Performance Optimization for a Class of Generalized Stochastic Petri Nets

Ran Li and Spyros Reveliotis
School of Industrial & Systems Engineering
Georgia Institute of Technology
{rli63,spyros}@isye.gatech.edu

***Abstract*— This paper considers the problem of optimizing the (long-term) performance of operations that are modeled by Generalized Stochastic Petri nets. The proposed methodology employs the representational power of the GSPN framework in order to articulate an explicit trade-off between the computational tractability of the formulated problem and the operational efficiency of the derived solutions. On the other hand, the solution of the considered formulations is based on recent results regarding the sensitivity analysis of Markov reward processes. A more expansive treatment of the presented results, together with a case study that highlights the relevance of the considered problem and the efficacy of the proposed methodology, can be found in a companion document that is accessible from the website of the second author.**

I. Introduction

Generalized stochastic Petri nets (GSPNs) are one of the best known and most extensively used models in the class of *timed* PN models. The defining element of these nets is that they differentiate their transitions into two classes, respectively known as "timed" and "untimed". Timed transitions present non-zero firing times drawn from exponential distributions, while untimed transitions have zero firing times. In [1], the seminal paper that introduced the GSPN model, but also in most of the remaining works in the GSPN-related literature [2], untimed transitions have been promoted as a "mechanism" for controlling the complexity of the time-based analysis of the corresponding PN models, while preserving all the behavioral traits of the underlying system that are captured by these transitions. In this paper, we consider a different role for the untimed transitions. More specifically, in our models, untimed transitions constitute "decisions" that are to be determined by an external control function in an effort to govern the system dynamics. Similar to the standard GSPN modeling framework, conflicting decisions at any given marking are resolved by the specification of a *"random switch"*, i.e., a probability distribution that regulates the selection of these decisions. But while in the past applications of the GSPN model the necessary random switches are specified by the human modeler so that they reflect properly the (intended) operation of the underlying system, in this work, the probabilities defining these random switches are "decision variables" to be determined so that certain performance objective(s) are optimized, by the techniques to be developed herein.[1]

From a more practical standpoint, the developments presented in this work have been motivated by the need to address the scheduling (especially the throughput maximization) problem of sequential resource allocation systems (RAS) that are controlled for deadlock-freedom, liveness and reversibility according to the supervisory control theory presented in [4], [5]. In [6], [4], it has been shown that this scheduling problem can be modeled as an average-reward Markov Decision Process (MDP), which is well-defined and effectively solvable due to the structural properties that are established by the aforementioned liveness-enforcing supervision (LES). On the other hand, the solvability of this MDP is practically limited by the state-space explosion that is typical for the considered RAS. As we shall show in the following, the GSPN modeling framework enables a succinct characterization of the aforementioned MDP problem, and it also facilitates a pertinent resolution of this problem through its restriction to certain policy spaces that are computationally tractable, and are naturally suggested by the GSPN modeling framework itself.

From a methodological standpoint, our results are enabled by fairly recent results concerning the sensitivity analysis of finite-space Markov reward processes and the computation of performance gradients with respect to (w.r.t.) various control variables of these processes through a sample-path-based analysis [7], [8]. In the context of our work, the stochastic processes of interest are the uniformized versions of the continuous-time Markov chains (CTMCs) that abstract the timed dynamics of the considered GSPN models, while the control variables involved are the probabilities that define the various random switches. The eventual integration of the gradient estimation capability described above with some stochastic approximation algorithms [9] enables the determination of a set of random switches that will optimize the performance criteria of interest, within the scope of the considered class of policies.

Having outlined the main perspectives and the intended contribution of this work in the previous paragraphs, the rest of this paper is organized as follows: The next section provides an overview of the GSPN modeling framework and of the steady-state performance evaluation of the

This work was partially supported by NSF grant CMMI-0928231.

[1] We notice, for completeness, that the work presented in [3] employs the GSPN modeling framework for representing decision-making processes that are conceptually similar to those considered in this work. However, the modeling details in the two works and the pursued approaches for the simplification and solution of the derived analytical formulations are fundamentally different.

GSPN models through their reduction to pertinent semi-Markov and Markov processes. Section III defines the MDP problem considered in this work, by further qualifying the structure of the considered GSPN models. Section IV discusses the solution of the MDP problem formulated in Section III through sensitivity analysis of the underlying stochastic processes and the integration of these results in stochastic approximation algorithms. Finally, Section V concludes the paper and proposes directions for future work.

Closing this introductory discussion, we must also notice that the imposed space limitations for this paper do not allow a full-blown development of the presented material. A more expansive and thorough coverage of the presented results, together with a case study that highlights the relevance of the considered problem and the efficacy of the proposed methodology, can be found in [10]; this document is available from the personal website of the second author.

II. Generalized Stochastic Petri Nets

In this section we provide an overview of the basic GSPN theory. In the subsequent discussion we assume that the reader is familiar with the basic PN modeling framework, and therefore, we focus primarily on the time-related aspects of the GSPN model and its analysis. An excellent introduction to the basic PN modeling framework can be found in [11].

Following [1], we define a *Generalized Stochastic Petri Net (GSPN)* as a Petri net system $\mathcal{N} = (P, T, W, m_0)$ where the transition set T is partitioned in two subsets T_t and T_u denoting respectively the sets of *timed* and *untimed* transitions. Furthermore, there is a mapping $R\colon T_t \to \mathbb{R}^+$ with $r_t \equiv R(t)$, $t \in T_t$, denoting the (instantaneous hazard) rate of the exponential distribution that determines the *firing times* for transition t. These firing times are interpreted as a delay between the enabling of transition t and the actual firing of the transition. On the other hand, transitions $t \in T_u$ have zero firing times, i.e., these transitions can fire as soon as they are enabled.

Let $\mathcal{R}(\mathcal{N})$ denote the reachability space of net $\mathcal{N}$. The firing dynamics of the GSPN $\mathcal{N}$ considered in [1] are further qualified by the following assumptions: (i) $|\mathcal{R}(\mathcal{N})| < \infty$, and (ii) for any marking $m \in \mathcal{R}(\mathcal{N})$, the set of enabled transitions $\mathcal{E}(m)$ is non-empty (in other words, the considered GSPN does not contain any total deadlocks). For any marking $m \in \mathcal{R}(\mathcal{N})$, let $\mathcal{E}_t(m)$ and $\mathcal{E}_u(m)$ denote respectively the subsets of the timed and untimed transitions enabled in m, and suppose first that $\mathcal{E}_u(m) = \emptyset$ and $\mathcal{E}_t(m) = \{t_1, t_2, \ldots, t_n\}$. From the basic properties of the exponential distribution [12], marking m has an expected sojourn time of $s(m) = 1/\sum_{t \in \mathcal{E}_t(m)} r_t > 0$, and therefore, it is called a *tangible* marking. Furthermore, the transitions in $\mathcal{E}_t(m)$ define an exponential race in m, and the probability that transition t_i, $i = 1, \ldots, n$, will fire first is equal to $p_{t_i}(m) = r_{t_i}/\sum_{t \in \mathcal{E}_t(m)} r_t$. On the other hand, in a marking $m \in \mathcal{R}(\mathcal{N})$ for which $\mathcal{E}_u(m) \neq \emptyset$, any transition $t \in \mathcal{E}_u(m)$ is expected to fire before a transition $t \in \mathcal{E}_t(m)$. In particular, if $\mathcal{E}_u(m)$ is a singleton, then, the single transition t in it will fire at m with certainty. If $\mathcal{E}_u(m)$ contains more than one transitions, then, in the standard GSPN theory, the selection of one of these transitions is regulated by an externally specified probability distribution defined on $\mathcal{E}_u(m)$ that is called the *random switch* associated with m. Furthermore, it should be clear from the above discussion that a marking $m \in \mathcal{R}(\mathcal{N})$ with $\mathcal{E}_u(m) \neq \emptyset$ has zero sojourn time, and therefore, such a marking is characterized as *vanishing* in the relevant literature. In the sequel, we shall denote the sets of tangible and vanishing markings by $\mathcal{R}_t(\mathcal{N})$ and $\mathcal{R}_v(\mathcal{N})$, respectively. Clearly, $\mathcal{R}_t(\mathcal{N})$ and $\mathcal{R}_v(\mathcal{N})$ define a partition of $\mathcal{R}(\mathcal{N})$. Finally, the random switch associated with a marking $m \in \mathcal{R}_v(\mathcal{N})$, with $\mathcal{E}_u(m) = \{t_1, t_2, \ldots, t_n\}$, will be denoted by $\Xi(m) = [\xi_1, \xi_2, \ldots, \xi_n]^T$.

The firing dynamics of the GSPN model that were described in the previous paragraph define a stochastic process on $\mathcal{R}(\mathcal{N})$ that has the particular structure of a semi-Markov process [1]; in the sequel, we shall denote this semi-Markov process by $\mathcal{SM}(\mathcal{N})$. Sojourn times and branching probabilities for states (or, equivalently, markings) $m \in \mathcal{R}_t(\mathcal{N})$ are determined by the corresponding exponential races described in the previous paragraph and the net flow relation W_t, i.e., the restriction of W on the timed transitions of the net. On the other hand, sojourn times for markings $m \in \mathcal{R}_v(\mathcal{N})$ are equal to zero and the corresponding branching probabilities are determined by the random switches $\Xi(m)$ assigned to those markings, together with the net flow relation W_u. When net $\mathcal{N}$ is also reversible, under the control of the selected random switches $\Xi(m)$, $m \in \mathcal{R}_v(\mathcal{N})$, process $\mathcal{SM}(\mathcal{N})$ can be shown to be ergodic [12], and thus, analyzable for its long-term (or *"steady-state"*) behavior through standard techniques borrowed from the theory of ergodic stochastic processes. For ergodic GSPNs it is worth-noticing that, since markings $m \in \mathcal{R}_v(\mathcal{N})$ have zero sojourn time in $\mathcal{SM}(\mathcal{N})$, they will also have zero steady-state probabilities. More generally, the timed dynamics of any given GSPN $\mathcal{N}$ can be effectively analyzed by the sub-process $\mathcal{M}(\mathcal{N})$ that projects the dynamics of $\mathcal{SM}(\mathcal{N})$ on the subspace of tangible markings. $\mathcal{M}(\mathcal{N})$ is a continuous-time Markov chain (CTMC) and its infinitesimal generator $Q(\mathcal{N})$ can be computed from the structural characterization of $\mathcal{SM}(\mathcal{N})$ [1]. For an ergodic $\mathcal{SM}(\mathcal{N})$, the availability of $Q(\mathcal{N})$ enables also the characterization of the steady-state probability distribution $\pi \equiv [\pi(m),\ m \in \mathcal{R}_t(\mathcal{N})]$ of $\mathcal{M}(\mathcal{N})$, and the long-term behavior of the underlying system.

III. The considered optimization problem

The original optimization problem The above discussion established that, for ergodic GSPNs, the analysis of the induced processes $\mathcal{SM}(\mathcal{N})$ and $\mathcal{M}(\mathcal{N})$ can provide a succinct characterization of the steady-state dynamics. Additional long-term performance considerations can be investigated in this limiting regime by introducing an *"(immediate) reward"* or *"cost"* function on the net markings and/or its transitions, and considering the expectation of these functions w.r.t. the steady-state probability distribution π. In this way, one can characterize and compute "throughput" and "utilization rates" w.r.t. various transitions of the net as well as "holding costs" associated with the marking of its various places.

This work intends to use the GSPN modeling framework in a more "prescriptive" manner, namely for the framing and the investigation of the following fundamental problem: *Given a (RAS-modeling) GSPN net defined by (i) the net topology and (ii) the firing rates of its timed transitions, and (iii) an immediate reward function defined on the net markings and/or transitions that characterizes a performance measure of interest, find a set of random switches for the net vanishing markings that maximizes the long-term performance of the net w.r.t. the aforementioned performance measure.*

As explained in the introductory section, we perceive the random switches associated with the various vanishing markings as "decisions" that must be effected upon the net dynamics through the selection of an enabled untimed transition from the corresponding random switch. On the other hand, timed transitions model the execution of the various processes / operations that are activated by these decisions. Hence, the net dynamics evolve through an alternation of tangible markings that correspond to the concurrent execution of a number of processes in the net, and a set of vanishing markings that constitute decision points in response to the completion of some of the activated processes in the net. In fact, such a process completion can activate a cascading sequence of decisions, that are modeled by a corresponding sequence of vanishing markings, before the net settles to another processing phase modeled by the next tangible marking.

The GSPN performance optimization problem described in the previous paragraphs, when combined with the Markovian nature of the stochastic processes that model the timed behavior of the GSPN nets, can be casted in the framework of MDP theory [13]. In particular, the specification of a set of random switches that will regulate the transition firing in the vanishing markings of the considered GSPNs essentially defines a stationary policy that shapes the net dynamics by defining the stochastic processes $\mathcal{SM}(\mathcal{N})$ and $\mathcal{M}(\mathcal{N})$ that were discussed in the previous section. In the sequel, we shall denote by ϕ the stationary policy that results by any particular selection of random switches, and we shall also use Φ to denote the space of all the considered stationary policies. We further stipulate that the considered policies are able to ensure ergodic behavior of the underlying net. From a practical standpoint, this requirement can be satisfied by (i) superimposing on the original net an appropriate supervisory control policy (SCP) [12] that will enforce the reversibility of the net behavior,[2] and (ii) lower-bounding the probability values of the random switches that will coordinate the restricted net behavior by a small value $\delta > 0$. Then, letting $f \equiv [f(m),\ m \in \mathcal{R}_t(\mathcal{N})]$ denote a reward rate function defined on the markings $m \in \mathcal{R}_t(\mathcal{N})$, $\pi(\phi)$ denote the steady-state probability distribution for the CTMC $\mathcal{M}(\mathcal{N})$ that results from the application of policy ϕ, and $\eta(\phi)$ denote the resulting (long-term) average reward, our optimization problem can be expressed by the following mathematical programming (MP) formulation:

$$\max_{\phi \equiv \{\Xi(m), m \in \mathcal{R}_v(\mathcal{N})\}} \eta(\phi) = \pi^T(\phi) \cdot f \tag{1}$$

s.t.

$$\pi^T(\phi) \cdot [Q(\phi)\ \mathbf{1}] = [\mathbf{0}^T 1] \tag{2}$$

$$\forall m \in \mathcal{R}_v(\mathcal{N}), \quad \Xi^T(m) \cdot \mathbf{1} = 1.0 \tag{3}$$

$$\forall m \in \mathcal{R}_v(\mathcal{N}),\ \forall t \in \mathcal{E}_u(m), \quad \delta \leq \xi(t) \tag{4}$$

Equation 1 in the above formulation expresses the objective of maximizing the long-term average reward that is collected according to the reward function f. Constraints 3 and 4 define the structure of the random switches, that are the primary decision variables in the considered optimization problem. On the other hand, Constraint 2 defines the steady-state probability distribution $\pi(\phi)$ for the CTMC that is induced by any pricing of the random switches $\Xi(m),\ m \in \mathcal{R}_v(\mathcal{N})$, according to Constraints 3 and 4; the reader should notice that in the considered formulation, the variables that represent the probability distribution $\pi(\phi)$ have an auxiliary role.

Complexity considerations and the revised optimization problem From a computational standpoint, the formulation of Equations 1–4 is challenged by the fact that its size, in terms of the employed decision variables and constraints, is commensurate to the size of $\mathcal{R}_v(\mathcal{N})$, which grows exponentially w.r.t. the size of underlying the GSPN model. This limitation is essentially the same with the computational challenges that limit the practical solution of the considered optimization problem through the classical MDP theory [14]. In particular, the employment of a separate random switch for every marking in $\mathcal{R}_v(\mathcal{N})$ implies that even the mere enumeration of a given policy ϕ from the considered class is a task of non-polynomial complexity w.r.t. the size of underlying RAS. We shall characterize this fact by saying that the aforementioned policies ϕ possess *exponential space complexity.*

To deal with this increased and frequently computationally prohibitive complexity, in this work we shall restrict attention to a class of policies that require a reduced space complexity for their characterization. The considered policies are suggested naturally by the structure of the underlying GSPN model, and in many cases, they also define policy classes of practical significance and value for the original optimization problem under consideration. From a mathematical modeling standpoint, these policies will be defined through the introduction of additional constraints in the MP formulation of Equations 1–4 that will establish some "coupling" among the variables of the random switches $\Xi(m)$ that appear in the formulation of Equations 1–4, and will enable the elimination of some of these variables. A simple way to implement this idea is by introducing the following additional constraint to the optimization problem of

[2] For notational economy, in the sequel we shall use $\mathcal{N}$ to refer to the controlled net, as well.

Equations 1–4:

$$\forall m, m' \in \mathcal{R}_v(\mathcal{N}) \text{ with } \mathcal{E}_u(m) = \mathcal{E}_u(m'), \quad \Xi(m) = \Xi(m') \tag{5}$$

Under Equation 5, the random switches $\Xi(m)$ employed by the resultant formulation are defined only by the set $\mathcal{E}_u(m)$ of the enabled untimed transitions that appear in their support, and not by the marking m itself. Such random switches are characterized as *static* in the GSPN literature. Also, we shall refer to the optimization problem that is defined by Equations 1–5 as the *"revised optimization problem"*. The reader should notice that the set of static random switches is of cardinality $O(2^{|T_u|})$, i.e., in principle, the problem space complexity remains an exponential function of the size of the underlying GSPN $\mathcal{N}$. However, it is generally true that $2^{|T_u|} << |\mathcal{R}_v(\mathcal{N})|$. Furthermore, in most practical cases, the subsets of T_u that define support sets for some random switch $\Xi(m)$ of $\mathcal{N}$ will be significantly less than $2^{|T_u|}$, since these subsets are further constrained by the topology of $\mathcal{N}$ and the dynamics that are induced by this topology. Hence, it is expected that the revised optimization problem will be much more manageable in terms of its space and time complexity than the original optimization problem of Equations 1–4.

We should also point out that, in spite of the simplicity of the logic that underlies Equation 5, the resultant class of policies is pretty rich and of practical relevance in many practical applications. More specifically, the policy space that is defined by the set of static random switches enables the modeling of static priority policies defined on the basis of various "class" concepts. Such policies have been attractive in many application contexts due to their operational simplicity, and in various cases they have been proven to be optimal. Furthermore, it is easy to see that in the revised problem formulation, Constraint 5 essentially defines an aggregation on the underlying subspace $\mathcal{R}_v(\mathcal{N})$ by imposing the requirement that decisions in vanishing markings m with the same set of enabled untimed transitions $\mathcal{E}_v(m)$ should be governed by the same decision rule (i.e., by the same random switch $\Xi(m)$). One can envision the performance enhancement of the resulting optimal policy through a refining process that partitions (some of) the aggregated state sets into smaller subsets. When this refinement is taken to its extreme, one retrieves the original formulation of Equations 1–4. In the general case, such a refinement can be effected through trial-and-error-based procedures and search-based mechanisms similar to those used in the area of combinatorial optimization [15]. In certain cases, it may also be based on special problem characteristics that are derived from the structure of the underlying system. The concise, explicit representation of the structure of the underlying system by the employed GSPN model enables a succinct articulation of the applied refinement logic, and it facilitates the identification of behavioral traits and attributes that may be exploitable by the refinement process.

Uniformization and a further reduction of the revised optimization problem In the next section we shall present a solution of the revised optimization problem based on some results for sensitivity analysis of Markov reward processes. To apply these results it is convenient to discretize the Markovian dynamics that underlie the revised problem formulation, by uniformizing these dynamics with an appropriate sampling rate r_u, e.g., $r_u = \sum_{t \in T_t} r_t$. In the sequel we shall denote the resulting discrete-time Markov chain (DTMC) by $\mathcal{U}(\mathcal{N})$ and the corresponding one-step-transition probability matrix by $\hat{P}(\mathcal{N})$. Also, let us denote by $\bar{\xi}$ the set of variables that remain in the final formulation of the considered problem, after removing all the variables that are rendered superfluous by the constraint of Equation 5 (or, more generally, by the additional constraints that define the target policies). Then, recognizing also the auxiliary role of variables $\pi(\phi)$, the final problem formulation can be formally reduced to

$$\max_{\bar{\xi}} \ \eta(\bar{\xi}) \tag{6}$$

s.t.

$$\forall \xi \in \bar{\xi}, \quad \delta \leq \xi \tag{7}$$

$$\forall \Xi, \quad \delta \leq 1.0 - \sum_{\xi \in \Xi \cap \bar{\xi}} \xi \tag{8}$$

In Equation 8, Ξ denotes those random switches that are recognized as distinct entities by the definition of the underlying policy space. We also notice that in the sequel we shall use interchangeably the vector $\bar{\xi}$ and the policy ϕ that is induced by this vector.

IV. Solving the considered optimization problem

The performance sensitivity formula The optimization problem of Equations 6–8 can be addressed by rather standard techniques from the theory of Mathematical Programming (MP) [16], provided that the partial derivatives $\partial\eta(\bar{\xi})/\partial\xi$ are well defined over the feasibility space defined by Constraints 7 and 8. In [7] it is shown that for discrete-time Markov reward processes of the type that underlie the MP formulation of Equations 6–8, the derivative of the performance index η w.r.t. any parameter ξ that is involved in the definition of the corresponding one-step-transition probability matrix $\hat{P}$, can be computed as follows:

$$\frac{\partial \eta(\bar{\xi})}{\partial \xi} = \pi(\bar{\xi})^T \cdot \frac{\partial}{\partial \xi}\hat{P}(\bar{\xi}) \cdot g(\bar{\xi}) \tag{9}$$

The vector $g(\bar{\xi})$ that appears in the above equation is the *relative value function* (or the *potential function*) for the uniformized Markov reward process $\mathcal{U}(\mathcal{N}, f; \phi)$. In the considered class of Markov reward processes, vector $g(\bar{\xi})$ is defined by the corresponding Bellman equation up to an additive constant [13]. However, since $\hat{P}(\bar{\xi})$ is a stochastic matrix, the matrix $\frac{\partial}{\partial \xi}\hat{P}(\bar{\xi})$ has zero row sums, and therefore, the computation of $\frac{\partial \eta(\bar{\xi})}{\partial \xi}$ through Equation 9 is invariant to the various selections of $g(\bar{\xi})$. Furthermore, in the sequel we shall show that, in the considered GSPN models, the elements of the matrix

$\hat{P}(\bar{\xi})$ are polynomials of the decision variables ξ. Therefore, the (matrix) partial derivative $\frac{\partial}{\partial \xi}\hat{P}(\bar{\xi})$, that appears in Equation 9, will always exist. Hence, as long as the variable vector $\bar{\xi}$ is selected in the space defined by Equations 7 and 8, and the resultant matrix $\hat{P}(\xi)$ corresponds indeed to the one-step-transition probability matrix of an irreducible, ergodic Markov chain, the derivative $\frac{\partial \eta(\bar{\xi})}{\partial \xi}$ will be well defined.

Sample-path-based estimation of the performance derivatives From a computational standpoint, the practical value of Equation 9 is limited by the fact that each of the three factors that appear in its right-hand-side is an entity of size commensurate to the size of the underlying state space; as explained in Section III, the state space for the considered processes will explode even for rather small GSPN configurations. But the results of [7] provide also sample-path-based estimators for $\frac{\partial \eta(\bar{\xi})}{\partial \xi}$ that are derived from Equation 9 and the finite and irreducible nature of the underlying Markov process. Such an estimator can be obtained, for instance, from the following result [7]:

$$\frac{\partial \eta(\bar{\xi})}{\partial \xi} = \frac{E\left[\sum_{k=u_\nu}^{u_{\nu+1}-1}[f(m_k)-\eta(\bar{\xi})]\sum_{j=u_{\nu(k)}}^{k}\frac{\frac{\partial}{\partial \xi}\hat{p}(m_{j-1},m_j;\bar{\xi})}{\hat{p}(m_{j-1},m_j;\bar{\xi})}\right]}{E[u_{\nu+1}-u_\nu]} \quad (10)$$

In Equation 10, $\mathbf{m} = \langle m_0, m_1, m_2, \ldots\rangle$ denotes a sample path of the considered process, and the sequence u_ν, $\nu = 0,1,2,\ldots$, collects the time points of the path visits to some selected state m^* that constitutes a regenerative point for the underlying stochastic process. Finally, $u_{\nu(k)}$ denotes the time step of the last visit, prior to time step k, to the aforementioned regenerative point m^*.

Equation 10 implies that the detailed specification of a $\frac{\partial \eta(\bar{\xi})}{\partial \xi}$ estimator on the basis of this equation necessitates the characterization of the elements of matrix $\hat{P}(\bar{\xi})$ as functions of $\bar{\xi}$, and of the partial derivatives of these elements w.r.t. each decision variable ξ. Next we provide these characterizations.

Computing the elements of matrix $\hat{P}(\bar{\xi})$ and their partial derivatives As explained in Section III, the interpretation of the random switches Ξ as "decisions" that drive the system behavior further implies that, in the underlying semi-Markov process $\mathcal{SM}(\mathcal{N})$, the transition from a tangible state m to another tangible state m' will be interfered, in general, by a sequence of vanishing markings $\hat{M} = \{\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_k\}$. Sequence $\hat{M}$ will contain no cyclical behavior; i.e., $\hat{m}_i \neq \hat{m}_j$ for $i \neq j$, since the repetition of vanishing markings in sequence $\hat{M}$ can be perceived as a manifestation of "confusion" in the underlying decision making process. However, it is conceivable that two or more of the vanishing markings in $\hat{M}$ might possess the same static random switch Ξ; i.e., it is possible that $\mathcal{E}_v(\hat{m}_i) = \mathcal{E}_v(\hat{m}_j)$ for some $i \neq j$, in which case, Equation 5 further implies that $\Xi(\hat{m}_i) = \Xi(\hat{m}_j)$. Next, consider the transition sequence $\hat{T} = \{\hat{t}_0, \hat{t}_1, \ldots, \hat{t}_k\}$ where $m \xrightarrow{\hat{t}_0} \hat{m}_1$, $\hat{m}_i \xrightarrow{\hat{t}_i} \hat{m}_{i+1}$ for $i = 1, \ldots, k-1$, and $\hat{m}_k \xrightarrow{\hat{t}_k} m'$. $\hat{T}$ can be considered as another representation of the dynamics that lead process $\mathcal{SM}(\mathcal{N})$ from m to m' through the sequence of vanishing markings $\hat{M}$, and it facilitates a straightforward computation of the realization probability of the corresponding dynamics. More specifically, let $\hat{p}(\hat{t}_i;\bar{\xi})$, $i = 0,1,\ldots,k$, denote the probability for firing transition $\hat{t}_i$ in the corresponding marking (i.e., marking m for $i = 0$, or marking $\hat{m}_i$ for $i = 1, \ldots, k$). Then, taking also into consideration (i) the transitional dynamics that are introduced by the uniformizing operation that was introduced in Section III, and (ii) the specification of the variable vector $\bar{\xi}$ employed by the formulation of Equations 6–8, we have that

$$\hat{p}(\hat{t}_i;\bar{\xi}) = \begin{cases} \frac{r_{\hat{t}_i}}{r_u}, \text{for } i = 0 \\ \xi(\hat{t}_i) \in \Xi(\hat{m}_i), \text{for } i = 1, \ldots, k \wedge \xi(\hat{t}_i) \in \bar{\xi} \\ 1.0 - \sum_{\xi \in \Xi(\hat{m}_i) \cap \bar{\xi}} \xi, \text{for } i = 1, \ldots, k \wedge \xi(\hat{t}_i) \notin \bar{\xi} \end{cases} \quad (11)$$

Also, the probability for the entire transition sequence $\hat{T}$ is given by

$$\hat{p}(\hat{T};\bar{\xi}) = \prod_{i=0}^{k} \hat{p}(\hat{t}_i;\bar{\xi}) \quad (12)$$

From Equations 11 and 12 it is easy to see that $\hat{p}(\hat{T};\bar{\xi})$ constitutes a polynomial function in $\bar{\xi}$. Generally, there will be more than one sequence $\hat{M}$ materializing the transition from m to m'. Let all these sequences be denoted by $\hat{M}^1, \ldots, \hat{M}^l$, and furthermore, let $\hat{T}^1, \ldots, \hat{T}^l$ denote the corresponding transition sequences defined according to the above discussion. The total probability of transitioning from a tangible marking m to another marking m' in the uniformized process $\mathcal{U}(\mathcal{N};\bar{\xi})$ is given by

$$\hat{p}(m,m';\bar{\xi}) = \sum_{j=1}^{l} \hat{p}(\hat{T}^j;\bar{\xi}) = \sum_{j=1}^{l}\prod_{i=0}^{k(j)} \hat{p}(\hat{t}_i^j;\bar{\xi}) \quad (13)$$

and $\hat{p}(m,m';\bar{\xi})$ remains a polynomial function in $\bar{\xi}$.

Equation 13 suggests a straightforward algorithm for computing $\hat{p}(m,m';\bar{\xi})$. For any given transition pair (m,m') in $\mathcal{U}(\mathcal{N};\bar{\xi})$, we first construct the *acyclic* digraph $\mathcal{G}_v(\mathcal{N};m)$ which unfolds the transitional dynamics of GSPN $\mathcal{N}$ when the latter is initialized at marking m and evolves first through the execution of the transitions that are enabled in m and subsequently through the execution of transition sequences that consist of untimed transitions only. Hence, graph $\mathcal{G}_v(\mathcal{N};m)$ contains (i) marking m as its "source" node, (ii) the set of markings $\tilde{M}$ that can be reached from m through the firing of any transition $t \in \mathcal{E}(m)$, and (iii) all the markings that can be reached from the markings in $\tilde{M}$ by firing untimed transitions only. Furthermore, the "leaf" nodes of $\mathcal{G}_v(\mathcal{N};m)$ will be a set $\bar{M}$ of tangible markings with $m' \in \bar{M}$.[3] The availability of $\mathcal{G}_v(\mathcal{N};m)$ subsequently enables

[3] The reader should also notice that the "unfolding" nature of the construction of $\mathcal{G}_v(\mathcal{N};m)$ implies that it is possible that $m \in \bar{M}$. In fact, this will be the case for the "fictitious" transitions in $\mathcal{U}(\mathcal{N})$ that are introduced from the uniformization.

the enumeration of all the transition sequences T^j that connect markings m and m' according to the previous discussion, and the computation of $\hat{p}(m,m';\bar{\xi})$ according to the formula of Equation 13.

The deployment of the digraph $\mathcal{G}_v(\mathcal{N};m)$, for any tangible marking $m \in \mathcal{R}_t(\mathcal{N})$, is a "local" computation when perceived in the context of the entire reachability space $\mathcal{R}(\mathcal{N})$, and, in general, it is expected that each such digraph $\mathcal{G}_v(\mathcal{N};m)$ will be a pretty small graph. This is also suggested from the conceptual interpretation of $\mathcal{G}_v(\mathcal{N};m)$ as the digraph that encodes the decision sequences that can be effected in response to the events (i.e., the transition firings) that take place in marking m. Hence, the computation of the digraphs $\mathcal{G}_v(\mathcal{N};m)$, and the extraction of the necessary transition sequences T^j from them, are supposed to be performed on the fly, upon the observation of the corresponding transition (m,m') in the processed sample path $\mathbf{m}$.

Furthermore, the availability of the transition sequences T^j for any pair $m,m' \in \mathcal{R}_t(\mathcal{N})$ enables also the computation of $\frac{\partial}{\partial \xi}\hat{p}(m,m';\bar{\xi})$ for any $\xi \in \bar{\xi}$. To define an algorithm for this computation, we first notice that it suffices to provide an algorithm for the computation of $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$, for any sequence T^j; once this algorithm is available, Equation 13 implies that $\frac{\partial}{\partial \xi}\hat{p}(m,m';\bar{\xi})$ can be computed by applying it for every T^j and summing the obtained results. But an algorithm for the computation of $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$ can be obtained from Equation 12, which implies that

$$\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi}) = \sum_{q=1}^{k(j)}\{[I_{\{\xi(\hat{t}^j_q)=\xi\}} - I_{\{(\xi(\hat{t}^j_q)\notin\bar{\xi})\wedge(\xi\in\Xi(\hat{m}_q)\cap\bar{\xi})\}}]\prod_{i=0;i\neq q}^{k(j)}\hat{p}(\hat{t}^j_i;\bar{\xi})\} \tag{14}$$

In Equation 14, the quantity $I_{\{\cdot\}}$ denotes the indicator variable for the condition that is expressed in the brackets. Hence, in more practical terms, $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$ can be computed as follows: First, $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$ is initialized to zero. Subsequently, the subsequence $\{\hat{t}^j_q : q = 1,\ldots,k(j)\}$ is parsed from left to right, and every time that a transition $\hat{t}^j_q$ is encountered such that the probability $\xi(\hat{t}^j_q)$ is expressed by variable ξ, then the corresponding product $\prod_{i=0;i\neq q}^{k(j)}\hat{p}(\hat{t}^j_i;\bar{\xi})$ is added to the partial sum expressing $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$. On the other hand, if the encountered transition $\hat{t}^j_q$ has a corresponding probability $\xi(\hat{t}^j_q) \notin \bar{\xi}$ but the corresponding random switch $\Xi(\hat{m}_q)$ contains ξ, then the product $\prod_{i=0;i\neq q}^{k(j)}\hat{p}(\hat{t}^j_i;\bar{\xi})$ is subtracted from the partial sum expressing $\frac{\partial}{\partial \xi}\hat{p}(\hat{T}^j;\bar{\xi})$. In all other cases, the computed sum remains unchanged.

Solving the MP formulation of Equations 6–8 through stochastic approximation algorithms With the performance gradient $\nabla\eta(\bar{\xi})$ – or, more specifically, a good estimator of this gradient – effectively computable, the optimization problem of Equations 6–8 can be addressed through algorithms provided by the theory of stochastic approximation [9], [8]. In [10] we adapt a rather classical stochastic approximation algorithm presented in [9] to the considered problem, and we establish its ability to converge to stationary points of the underlying objective function. Furthermore, the overall dynamics of the considered algorithm, and the corresponding stability properties, render unlikely its entrapment in stationary points of the wrong type. The developments presented in [10] provide also a case study that demonstrates the efficacy and the performance of the proposed optimization algorithm, and corroborates the aforestated expectations.

V. Conclusions

This paper introduced a novel methodology for performance optimization of many contemporary operations, based on their GSPN model and some recent results on the sensitivity analysis of Markov reward processes. Future work will seek a systematic assessment of the quality of the obtained policies, and the further refinement of the method for enhanced modeling power and performance, along the lines that were outlined in the earlier parts of this document.

References

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

[2] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modeling with Generalized Stochastic Petri Nets*. John Wiley & Sons, 1994.

[3] M. Beccuti, G. Franceschinis, and S. Haddad, "Markov Decision Petri Net and Markov Decision Well-Formed Net formalisms," in *ICATPN 2007 – LNCS 4546*, 2007, pp. 43–62.

[4] S. A. Reveliotis, *Real-time Management of Resource Allocation Systems: A Discrete Event Systems Approach*. NY, NY: Springer, 2005.

[5] S. Reveliotis and A. Nazeem, "Deadlock avoidance policies for automated manufacturing systems using finite state automata," in *Formal Methods in Manufacturing*, J. Campos, C. Seatzu, and X. Xie, Eds. CRC Press / Taylor & Francis (to appear).

[6] J. Y. Choi and S. A. Reveliotis, "Relative value function approximation for the capacitated re-entrant line scheduling problem," *IEEE Trans. on Automation Science and Engineering*, vol. 2, pp. 285–299, 2005.

[7] X. Cao, *Stochastic Learning and Optimization*. NY,NY: Springer, 2007.

[8] P. Marbach and J. N. Tsitsiklis, "Simulation-based optimization of Markov reward processes," *IEEE Trans. on Automatic Control*, vol. 46, pp. 191–209, 2001.

[9] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. NY, NY: Springer, 2003.

[10] R. Li and S. Reveliotis, "Performance optimization for a class of Generalized Stochastic Petri Nets," ISyE, Georgia Tech, Tech. Rep. (submitted for publication), 2013.

[11] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, pp. 541–580, 1989.

[12] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems (2nd ed.)*. NY,NY: Springer, 2008.

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

[14] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control," *Math. of OR*, vol. 24, pp. 293–305, 1999.

[15] A. Gosavi, *Simulation-based Optimization: Parametric Optimization Techniques and Reinforcement Learning*. Norwell, MA: Kluwer Academic Publishers, 2003.

[16] D. G. Luenberger, *Linear and Nonlinear Programming (2nd ed.)*. Reading, MA: Addison Wesley, 1984.