

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

Jin Young Choi and Spyros A. Reveliotis*
 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*

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, \dots, W_m . Each workstation W_i , $i = 1, \dots, m$, possesses S_i identical servers, and the production of each unit occurs in n stages, J_1, J_2, \dots, J_n , with stage J_j , $j = 1 \dots, n$, 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].

¹Due 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.

* corresponding author

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]. Since their inception, timed PN models, in general, and GSPN models, in particular, have been strongly advocated for the modelling and performance evaluation of production systems, primarily due to the clarity and immediacy of their semantics, that provides a simple and flexible interface between the production systems design and control problem, and the underlying concepts and results borrowed from the more general theory of stochastic systems. We refer the reader to [7], [8] for an extensive coverage of the use of timed PN models for manufacturing system modelling and analysis until the middle 1990's. More recently, the works of [9], [10] and [11] have used timed PN's for the performance evaluation of the production activity taking place in the more specific context of modern semiconductor manufacturing facilities. The first of these works [9] discusses the fundamentals for the modelling and performance evaluation of the fab operations through timed PN's, in the context of a broader survey on the (potential) use of the PN modelling framework in semiconductor manufacturing. The work of [10] adjusts to a class of Markovian Timed PN's modelling the operations of a semiconductor manufacturing re-entrant line, a methodology for the computation of performance bounds, that was originally developed in [12] for multi-class queueing networks. Finally, the work of [11] uses ideas and results from the theory of deterministic marked graphs [13] in order to compute performance measures of various cluster tool [14] configurations, operated under pre-specified scheduling policies.

Contrary to these past works that have primarily addressed the descriptive problem of the system performance evaluation, under a pre-selected set of scheduling policies, the work presented in this paper seeks to exploit the modelling and analytical power of the GSPN framework in order to address the prescriptive problem of computing a scheduling policy that optimizes a pre-specified performance objective. Hence, the first part of the presented work develops a canonical model for the operation of the capacitated re-entrant line that allows the seamless integration of the logical control requirements in the more traditional problem of fab performance-oriented control, and the systematic investigation of the nature of optimal scheduling policies in this new operational context. Specifically, the proposed model can support the explicit representation of all the logical/structural features of the fab operations that can drastically affect the performance index under consideration, as identified in the work of [3]. On the other hand, its GSPN nature allows the analytical characterization of the underlying performance control problem as a mathematical programming (MP) formulation, through the theory of semi-Markov processes [6]. This MP formulation is investigated in the second part of the paper, which identifies an important structural property of the space of optimal scheduling policies, and employs it towards the de-

velopment of an exact algorithm for the derivation of an optimal solution with finite computational effort.

A limitation of the developed results arises from the fact that, similar to most typical analytical approaches emerging from the GSPN modelling framework, they require the explicit enumeration of the system state space; therefore, their applicability to "real-life" systems is constrained by computational considerations. Yet, the ability to analytically characterize and, in certain cases, explicitly compute the optimal scheduling policy for these new fab models, in spite of their increased operational complexity, is of paramount practical importance, since it can provide the necessary analytical insight for the eventual development of pertinent approximating algorithms / heuristics with more favorable computational properties. The design of such efficient near-optimal approximating schemes is part of our current work.

The rest of the paper is organized as follows: Section II introduces the key features of the GSPN modelling framework, and overviews the basic technique for the performance evaluation of the resulting models through the analysis of the underlying semi-Markov process. Section III introduces the structurally controlled capacitated re-entrant line, and proceeds to its systematic modelling in the GSPN framework. The last part of this section provides an analytical characterization of the re-entrant line scheduling problem, by developing the relevant MP formulation. Section IV first establishes an important property for the structure of the optimal solution space of the MP formulation developed in Section III, and subsequently it exploits this property towards the synthesis of an algorithm that computes an optimal solution in a finite number of steps. Finally, Section V concludes the paper and suggests directions for future work. Throughout the paper development, the derived results are elucidated by application on a small but pertinent example system.

II. GENERALIZED STOCHASTIC PETRI NETS AND THEIR PERFORMANCE EVALUATION

This section provides a brief introduction to the modelling framework of *Generalized Stochastic Petri nets (GSPN)* [5], [6], and the way in which it supports the performance evaluation of the modelled system. In the following discussion it is assumed that the reader is familiar with the basic Petri net structural concepts, and the emphasis is placed on the modelling elements and techniques concerning the time-related aspects of the system behavior. We refer the reader to [15] for an introduction to the basic PN theory.

The GSPN modelling framework According to [6], a Generalized Stochastic Petri net (GSPN) is defined as a PN $\mathcal{N} = (P, T, W, M_0)$ with its transition set T partitioned into two sub-sets T_I and T_T , defining respectively the set of *immediate* and *timed* transitions. Immediate transitions fire in zero time, once they are enabled, whereas, timed transitions fire after a random, *exponentially* distributed, enabling time. Hence, in order to complete the formal definition of a GSPN \mathcal{N} , transitions $t \in T_T$ are associated

with a (possibly marking-dependent) *firing rate*, $r(t)$, that constitutes the defining parameter of the corresponding exponential distribution.

The above characterization of immediate and timed transitions implies that in a net reachable marking, m , where, both, immediate and timed transitions are enabled, immediate transitions have precedence over the timed ones (since they fire instantaneously). Furthermore, such a marking m has zero duration in the net dynamics, and therefore, it is characterized as *vanishing*. On the other hand, a marking m in which all enabled transitions are timed transitions, has an expected duration $E(m) = 1/\sum_{t \in T(m)} r(t)$, where $T(m)$ denotes the set of enabled timed transitions in marking m ; therefore, such a marking is characterized as *tangible*. In the following, given a GSPN net \mathcal{N} with initial marking M_0 , the set of reachable tangible markings will be denoted by $R_T(\mathcal{N}, M_0)$ and the set of reachable vanishing markings will be denoted by $R_V(\mathcal{N}, M_0)$. Obviously, the set of reachable markings $R(\mathcal{N}, M_0) = R_T(\mathcal{N}, M_0) \cup R_V(\mathcal{N}, M_0)$.

The exponential nature of the firing times of the transitions enabled in a tangible marking m defines also an *arbitration* mechanism for their firing, i.e., each of these transitions will fire first with probability $r(t)/\sum_{t \in T(m)} r(t)$. On the other hand, in a vanishing marking with more than one enabled immediate transitions, the contest of these transitions for firing must be arbitrated through some externally imposed logic. Specifically, given a marking m with a set of simultaneously enabled immediate transitions, $I(m)$, the modeler must provide a probability distribution regulating the firing of the transitions in $I(m)$. In the GSPN terminology, this probability distribution is characterized as a *random switch* $\Xi = \{\xi_1, \xi_2, \dots, \xi_{I(m)}\}$. Furthermore, if the set of random switches regulating the net behavior are marking-dependent, they are characterized as *dynamic*; otherwise, they are *static*.

According to [6], the motivation for the introduction of the immediate transitions in the net model is the desire to focus the time-related characterization of its behavior on those activities that have the significantly longest durations, and therefore, the strongest impact on the system performance. By placing the emphasis on the events with the longest timings and their associated tangible markings, the computational cost for the performance evaluation of the considered system is significantly reduced. We believe that another important modelling feature, resulting from the presence of immediate transitions in the GSPN modelling framework, is the ability to naturally separate the modelling of the control function from the modelling of the net dynamics corresponding to the various physical processes. Specifically, the imposition of various control commands on the underlying (production) system are logical events which can be modelled by immediate transitions, whereas, the events corresponding to processing, transport or staging activities resulting from the physical execution of these commands, are more appropriately modelled by timed transitions. This separation of the control function from the physical system behavior is further exemplified in

Section III.

Performance evaluation of GSPN models The limitation of the timing models regulating the firing of the net transitions to immediate transitions and transitions with exponentially distributed enabling time, implies that the stochastic process modelling the time-based behavior of a GSPN \mathcal{N} – i.e., its time-stamped sample-paths among its set of reachable markings $R(\mathcal{N}, M_0)$ – is a *semi-Markov process* [6]. In particular, 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 transition probability matrix $Q = [q_{kl}]$ is determined by the externally specified random switches, in the case of vanishing markings, and the exponential race of the enabled events, in the case of tangible markings. If this EMC is finite-state, homogeneous and irreducible, it possesses a steady-state distribution \mathbf{y} , obtained by the system of equations

$$\mathbf{y}^T = \mathbf{y}^T \cdot Q; \quad \sum_{m_k \in R(\mathcal{N}, M_0)} y_k = 1 \quad (1)$$

The availability of the EMC steady-state distribution, \mathbf{y} , subsequently allows the computation of the steady-state probabilities, π_k , characterizing the timed system behavior, through the following formula [6]:

$$\pi_k = \begin{cases} 0, & m_k \in R_V(\mathcal{N}, M_0) \\ \frac{y_k E[m_k]}{\sum_{m_l \in R_T(\mathcal{N}, M_0)} y_l E[m_l]}, & m_k \in R_T(\mathcal{N}, M_0) \end{cases} \quad (2)$$

Notice that, as expected, the steady state probability, π_k , is equal to zero for all reachable vanishing markings $m_k \in R_V(\mathcal{N}, M_0)$. On the other hand, the second branch of Equation 2 indicates that the percentage of time that the system spends in a reachable tangible marking m_k , is a function of the relative frequency with which this state is visited, determined from the untimed system dynamics, and the expected times that it spends in each reachable tangible marking. Once the steady-state probability vector π has been obtained, various performance measures of interest, characterizing the long-run system behavior, can be defined as appropriate functions of π and the other system parameters.

Extending the GSPN-based modelling to systems with non-Markovian dynamics We conclude this section with another remark regarding the modelling and analytical power of GSPN's. As it was mentioned above, the ability to evaluate analytically the steady-state probabilities characterizing the long-run system behavior is established on the requirement that all the distributions regulating the firing of the net timed transitions are of exponential type. This would seem quite restrictive since in many "real-life" environments, including those related to semiconductor manufacturing, the actual event timings might not be exponentially distributed. However, this restriction can be circumvented, whenever the exponential distribution is deemed as a too unrealistic assumption, by substituting each timed transition in the original GSPN model

with a GSPN subnet modelling a *phase-type* distribution, that approximates the timing distribution of the replaced transition to any desired degree of accuracy, without compromising the GSPN structure of the final model. We refer the reader to [16] for a detailed treatment of phase-type distributions and the relevant approximation theory.

III. THE STRUCTURALLY CONTROLLED CAPACITATED RE-ENTRANT LINE AND ITS GSPN-BASED MODELLING

The structurally controlled capacitated re-entrant line The capacitated re-entrant line considered in this work refines the basic re-entrant line model, presented in the introductory section, through the explicit modelling 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 , $i = 1, \dots, m$, 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 system 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 modelling 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 modelling 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 , $j = 1, \dots, n$, 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 uni-

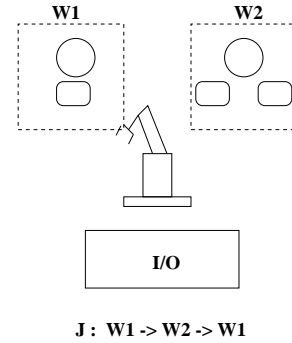


Fig. 1. Example: The capacitated re-entrant line

formity 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 $\lambda \rightarrow \infty$ in the derived expressions. Finally, we remind the reader that the rather unrealistic assumption of exponentially distributed processing and transfer times can be eventually relaxed in the resulting GSPN model through the approximating scheme based on phase-type distributions, discussed in the previous section.

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 $J = \langle J_1, J_2, J_3 \rangle$, 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

$$|J_1| + |J_2| \leq C_1 + C_2 - 1 = 2 \quad (3)$$

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

GSPN-based modelling of structurally controlled capacitated re-entrant lines The GSPN modelling the behavior of the capacitated re-entrant line of Figure 1, under the control of the maximally permissive structural control policy (SCP) of Equation 3, 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 modelled by the corresponding net path $\langle T_{ja}, P_{jt}, T_{jt}, P_{ji}, T_{jl}, P_{jp}, T_{jp}, P_{jo}, T_{jd} \rangle$, 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}

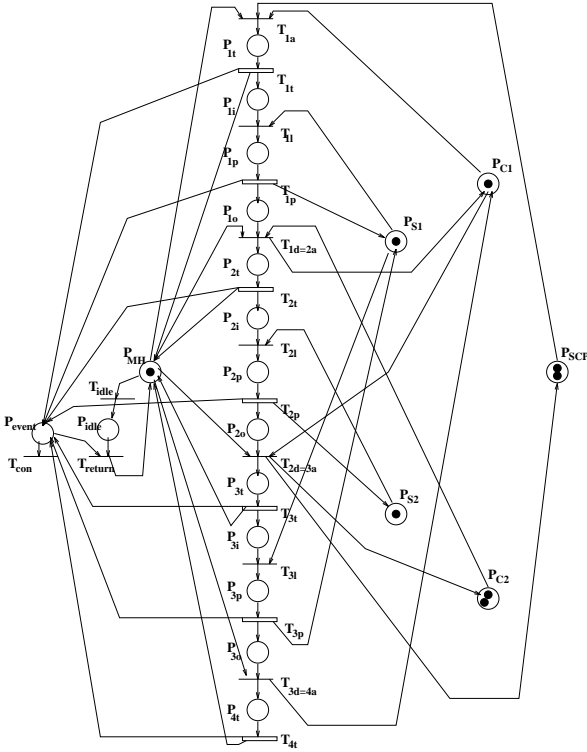


Fig. 2. Example: The GSPN model

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, modelling practice of resource-process nets [17]. Notice that transitions T_{ja} , T_{jl} and T_{jd} , which are associated with the various decisions regarding the allocation of the system buffering, processing and/or transport capacity, are immediate transitions, while the delays experienced from the processing and/or transfer times involved with the execution of these decisions, are modelled by the timed transitions T_{jt} and T_{jp} .

As mentioned in the previous section, the untimed nature of the transitions modelling the material flow control function reflects the immediate impact of the ongoing decision making on the logical resource allocation state representation, maintained by the system controller, and allows the explicit identification of the control function in the overall net structure. Specifically, under the proposed representation, the set of the reachable vanishing markings, $R_V(\mathcal{N}, M_0)$, define the points of the controller intervention in the system operation, while the set of the *dynamic random switches* that resolve the conflicts among the immediate transitions simultaneously enabled in those markings express the scheduling logic imposed by the system controller.

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-modelling 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.

GSPN-based performance optimization of structurally controlled capacitated re-entrant lines 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 remaining switching probabilities, ξ_i , constitute the decision variables of the scheduling problem, and they must be priced in a way that optimizes the performance objective under consideration. Letting $\bar{Q}(\xi)$ denote the transition probability matrix (TPM) of the modified EMC, resulting from the removal of any absorbing states, the problem of maximizing throughput for a capacitated re-entrant line can be formally expressed by the following MP formulation:

$$\max_{\xi} TH(\xi) \equiv \sum_{(k,t)} \pi_k r(t) I \left\{ \begin{array}{l} \text{Transition } t \text{ enabled in } m_k \\ \wedge \text{ cor. to an unloading event} \end{array} \right\} \quad (4)$$

s.t.

$$\forall l, \xi_l \geq 0 \quad (5)$$

$$\forall \text{ random switch } \Xi_u, \sum_{l: \xi_l \in \Xi_u} \xi_l = 1.0 \quad (6)$$

and

$$\text{Equations (1) and (2)}$$

applied over the modified EMC.

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, at the end of the document. 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

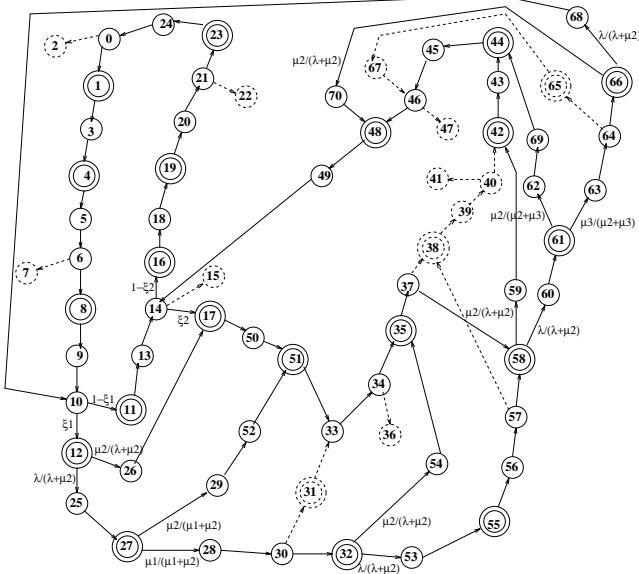


Fig. 3. Example: The Embedded Markov Chain (EMC)

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 6, leaves us with two decision variables ξ_1 and ξ_2 .

IV. OBTAINING AN OPTIMAL SCHEDULING POLICY

The solution of the MP formulation defined by Equations 1, 2, 4, 5 and 6 is a challenging problem because of the non-linearity arising in Equations 1 and 2. In particular, the products of the primary variables ξ_l with the auxiliary variables y_k appearing in Equation 1, give to the overall formulation a *bilinear* structure, which, in general, is very hard to solve for global optimality. 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 [18]; 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 1, 2, 4, 5 and 6 can be transformed to an equivalent optimization problem of the form:

$$\max_{\xi} TH(\xi) = \frac{N(\xi)}{D(\xi)} \quad (7)$$

s.t.

$$\text{Equations (5) and (6)}$$

where functions $N(\xi)$ and $D(\xi)$ are *multi-linear*² in ξ . Furthermore, $D(\xi) \neq 0$, $\forall \xi$ satisfying Equations 5 and 6.

Proof: Notice that, according to Equation 1, the variable vector \mathbf{y} , denoting the steady state probabilities of the net modified EMC, satisfies the linear system of equations:

$$\begin{bmatrix} \overline{Q}^T(\xi) \\ \mathbf{1}^T \end{bmatrix} \mathbf{y} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \quad (8)$$

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 8 as

$$H\mathbf{y} = \mathbf{b} \quad (9)$$

The ergodic nature of the modified EMC defined by every feasible value of the variable vector ξ , implies that the linear system of Equation 9 has a unique solution, obtained by Cramer's rule [19]:

$$\forall m_k \in R(\mathcal{N}, M_0), \quad y_k(\xi) = \frac{\det(H_k(\xi))}{\det(H(\xi))} \quad (10)$$

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 2 implies that for all $m_k \in R_T(\mathcal{N}, M_0)$,

$$\pi_k = \frac{E[m_k] \det(H_k(\xi))}{\sum_{m_l \in R_T(\mathcal{N}, M_0)} E[m_l] \det(H_l(\xi))} = \frac{N_k(\xi)}{D(\xi)} \quad (11)$$

and $N_k(\xi)$ and $D(\xi)$ are multi-linear functions in ξ . The main result of Lemma 1 is obtained from Equation 11, by

²i.e., first-degree polynomials with respect to each single variable ξ_l

noticing that, according to Equation 4, $TH(\xi)$ is defined as the weighted sum of an appropriately selected set of π_k . The fact that $D(\xi) \neq 0$ in the entire feasible region defined by Equations 5 and 6, is established by the existence of a limiting distribution for the continuous-time stochastic process modelling the time-based behavior of the GSPN under consideration. \diamond

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 $\det(\cdot)$ operator [19], and the definitions of functions $N(\xi)$ and $D(\xi)$ in the proof of Lemma 1. \diamond

Theorem 1: The MP formulation of Equations 7, 5 and 6, 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 6 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 7, 5 and 6 in a reduced variable space, as follows:

$$\max_{\xi} TH(\xi) = \frac{\hat{N}(\xi)}{\hat{D}(\xi)} \quad (12)$$

s.t.

$$\forall \text{ random switch } \Xi_u, \forall l \neq i(u), \xi_l \geq 0 \quad (13)$$

$$\forall \text{ random switch } \Xi_u, \sum_{l \neq i(u): \xi_l \in \Xi_u} \xi_l \leq 1.0 \quad (14)$$

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 12 is monotone with respect to every single variable ξ_l , and therefore, the optimal solution of the formulation defined by Equations 12, 13 and 14 must be on the boundary of its feasible region. This further implies that any optimal solution ξ^* must bind at least one of the Constraints 13 and 14, 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 13 is bounded), or $\xi_{u(i)} = 0$ (i.e., Constraint 14 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 7, 5 and 6, and furthermore, (ii) we set equal to one all

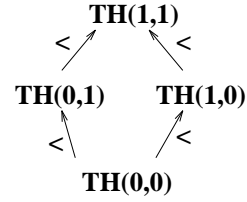


Fig. 4. Example: Characterizing the dominance among the candidate scheduling policies

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 7, 5 and 6, 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 . \diamond

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* [20] for the polyhedron defined by Equations 5 and 6. The next example demonstrates how the result of Theorem 1 facilitates the computation 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 1 – 4, 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 a *First-Buffer-First-Serve (FBFS)* [21] logic in the considered operational context. On the other hand, the *Last-Buffer-First-Serve (LBFS)* [21] 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 [21]. 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].

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] S. Kumar and P. R. Kumar, "Performance bounds for queueing networks and scheduling policies," *IEEE Trans. Autom. Control*, vol. 39, pp. 1600–1611, 1994.
- [13] J. Sifakis, *Use of Petri nets for Performance Evaluation*, North Holland, Amsterdam, Netherlands, 1977.
- [14] P. Singer, "The driving forces in cluster tool development," *Semiconductor International*, vol. July '95, pp. 113–118, 1995.
- [15] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, pp. 541–580, 1989.
- [16] H. T. Papadopoulos, C. Heavy, and J. Browne, *Queueing Theory in Manufacturing Systems Analysis and Design*, Chapman & Hall, New York, NY, 1993.
- [17] 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.
- [18] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, 1994.
- [19] G. Strang, *Linear Algebra and its Applications*, 3rd. Ed., Harcourt College Pub., 1988.
- [20] V. Chvátal, *Linear Programming*, W. H. Freeman & Co., N.Y., N.Y., 1983.
- [21] 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.

TABLE I
EXAMPLE: THE EMC MARKINGS

s_k	$P_{1t}P_{1i}P_{1p}P_{1o}$	$P_{2t}P_{2i}P_{2p}P_{2o}$	$P_{3t}P_{3i}P_{3p}P_{3o}P_{At}$	$P_{MH}P_{idle}P_{event}$	PS_1PS_2	$PC_1PC_2P_{SCP}$
0	0000	0000	00000	100	11	122
1	1000	0000	00000	000	11	021
2	0000	0000	00000	010	11	122
3	0100	0000	00000	101	11	021
4	0010	0000	00000	010	01	021
5	0001	0000	00000	011	11	021
6	0001	0000	00000	100	11	021
7	0001	0000	00000	010	11	021
8	0000	1000	00000	000	11	111
9	0000	0100	00000	101	11	111
10	0000	0010	00000	100	10	111
11	0000	0010	00000	010	10	111
12	1000	0010	00000	000	10	010
13	0000	0001	00000	011	11	111
14	0000	0001	00000	100	11	111
15	0000	0001	00000	010	11	111
16	0000	0000	10000	000	11	022
17	1000	0001	00000	000	11	010
18	0000	0000	01000	101	11	022
19	0000	0000	00100	010	01	022
20	0000	0000	00010	011	11	022
21	0000	0000	00010	100	11	022
22	0000	0000	00010	010	11	022
23	0000	0000	00001	000	11	122
24	0000	0000	00000	101	11	122
25	0100	0010	00000	101	10	010
26	1000	0001	00000	001	11	010
27	0010	0010	00000	010	00	010
28	0001	0010	00000	011	10	010
29	0010	0001	00000	011	01	010
30	0001	0010	00000	100	10	010
31	0001	0010	00000	010	10	010
32	0000	1010	00000	000	10	100
33	0001	0001	00000	011	11	010
34	0001	0001	00000	100	11	010
35	0000	1001	00000	000	11	100
36	0001	0001	00000	010	11	010
37	0000	0101	00000	101	11	100
38	0000	0011	00000	010	10	100
39	0000	0002	00000	011	11	100
40	0000	0002	00000	100	11	100
41	0000	0002	00000	010	11	100
42	0000	0001	10000	000	11	011
43	0000	0001	01000	101	11	011
44	0000	0001	00100	010	01	011
45	0000	0001	00010	011	11	011
46	0000	0001	00010	100	11	011
47	0000	0001	00010	010	11	011
48	0000	0001	00001	000	11	111
49	0000	0001	00000	101	11	111
50	0100	0001	00000	101	11	010
51	0010	0001	00000	010	01	010
52	0010	0001	00000	100	01	010
53	0000	0110	00000	101	10	100
54	0000	1001	00000	001	11	100
55	0000	0110	00000	010	10	100
56	0000	0101	00000	011	11	100
57	0000	0011	00000	100	10	100
58	0000	0010	10000	000	10	011
59	0000	0001	10000	001	11	011
60	0000	0010	01000	101	10	011
61	0000	0010	00100	010	00	011
62	0000	0001	00100	011	01	011
63	0000	0010	00010	011	10	011
64	0000	0010	00010	100	10	011
65	0000	0010	00010	010	10	011
66	0000	0010	00001	000	10	111
67	0000	0001	00010	011	11	011
68	0000	0010	00000	101	10	111
69	0000	0001	00100	100	01	011
70	0000	0001	00001	001	11	111