	0 0 0 1	0 0 0 1 0	1 0 0	1 1	0 1 1
47	0 0 0 0	0 0 0 1	0 0 0 1 0	0 1 0	1 1	0 1 1
48	0 0 0 0	0 0 0 1	0 0 0 0 1	0 0 0	1 1	1 1 1
49	0 0 0 0	0 0 0 1	0 0 0 0 0	1 0 1	1 1	1 1 1
50	0 1 0 0	0 0 0 1	0 0 0 0 0	1 0 1	1 1	0 1 0
51	0 0 1 0	0 0 0 1	0 0 0 0 0	0 1 0	0 1	0 1 0
52	0 0 1 0	0 0 0 1	0 0 0 0 0	1 0 0	0 1	0 1 0
53	0 0 0 0	0 1 1 0	0 0 0 0 0	1 0 1	1 0	1 0 0
54	0 0 0 0	1 0 0 1	0 0 0 0 0	0 0 1	1 1	1 0 0
55	0 0 0 0	0 1 1 0	0 0 0 0 0	0 1 0	1 0	1 0 0
56	0 0 0 0	0 1 0 1	0 0 0 0 0	0 1 1	1 1	1 0 0
57	0 0 0 0	0 0 1 1	0 0 0 0 0	1 0 0	1 0	1 0 0
58	0 0 0 0	0 0 1 0	1 0 0 0 0	0 0 0	1 0	0 1 1
59	0 0 0 0	0 0 0 1	1 0 0 0 0	0 0 1	1 1	0 1 1
60	0 0 0 0	0 0 1 0	0 1 0 0 0	1 0 1	1 0	0 1 1
61	0 0 0 0	0 0 1 0	0 0 1 0 0	0 1 0	0 0	0 1 1
62	0 0 0 0	0 0 0 1	0 0 1 0 0	0 1 1	0 1	0 1 1
63	0 0 0 0	0 0 1 0	0 0 0 1 0	0 1 1	1 0	0 1 1
64	0 0 0 0	0 0 1 0	0 0 0 1 0	1 0 0	1 0	0 1 1
65	0 0 0 0	0 0 1 0	0 0 0 1 0	0 1 0	1 0	0 1 1
66	0 0 0 0	0 0 1 0	0 0 0 0 1	0 0 0	1 0	1 1 1
67	0 0 0 0	0 0 0 1	0 0 0 1 0	0 1 1	1 1	0 1 1
68	0 0 0 0	0 0 1 0	0 0 0 0 0	1 0 1	1 0	1 1 1
69	0 0 0 1	0 0 0 1	0 0 1 0 0	1 0 0	0 1	0 1 1
70	0 0 0 0	0 0 0 1	0 0 0 0 1	0 0 1	1 1	1 1 1

Example The EMC for the GSPN of Figure 2 is presented in Figure 3, while the net markings corresponding to the various states depicted in Figure 3 are listed in Table I. In Figure 3, states corresponding to vanishing markings are depicted by single circles, while states corresponding to tangible markings are depicted by double circles. Furthermore, the part of the chain depicted in dashed lines should be inaccessible under operation by any optimal scheduling policy, either because it leads to dead/absorbing states (c.f. the relevant discussion above), or because the transitions branching to that part of the chain essentially introduce some unnecessary delay in the system operation, by deliberately idling the server. As a more concrete example of the latter case, consider state s_{30} in Figure 3, which, according to Table I, corresponds to a state where a job, j_1, in workstation W_1, having finished processing of stage J_1 requests transfer to workstation W_2, that currently contains only another job, j_2, in processing of its second stage. Moreover, the system transporter is available, and it is easy to check that the requested transfer is physically feasible and admissible by the applied SCP. Under these circumstances, deliberately idling the transporter, by firing transition T_{idle}, will definitely be a suboptimal decision, since the only way that the system can progress once job j_2 has completed the execution of its current stage, is by eventually executing the postponed transfer of job j_1 to W_2, and the overall operation of the system will have been slowed down by the corresponding unnecessary delay. The remaining modified EMC, depicted with solid lines in Figure 3, contains only two random switches of two options each, which combined with Equation 7, leaves us with two decision variables ξ_1 and ξ_2. Finally, the reader can verify that any pricing $(\xi_1, \xi_2) \in [0,1]^2$ leads to unichain behavior for the controlled system. □

IV. Obtaining an optimal scheduling policy

The solution of the MP formulation defined by Equations 2, 3, 5, 6 and 7 is a challenging problem because of (i) the non-linearity arising in Equations 2 and 3, and (ii) the additional requirement that $\xi \in UP$, which is necessary for the existence of the steady-state distribution implied by Equations 2 and 3. However, in this section, we establish that the considered formulation will always have an optimal solution which prices all primary decision variables, ξ_l, at one of their extreme values, 0 or 1, and therefore, it can be solved through enumerative techniques. From a modelling standpoint, such an optimal solution defines a *deterministic* scheduling policy. We notice that this finding is consistent with a more general result on the optimality of deterministic scheduling policies provided by the theory of Markovian Decision Processes [14]; our work provides a specialization and a complete alternative derivation for it in the GSPN modelling framework. We proceed to this development through a series of lemmata.

Lemma 1: The optimization problem defined by Equations 2, 3, 5, 6 and 7 can be transformed to an equivalent optimization problem of the form:

$$\max_{\xi \in UP} TH(\xi) = \frac{N(\xi)}{D(\xi)} \tag{8}$$

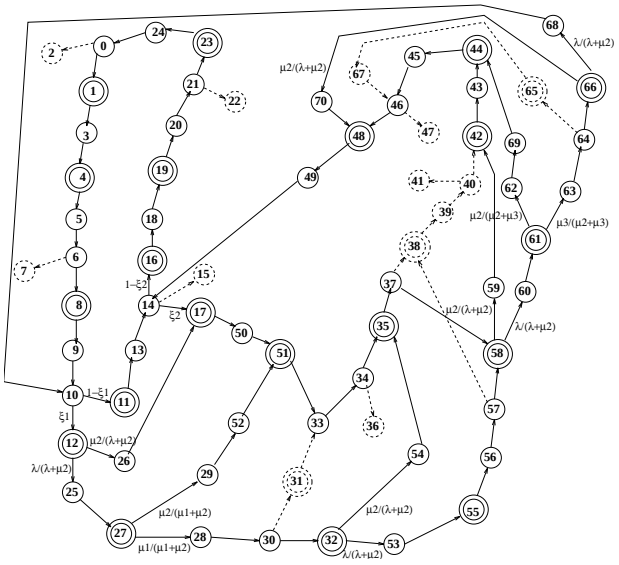

Fig. 3. Example: The Embedded Markov Chain (EMC)

s.t.

Equations (6) and (7)

where functions $N(\xi)$ and $D(\xi)$ are *multi-linear*[4] in ξ. Furthermore, $D(\xi) \neq 0,\ \forall \xi \in UP$ satisfying Equations 6 and 7.

Proof: Notice that, according to Equation 2, the variable vector $\mathbf{y}$, denoting the steady state probabilities of the net modified EMC, satisfies the linear system of equations:

$$\left[\begin{array}{c} \overline{Q}^T(\xi) \\ \mathbf{1}^T \end{array} \right] \mathbf{y} = \left[\begin{array}{c} \mathbf{0} \\ 1 \end{array} \right] \tag{9}$$

where $\mathbf{1}$ and $\mathbf{0}$ denote column vectors with all their elements equal to 1 and 0, respectively. Furthermore, the dynamic nature of random switches, assumed in this work, implies that each variable ξ_l appears in matrix $\overline{Q}^T(\xi)$ only once, namely in the column corresponding to the associated vanishing marking m. To facilitate the subsequent discussion, let us rewrite Equation 9 as

$$H\mathbf{y} = \mathbf{b} \tag{10}$$

The ergodic nature of the modified EMC defined by the considered values of the variable vector ξ, implies that the linear system of Equation 10 has a unique solution, obtained by Crammer's rule [15]:

$$\forall m_k \in R(\mathcal{N}, M_0), \quad y_k(\xi) = \frac{\det(H_k(\xi))}{\det(H(\xi))} \tag{11}$$

where matrix $H_k(\xi)$ is obtained from matrix $H(\xi)$ by replacing its k-th column by vector $\mathbf{b}$. Furthermore, the fact that each variable ξ_l appears in a single element of matrix $H(\xi)$ implies that $\forall k,\ \det(H_k(\xi))$ is a multi-linear function in ξ. But then, Equation 3 implies that for all $m_k \in R_T(\mathcal{N}, M_0)$,

$$\pi_k = \frac{E[m_k]\det(H_k(\xi))}{\sum_{m_l \in R_T(\mathcal{N},M_0)} E[m_l]\det(H_l(\xi))} = \frac{N_k(\xi)}{D(\xi)} \tag{12}$$

and $N_k(\xi)$ and $D(\xi)$ are multi-linear functions in ξ. The main result of Lemma 1 is obtained from Equation 12, by noticing that, according to Equation 5, $TH(\xi)$ is defined as the weighted sum of an appropriately selected set of π_k. The fact that $D(\xi) \neq 0$ over the considered feasible region, is established by the requirement that $\xi \in UP$, since it implies the existence of a limiting distribution for the continuous-time stochastic process modelling the time-based behavior of the controlled system. □

The next lemma establishes some additional structure for the polynomial functions $N(\xi)$ and $D(\xi)$, which is invoked in the proof of the theorem stating the main result of this section.

Lemma 2: In the multi-linear functions $N(\xi)$ and $D(\xi)$ defined in Lemma 1, there are no products of variables ξ_l belonging in the same random switch Ξ_u.

Proof: Remember that, according to the proof of Lemma 1, all variables ξ_l belonging to a single random switch Ξ_u regulating the transitions out of a vanishing marking m, appear in the same column of matrix $H(\xi)$. Then, the truth of Lemma 2 follows from the elementary definition of the det() operator [15], and the definitions of functions $N(\xi)$ and $D(\xi)$ in the proof of Lemma 1. □

Theorem 1: The MP formulation of Equations 8, 6 and 7, introduced in Lemma 1, will always have an optimal solution in which the primary decision variables, ξ_l, are priced in the set $\{0, 1\}$.

Proof: Without loss of generality, suppose that each random switch Ξ_u has $|\Xi_u| \geq 2$. Then, solving the corresponding constraint in Equation 7 for one of the involved decision variables, to be denoted by $\xi_{i(u)}$, and replacing $\xi_{i(u)}$ in the objective function by the resulting expression, we can rewrite the formulation of Equations 8, 6 and 7 in a reduced variable space, as follows:

$$\max_{\xi \in UP} TH(\xi) = \frac{\hat{N}(\xi)}{\hat{D}(\xi)} \tag{13}$$

s.t.

$$\forall \text{ random switch } \Xi_u,\ \forall l \neq i(u),\ \xi_l \geq 0.0 \tag{14}$$

$$\forall \text{ random switch } \Xi_u, \sum_{l \neq i(u): \xi_l \in \Xi_u} \xi_l \leq 1.0 \tag{15}$$

Lemma 2 implies that the functions $\hat{N}(\xi)$ and $\hat{D}(\xi)$ remain multi-linear polynomials in ξ. Then, the partial differentiation of function $TH(\xi)$ with respect to each variable ξ_l reveals that the objective function defined by Equation 13 is monotone with respect to every single variable ξ_l.

This monotonicity property of $TH(\xi)$, combined with the fact that for any $\xi \in UP$ such that $\forall l,\ \xi_l \in (0, 1)$, $\exists \delta > 0$ such that $\forall$ unit radius $\mathbf{r},\ \xi + \delta\mathbf{r} \in UP$, further imply that there exists an optimal solution of the formulation defined by Equations 13, 14 and 15 that lies on the boundary of its feasible region. Hence, any such optimal solution $\xi^* \in UP$ must bind at least one of the Constraints 14 and 15, for each random switch Ξ_u. Therefore, $\forall\ \Xi_u$, either $\exists\ l \neq i(u) :\ \xi_l = 0$ (if one of the equations defined by Constraint 14 is bounded), or $\xi_{u(i)} = 0$ (i.e., Constraint 15 is bounded). In order to price the remaining free variables ξ_l, (i) we remove the variables priced to zero from the set of variables engaged by the original formulation of Equations 8, 6 and 7, and furthermore, (ii) we set equal to one all variables ξ_l that belong to a random switch Ξ_u which constitutes a singleton (set) after the variable elimination of Step (i). The resulting formulation preserves the structure of the original one of Equations 8, 6 and 7, but it engages a reduced set of variables. Hence, the truth of Theorem 1, is established by repetitively applying the entire argument developed above on this reduced formulation and all the subsequent formulations derived from it, while taking into consideration the finiteness of the initial sets Ξ_u. □

We notice that a solution of the type defined in Theorem 1, corresponding to a deterministic scheduling policy for the underlying GSPN, constitutes an *extreme point* [16] for the polyhedron defined by Equations 6 and 7. The next example demonstrates how the result of Theorem 1 facilitates the computation

[4] i.e., first-degree polynomials with respect to each single variable ξ_l

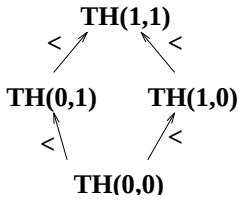

Fig. 4. Example: Characterizing the dominance among the candidate scheduling policies

of an optimal scheduling policy for any given instance from the considered GSPN class, through an enumerative approach that terminates in a finite number of steps.

Example Theorem 1 implies that an optimal scheduling policy for the modified EMC of Figure 3 can be obtained by (i) computing, through Equations 2 – 5, the closed-form expressions for $TH(0,0)$, $TH(0,1)$, $TH(1,0)$ and $TH(1,1)$, and (ii) determining the parameter ranges over which each of these expressions dominates the others. Working according to this plan, one can establish that the dominance relationships among these four expressions are those depicted by the lattice of Figure 4. □

The reader can verify that the optimal policy, defined by $(\xi_1 = 1, \xi_2 = 1)$, essentially implements the *First-Buffer-First-Serve (FBFS)* [17] policy on the re-entrant line of Figure 1. On the other hand, the *Last-Buffer-First-Serve (LBFS)* [17] policy corresponds to the deterministic scheduling policy defined by $(\xi_1 = 1, \xi_2 = 0)$, and as it is shown in Figure 4, it is a suboptimal policy. This result is drastically different from the situation applying to the original model of uncapacitated re-entrant lines, where the LBFS policy has been shown to be optimal – i.e., it maximizes the long-run system throughput – over all possible configurations [17]. Hence, this example and the overall analysis pursued in this work corroborate the findings of the work presented in [4], and establish the fundamental difference between the structure of the optimal scheduling policies in capacitated and uncapacitated re-entrant lines, under a stochastic operational regime which is broader than the deterministic case considered in [4].

Concluding this section, we notice that the result of Theorem 1, regarding the existence of a deterministic optimal scheduling policy, can be immediately generalized to any other MP formulation obtained from that of Equations 2, 3, 5, 6 and 7, by replacing Equation 5 by any other weighted sum of the steady-state probabilities π_k. Such an objective can be, for instance, the minimization of the average Work-In-Process, $\overline{WIP}$, of the re-entrant line under steady-state operation, defined by:

$$\overline{WIP} = \sum_k \pi_k \cdot WIP(s_k) \tag{16}$$

where $WIP(s_k)$ denotes the number of parts loaded in the system in state s_k. In fact, the result of Theorem 1 applies also to the objective of minimizing the job average sojourn time, $\overline{\tau}$, since (i) by Little's law, this quantity can be expressed by

$$\overline{\tau} = \frac{\overline{WIP}}{TH} \tag{17}$$

and (ii) Equations 5 and 16 imply that, when expressed as fractional functions of ξ, both quantities $\overline{WIP}$ and TH have the same denominator $D(\xi)$ defined in Equation 12.

V. Conclusions

The starting point for this work was the observation that the increasing level of automation in modern semiconductor fabs necessitates a more detailed modelling and analysis of their real-time operations, while the super-imposition of the appropriate supervisory control logic invalidates the previous analytical studies regarding the performance modelling and control of these environments. As a result, the presented work proposed a novel modelling and analysis framework for these systems, which is based on the formal tool of Generalized Stochastic Petri net, and allows the seamless integration of the fab logical and timed dynamics in a single representation. Furthermore, the proposed framework supports the analytical representation of the fab scheduling problem as a Mathematical Programming formulation, which can be effectively solved to optimality through enumerative techniques. The framework presentation and its capabilities were elucidated by detailed application on a small system configuration. However, a severe limitation of the presented approach is that it requires the explicit enumeration of the underlying state space, which explodes very fast. Therefore, part of our future work seeks to develop novel approximating schemes, based on the characterizations and insights provided by this work, that will lead to (near-)optimal scheduling policies for modern fabs, while maintaining computational tractability.

Acknowledgment

This work was partially supported by NSF grant ECS-9979693 and by The Logistics Institute Asia Pacific.

References

[1] P. R. Kumar, "Scheduling manufacturing systems of re-entrant lines," in *Stochastic Modeling and Analysis of Manufacturing Systems*, D. D. Yao, Ed., pp. 325–360. Springer-Verlag, 1994.

[2] S. Kumar and P. R. Kumar, "Queueing network models in the design and analysis of semiconductor wafer fabs," *IEEE Trans. on R&A*, vol. 17, pp. 548–561, 2001.

[3] J. Park, S. A. Reveliotis, D. Bodner, C. Zhou, J.-F. Wu, and L. McGinnis, "High-fidelity rapid prototyping of 300mm fabs through discrete event system modeling," *Computers in Industry : invited paper for the special issue on MASM'2000*, vol. 1528, pp. 1–20, 2001.

[4] S. A. Reveliotis, "The destabilizing effect of blocking due to finite buffering capacity in multi-class queueing networks," *IEEE Trans. on Autom. Control*, vol. 45, pp. 585–588, 2000.

[5] M. A. Marsan, G. Conte, and G. Balbo, "A class of generalized stochastic petri nets for performance evaluation of multiprocessor systems," *ACM Trans. Comput. Sys.*, vol. 2, pp. 93–122, 1984.

[6] M. A. Marsan, G. Balbo, and G. Conte, *Performance Models of Multiprocessor Systems*, The MIT Press, Cambridge, MA, 1986.

[7] N. Viswanadham and Y. Narahari, *Performance Modeling of Automated Manufacturing Systems*, Prentice Hall, Englewood Cliffs, NJ, 1992.

[8] A. A. Desrochers and R. Y. Al-Jaar, *Applications of Petri nets in Manufacturing Systems*, IEEE Press, Piscataway, NJ, 1995.

[9] M. Zhou and M. Jeng, "Modeling, analysis, simulation, scheduling and control of semiconductor manufacturing systems: A petri net approach," *IEEE Trans. on Semiconductor Manufacturing*, vol. 11, pp. 333–357, 1998.

[10] M. Jeng, X. Xie, and W.-Y. Hung, "Markovian timed petri nets for performance analysis of semiconductor manufacturing systems," *IEEE Trans. on Systems, Man and Cybernetics – Part B*, vol. 30, pp. 757–771, 2000.

[11] W. M. Zuberek, "Timed petri nets in modeling and analysis of cluster tools," *IEEE Trans. on Robotics and Automation*, vol. 17, pp. 562–575, 2001.

[12] H. T. Papadopoulos, C. Heavy, and J. Browne, *Queueing Theory in Manufacturing Systems Analysis and Design*, Chapman & Hall, New York, NY, 1993.

[13] Z. A. Banaszak and B. H. Krogh, "Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows," *IEEE Trans. on Robotics and Automation*, vol. 6, pp. 724–734, 1990.

[14] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, 1994.

[15] G. Strang, *Linear Algebra and its Applications, 3rd. Ed.*, Harcourt College Pub., 1988.

[16] V. Chvátal, *Linear Programming*, W. H. Freeman & Co., N.Y., N.Y., 1983.

[17] S. H. Lu and P. R. Kumar, "Distributed scheduling based on due dates and buffer priorities," *IEEE Trans. on Aut. Control*, vol. 36, pp. 1406–1416, 1991.