Deadlock Avoidance in Sequential Resource Allocation Systems with Multiple Resource Acquisitions and Flexible Routings

Jonghun Park and Spyros A. Reveliotis*
School of Industrial & Systems Engineering
Georgia Institute of Technology

Abstract

This paper considers the deadlock avoidance problem for the class of conjunctive / disjunctive (sequential) resource allocation systems (C/D-RAS), which allows for multiple resource acquisitions and flexible routings. First, a new siphon-based characterization for the liveness of Petri nets (PN's) modeling C/D-RAS is developed, and subsequently, this characterization facilitates the development of a polynomial-complexity deadlock avoidance policy (DAP) that is appropriate for the considered RAS class. The resulting policy is characterized as C/D-RUN, since the starting point for the policy development was motivated by the RUN DAP, originally developed for sequential RAS with unit resource allocations and no routing flexibility. The last part of the paper exploits the aforementioned siphon-based characterization of C/D-RAS liveness, in order to develop a sufficiency condition for C/D-RAS liveness that takes the convenient form of a Mixed Integer Programming (MIP) formulation. The availability of this MIP formulation subsequently allows the "automatic" correctness verification of any tentative C/D-RAS DAP for which the controlled system behavior remains in the class of PN's modeling C/D-RAS, and the effective flexibility enhancement of the aforementioned C/D-RUN DAP implementations. Finally, we notice that, in addition to extending and complementing the current theory on deadlock-free sequential resource allocation to the most powerful class of C/D-RAS, the presented results also (i) non-trivially generalize important concepts and techniques of ordinary PN structural analysis to the broader class of non-ordinary PN's, while (ii) from a practical standpoint, they can find direct application in the (work-)flow management of modern production, service and/or transportation environments.

Keywords: Deadlock Avoidance, Liveness, Resource Allocation Systems, Petri Net Structural Analysis

*Corresponding author: phone #: (404) 894-6608, Fax #: (404) 894-2301, e-mail: *spyros@isye.gatech.edu*, mail address: 765 Ferst Dr., Atlanta, GA 30332

1 Introduction

Deadlock avoidance in (sequential) resource allocation systems (S-RAS) is a well-defined problem in Discrete Event System literature. From a theoretical standpoint, the problem arises in any resource sharing system where a set of concurrently executing sequential processes can get permanently blocked – i.e., *deadlocked* – due to the fact that each process in that set is allocated and holds resource(s) requested by some other process in the set for its further advancement. From a practical standpoint, the problem is experienced in the operational control of many contemporary technological systems, including the material flow control of flexibly automated production systems, the traffic management of unmanned discrete material handling systems like automated and/or rail guided vehicle systems, and the traffic control of railway and urban monorail transport systems. In all of these systems, the resolution of a developed deadlock can imply a major disruption of the (normal) system operation, while during its occurrence, the utilization of the involved resources is driven to zero. It is, therefore, desirable that the controller supervising the real-time operation of these systems incorporates a control function that foresees and effectively prevents the occurrence of the problematic deadlock states, by appropriately restricting the allocation of the system resources to the various requesting processes.

Ideally, due to performance considerations, the aforementioned deadlock avoidance function should be carried out in the *least* restrictive way. However, it has been formally established [1] that, in the general case, implementation of the *optimal* – i.e., least restrictive – deadlock avoidance policy (DAP) is an *NP-Hard* [9] problem for the considered class of S-RAS. Hence, in the light of this result, one should aim at the development of suboptimal, but computationally efficient – i.e., polynomial complexity – deadlock avoidance policies, which still maintain a significant level of operational flexibility.[1] It has also been found that the complexity of the synthesis of effective and efficient deadlock avoidance policies for the considered class of resource allocation systems strongly depends on the underlying system structure and the particular assumptions regarding the admissible resource allocation requests. A taxonomy that attempts to manage the problem complexity by classifying S-RAS on the basis of the resource allocation request structure is presented in [17]. According to that taxonomy, S-RAS are classified in four major classes: (i) *Single-Unit (SU)* RAS, in which every process stage requires only a *single* unit of a *single* resource for its successful execution, (ii) *Single-Type (ST)* RAS, in which every process stage requires an *arbitrary* number of units of a *single* resource for its successful execution, (iii) *Conjunctive (C)* RAS, in which every process stage requires an *arbitrary* number of units from an *arbitrary* resource (sub-)set for its successful execution, and (iv)

[1] Another interesting research line in the area of Deadlock Avoidance for S-RAS has sought to identify special system structure that admits polynomial complexity optimal deadlock avoidance; c.f. [11] for an overview of these results.

Conjunctive / Disjunctive (C/D) RAS, in which every process stage poses a finite number of *alternative* conjunctive-type resource requests. Notice that the class of C/D-RAS incorporates the most general resource allocation schemes, allowing for, both, multiple resource acquisitions and flexible routings. Presently, systematic study of the class of C/D-RAS and its underlying behavioral dynamics is not only theoretically interesting, but also practically important, since emerging technological applications are expected to support high degree of resource sharing and flexibility, which, in turn, leads to more efficient utilization of the system resources.[2]

Past research has addressed successfully the aforestated deadlock avoidance problem primarily in the context of single-unit resource allocation (SU-RAS). Some indicative results can be found in [2, 16, 22, 8, 12]. Of particular interest to the work presented in this paper are the developments presented in [7, 5, 14], which also leverage recently obtained results in the area of Petri Net (PN) structural analysis. Specifically, the results presented in [7, 5] have established that for the case of single-unit resource allocation, the occurrence of deadlock can be structurally explained in the PN formalism through the concept of *empty siphon*, while the work in [14] develops an analytical framework that exploits this deadlock characterization in order to effectively generalize / enrich the class of effectively computable DAP's that are appropriate for SU-RAS. Recently, the work of [20] has also shown that the empty siphon can be the deadlock interpreting mechanism even in the case of C/D-RAS, in which, however, resources are acquired one unit at a time. Attempts to generalize the notion of empty siphon towards the interpretation of deadlock occurring in the general C/D-RAS have been presented in [3, 19], but the siphon constructs proposed in those works fail to establish a complete resolution of this problem, in the form of a siphon-based necessary and sufficient condition for C/D-RAS liveness, while the developed solutions to the underlying deadlock avoidance problem present very high computational complexity.

In the context of the aforementioned research developments, and motivated by the above remarks, the work presented in this paper makes the following key contributions:

i. First, it provides a complete PN-based structural characterization of C/D-RAS liveness through a new *siphon* construct, that effectively extends the notion of empty siphon to non-ordinary Petri nets.

ii. Subsequently, it employs the C/D-RAS liveness characterization derived in Step (i), towards the DAP development for the considered RAS class. A key advantage and distinguishing characteristic of this policy with respect to (w.r.t.) similar attempts existing in the literature, is that it presents *polynomial-time complexity* for its initial development

[2]For an extensive discussion on the applicability of the theoretical results presented in this paper to the management of the material and auxiliary tool (reticle) flow in the context of the emerging, flexibly automated 300mm fab, the reader is referred to [15].

and run-time execution. Furthermore, since the starting point for the policy development has been the RUN DAP, which was originally developed in [16] for SU-RAS, the resulting policy will be called C/D-RUN.

iii. Finally, the last part of the paper exploits the derived siphon-based liveness characterization of C/D-RAS, in order to develop a *sufficiency* test for the correctness of any tentative C/D-RAS DAP that can be expressed as a set of invariant-imposing "control" places, superimposed to the PN modeling the original (uncontrolled) RAS behavior.[3] This test takes the convenient form of a *Mixed Integer Programming (MIP)* [21] formulation, and therefore, it can support *"automatic" DAP correctness verification* in the considered RAS context. From a practical standpoint, this part of the work allows the potential enrichment of the space of effectively computable and computationally efficient DAP's for any given C/D-RAS configuration with additional policies, beyond the class generated by the C/D-RUN defining logic. Furthermore, it is shown that a pertinent combination of the C/D-RUN defining logic with the capabilities and computational effectiveness of this new DAP correctness verification tool, can lead to the systematic development of variations of the original C/D-RUN DAP, that are characterized by enhanced operational flexibility.

The rest of the paper is organized as follows: Section 2 revises the basic concepts of PN-based modeling and analysis, and it develops a PN-based characterization of the C/D-RAS structure and behavior, by defining a new PN subclass to be known as the class of S^3PGR^2 nets. Section 3 investigates the liveness properties of S^3PGR^2 nets, and establishes their strong relationship to the development of a new siphon construct, to be known as the *deadly marked siphon*. Section 4 exploits the findings of Section 3 towards the development of C/D-RUN, a polynomial-complexity DAP able to support deadlock and induced deadlock-free operation in the C/D-RAS context. Section 5 first develops a MIP-based characterization for the existence of deadly marked siphons in any given marking of a structurally bounded non-ordinary PN, and subsequently, it extends this criterion to a sufficiency condition for the liveness of PN's modeling C/D-RAS. Furthermore, the last part of this section discusses how the derived liveness test for C/D-RAS can support the flexibility enhancement of any C/D-RUN implementation, and the correctness verification of other tentative DAP's such that the PN modeling the controlled system behavior belongs to the class of S^3PGR^2 nets. Finally, Section 6 concludes the paper and suggests directions for future research.

[3] Such are the policies developed, for instance, in [7, 3, 14].

2 Petri Net-based Modeling of C/D-RAS

This section first revises the Petri net (PN) related concepts that are necessary for the formal modeling and analysis of the considered class of C/D-RAS, and subsequently, it provides a detailed characterization of the PN structure modeling the considered resource allocation environments. Some excellent more extensive treatments of the Petri net modeling framework and the structural and behavioral analysis of the resulting models can be found in [13, 6].

Petri net preliminaries A marked Petri Net is defined by a quadruple $\mathcal{N} = (P, T, W, M_0)$, where P is the set of *places*, T is the set of *transitions*, $W : (P \times T) \cup (T \times P) \to Z^+$ is the *flow relation*, and $M_0 : P \to Z^+$ is the net *initial marking*, assigning to each place $p \in P$, $M_0(p)$ *tokens*. In the special case that the flow relation W maps onto $\{0, 1\}$, the Petri net is said to be *ordinary*. The set of input (resp., output) transitions of a place p is denoted by ${}^\bullet p$ (resp., $p^\bullet$). Similarly, the set of input (resp., output) places of a transition t is denoted by ${}^\bullet t$ (resp., $t^\bullet$). This notation is also generalized to any set of places or transitions, X, e.g. ${}^\bullet X = \bigcup_{x \in X} {}^\bullet x$. The ordered set $X =< x_1 \dots x_n > \subseteq P \cup T$ is a *path*, if and only if (iff) $x_{i+1} \in x_i^\bullet, i = 1, \dots, n-1$. Furthermore, a path X is characterized as a *circuit* iff $x_1 \equiv x_n$. Finally, if $\forall t \in T$, $|t^\bullet| = |{}^\bullet t| = 1$, the PN is called a *state machine*.

Given a marking M, a transition t is *enabled* iff $\forall p \in {}^\bullet t$, $M(p) \geq W(p, t)$, and this is denoted by $M[t\rangle$. $t \in T$ is said to be *disabled* by $p \in {}^\bullet t$ at M iff $M(p) < W(p, t)$. Furthermore, a place $p \in P$ for which $\exists t \in p^\bullet$ s.t. $M(p) < W(p, t)$ is said to be a *disabling* place at M. Firing an enabled transition t results in a new marking M', which is obtained by removing $W(p, t)$ tokens from each place $p \in {}^\bullet t$, and placing $W(t, p')$ tokens in each place $p' \in t^\bullet$. The set of markings reachable from M_0 through any fireable sequence of transitions is denoted by $R(\mathcal{N}, M_0)$. A marked PN $\mathcal{N}$ with initial marking M_0 is said to be *bounded* iff all markings $M \in R(\mathcal{N}, M_0)$ are bounded, while $\mathcal{N}$ is said to be *structurally bounded* iff it is bounded for any initial marking M_0.

In case that a marked PN is *pure* (i.e., $\forall (x, y) \in (P \times T) \cup (T \times P)$, $W(x, y) > 0 \Rightarrow W(y, x) = 0$), the flow relation can be represented by the *flow matrix* $\Theta = \Theta^+ - \Theta^-$ where $\Theta^+[p, t] = W(t, p)$ and $\Theta^-[p, t] = W(p, t)$. A *p-semiflow* y is a $|P|$-dimensional vector satisfying $y^T\Theta = 0$ and $y \geq 0$, and a *t-semiflow* x is a $|T|$-dimensional vector satisfying $\Theta x = 0$ and $x \geq 0$. A p-semiflow y (t-semiflow x, resp.) is said to be *minimal* iff $\nexists$ a p-semiflow y' (t-semiflow x', resp.) such that $\|y'\| \subset \|y\|$ ($\|x'\| \subset \|x\|$, resp.), where $\|y\| = \{p \in P \mid y(p) > 0\}$ ($\|x\| = \{t \in T \mid x(t) > 0\}$, resp.).

Given a marked PN $\mathcal{N} = (P, T, W, M_0)$, a transition $t \in T$ is *live* iff $\forall M \in R(\mathcal{N}, M_0), \exists M' \in R(\mathcal{N}, M)$ s.t. $M'[t\rangle$, and $t \in T$ is *dead* at $M \in R(\mathcal{N}, M_0)$ iff $\nexists M' \in R(\mathcal{N}, M)$ s.t. $M'[t\rangle$. A marking $M \in R(\mathcal{N}, M_0)$ is a (total) *deadlock* iff $\forall t \in T$, t is dead. A marked PN $\mathcal{N}$ is *quasi-live*

iff $\forall t \in T, \exists M \in R(\mathcal{N}, M_0)$ s.t. $M[t\rangle$, it is *weakly live* iff $\forall M \in R(\mathcal{N}, M_0), \exists t \in T$ s.t. $M[t\rangle$, and it is *live* iff $\forall t \in T$, t is live. Of particular interest for the liveness analysis of marked PN is a structural element known as *siphon*, which is a set of places $S \subseteq P$ such that ${}^\bullet S \subseteq S^\bullet$. A siphon S is *minimal* iff $\nexists$ a siphon S' s.t. $S' \subset S$. A siphon S is said to be *empty* at marking M iff $M(S) \equiv \sum_{p \in S} M(p) = 0$.

C/D-RAS and their PN-based modeling For the purposes of this work, the Conjunctive/Disjunctive (C/D) RAS is formally defined by a set of *resource types* $\mathcal{R} = \{R_i, i = 1, \ldots, m\}$, and a set of *job types* $\mathcal{J} = \{J_j, j = 1, \ldots, n\}$. Every resource type R_i is further characterized by its *capacity* $C_i \in Z^+$, where Z^+ is the set of positive integers. Job type J_j is defined by a set of *stages* $\{p_{jk}, k = 1, \ldots, \lambda_j\}$, that is *partially ordered* through a set of *precedence constraints*. Furthermore, each job stage p_{jk} is associated with a *conjunctive* resource allocation requirement, formally expressed by an m-dimensional vector $a_{p_{jk}}$, with $a_{p_{jk}}[i], i = 1, \ldots, m$, indicating how many units of resource R_i are required to support the stage execution.

In the Petri net modeling framework, the workflow logic associated with job type J_j, and encoded in the partial ordering of the corresponding stage set $\{p_{jk}\}$, is represented by a particular net structure, known as *Simple Sequential Process (S^2P)* [7]. This net structure is formally defined by an ordinary strongly connected state machine $\mathcal{N}_j = (P_{S_j} \cup \{p_{0_j}\}, T_j, W_j)$ such that (i) $P_{S_j} \neq \emptyset$, $p_{0_j} \notin P_{S_j}$, and (ii) every circuit of $\mathcal{N}_j$ contains $\{p_{0_j}\}$. Each place $p \in P_{S_j}$ corresponds to a job stage of J_j, while place p_{0_j} is characterized as the *idle place*, since its marking corresponds to jobs waiting to initiate the execution of the considered job type. Furthermore, in order to facilitate the subsequent developments, we define the *descendant* set of a place (equiv., stage) $p \in P_{S_j}$, by $D_p = \{q \in P_{S_j} \mid \exists$ a path $\pi =< p, \ldots, q >$ s.t. $p_{0_j} \notin \pi\}$. Similarly, the set of resources supporting the execution of stage $p \in P_{S_j}$ is denoted by $Q_p = \{R_i \in \mathcal{R} \mid a_p[i] > 0\}$. Obviously, the entire set of resource allocation sequences according to which a job instance of type J_j can be executed, is given by the number of circuits in net $\mathcal{N}_j$ that are starting from place p_{0_j}.[4]

The PN-based modeling of the resource allocation dynamics taking place in the C/D-RAS class is completed by interconnecting the state machines modeling the various system job types, through *resource places*, modeling the availability of the various resource types (similar to the resource allocation modeling nets of [2, 7, 5]). The resulting PN class is characterized as *System of Simple Sequential Processes with General Resource Requirements*, and it will be denoted by

[4]Notice that the notion of *disjunction* implied by this statement is more general – i.e., subsumes – the notion of disjunction defined in the taxonomy of [17]. Specifically, the C/D-RAS model proposed herein allows for the representation of job types in which the execution of a certain stage through a particular alternative constrains the ways in which the remaining processing will be carried out; this behavior is not covered by the C/D-RAS class defined in [17].

S^3PGR^2. Formally, it is defined as follows:

Definition 1 *A* well-marked S^3PGR^2 *net is a marked PN* $\mathcal{N} = (P, T, W, M_0)$ *such that*

i. $P = P_S \cup P_0 \cup P_R$, *where* $P_S = \bigcup_{j=1}^{n} P_{S_j}$ *s.t.* $P_{S_i} \cap P_{S_j} = \emptyset, \forall i \neq j$, $P_0 = \bigcup_{j=1}^{n} \{p_{0_j}\}$ *s.t.* $P_0 \cap P_S = \emptyset$, *and* $P_R = \{r_1, \ldots, r_m\}$ *s.t.* $(P_S \cup P_0) \cap P_R = \emptyset$.

ii. $T = \bigcup_{j=1}^{n} T_j$.

iii. $W = W_S \cup W_R$, *where* $W_S : ((P_S \cup P_0) \times T) \cup (T \times (P_S \cup P_0)) \rightarrow \{0, 1\}$ *s.t.* $\forall j \neq i, ((P_{S_j} \cup P_{0_j}) \times T_i) \cup (T_i \times (P_{S_j} \cup P_{0_j})) \rightarrow \{0\}$, *and* $W_R : (P_R \times T) \cup (T \times P_R) \rightarrow Z^+$.

iv. $\forall j,\ j = 1, \ldots, n$, *the subnet* $\mathcal{N}_j$ *generated by* $P_{S_j} \cup \{p_{0_j}\} \cup T_j$ *is a strongly connected state machine such that every circuit contains* $\{p_{0_j}\}$.

v. $\forall r \in P_R$, $\exists$ *a unique minimal p-semiflow* y_r *s.t.* $\|y_r\| \cap P_R = \{r\}$, $\|y_r\| \cap P_0 = \emptyset$, $\|y_r\| \cap P_S \neq \emptyset$, *and* $y_r(r) = 1$. *Furthermore,* $P_S = \bigcup_{r \in P_R} (\|y_r\| - P_R)$.

vi. $\mathcal{N}$ *is pure and strongly connected.*

vii. $\forall p \in P_S,\ M_0(p) = 0$; $\forall r \in P_R,\ M_0(r) \geq max_{p \in \|y_r\|} y_r(p)$; *and* $\forall p_{0_j} \in P_0,\ M_0(p_{0_j}) \geq 1$.

We note that the proposed S^3PGR^2 net structure is a weighted generalization of the ES^3PR net, proposed in [20]. As in [20], let the set of *holders* of a resource place r ($\in P_R$) be defined by $H(r) = \|y_r\| - \{r\}$. Then, given an S^3PGR^2 net representing a C/D-RAS, it follows that $\forall r_i \in P_R$, $\sum_{p \in \{r_i\} \cup H(r_i)} y_{r_i}(p) \cdot M(p) = M_0(r_i) \equiv C_i$.

Finally, the subsequent theoretical developments involve also the notion of the *modified* S^3PGR^2 markings, formally defined as follows:

Definition 2 *Given a well-marked* S^3PGR^2 *net* $\mathcal{N} = (P_S \cup P_0 \cup P_R, T, W, M_0)$ *and* $M \in R(\mathcal{N}, M_0)$, *the* modified marking $\overline{M}$ *is defined by*

$$\overline{M}(p) = \begin{cases} M(p) & \text{if } p \notin P_0 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Furthermore, the set of all modified markings induced by the reachable markings is defined by $\overline{R(\mathcal{N}, M_0)} = \{\overline{M} \mid M \in R(\mathcal{N}, M_0)\}$

Example 1 (modified from [20]) Figure 1 shows the S^3PGR^2 net representing a C/D-RAS consisting of three resource types R_1, R_2, and R_3, with capacities $C_1 = C_2 = 4$, $C_3 = 2$, and supporting two job types J_1 and J_2. Job type J_1 (resp., J_2) is defined by the set of partially ordered job stages $\{p_{11}, p_{12}, p_{13}, p_{14}, p_{15}\}$ (resp., $\{p_{21}, p_{22}, p_{23}, p_{24}\}$). The conjunctive resource requirements associated with the various job stages are as follows: $a_{p_{11}} = (2, 0, 0)$, $a_{p_{12}} = (3, 0, 0)$,

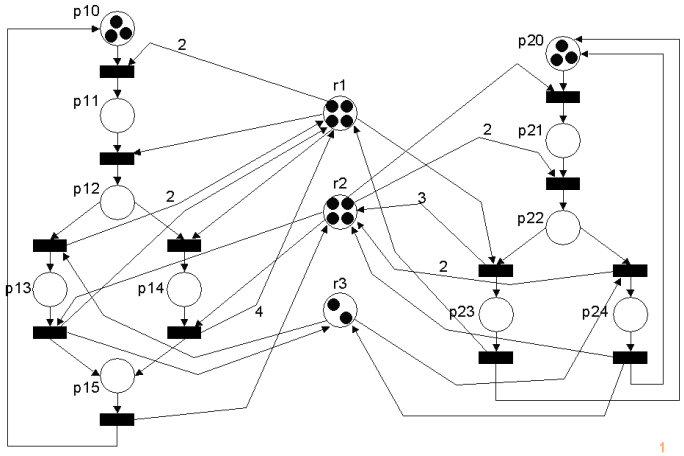

Figure 1: An example S^3PGR^2 net

$a_{p_{13}} = (1,0,1)$, $a_{p_{14}} = (4,0,0)$, $a_{p_{15}} = (0,1,0)$, $a_{p_{21}} = (0,1,0)$, $a_{p_{22}} = (0,3,0)$, $a_{p_{23}} = (1,0,0)$, and $a_{p_{24}} = (0,1,1)$. Hence, $Q_{p_{11}} = Q_{p_{12}} = Q_{p_{14}} = Q_{p_{23}} = \{R_1\}$, $Q_{p_{13}} = \{R_1, R_3\}$, $Q_{p_{15}} = Q_{p_{21}} = Q_{p_{22}} = \{R_2\}$, and $Q_{p_{24}} = \{R_2, R_3\}$. Also, $D_{p_{11}} = \{p_{12}, p_{13}, p_{14}, p_{15}\}$, $D_{p_{12}} = \{p_{13}, p_{14}, p_{15}\}$, $D_{p_{13}} = \{p_{15}\}$, $D_{p_{14}} = \{p_{15}\}$, $D_{p_{15}} = \emptyset$, $D_{p_{21}} = \{p_{22}, p_{23}, p_{24}\}$, $D_{p_{22}} = \{p_{23}, p_{24}\}$, $D_{p_{23}} = \emptyset$, and $D_{p_{24}} = \emptyset$. Finally, assuming that places are ordered in the flow matrix Θ according to the sequence $< p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{20}, p_{21}, p_{22}, p_{23}, p_{24}, r_1, r_2, r_3 >$, the p-semiflows corresponding to the system resource types are: $y_1 = < 0,2,3,1,4,0,0,0,0,1,0,1,0,0 >^T$, $y_2 = < 0,0,0,0,0,1,0,1,3,0,1,0,1,0 >^T$, and $y_3 = < 0,0,0,1,0,0,0,0,0,0,1,0,0,1 >^T$. $\diamond$

3 Liveness Analysis of the S^3PGR^2 Net

In this section we derive some important properties related to the liveness of the S^3PGR^2 net. It turns out that the net (non-)liveness is strongly dependent on the development of a special structure in the net dynamics expressed by the modified reachability space, that will be characterized as a *deadly marked siphon*. A formal characterization of this concept is as follows:

Definition 3 *Consider a marked PN* $\mathcal{N} = (P, T, W, M_0)$*. A siphon* $S \subseteq P$ *is said to be* deadly marked *at* $M \in R(\mathcal{N}, M_0)$ *iff* $\forall t \in {}^\bullet S$*,* t *is disabled by some* $p \in S$*.*

Furthermore, an immediate implication of Definition 3 is the following:

Lemma 1 *Consider a marked PN* $\mathcal{N} = (P, T, W, M_0)$*, and let* $S \subseteq P$ *be a deadly marked siphon at* $M \in R(\mathcal{N}, M_0)$*. Then, (i)* $\forall t \in {}^\bullet S$*,* t *is a dead transition in* M*, and (ii)* $\forall M' \in R(\mathcal{N}, M)$*,* S *is deadly marked.*

Hence, the concept of deadly marked siphon established by Definition 3, is a generalization to the class of non-ordinary Petri nets of the concept of empty siphon that has been employed in the structural analysis of ordinary Petri nets.[5] Indeed, the rest of this section establishes that the (non-)liveness of S^3PGR^2 nets can be attributed to the development of a particular type of deadly marked siphons in the underlying net dynamics characterized by the space of modified reachable markings. We derive this result through a series of lemmata.

The first lemma in this series is a straightforward implication of Definition 1:

Lemma 2 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well-marked S^3PGR^2 net. Then, $\mathcal{N}$ is quasi-live.*

Proof: Consider $t \in T_i,\ i \in \{1, \ldots, n\}$. Item 4 in Definition 1 implies that there exists a path $\pi =< p_{0_i} \ldots t >$. Furthermore, item 7 in Definition 1 implies that the firing sequence σ that fires each transition in path π once, and in the order defined by path π, is feasible in M_0. $\diamond$

Lemma 2 has, in turn, the following implication:

Lemma 3 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well-marked S^3PGR^2 net. If there exists a dead transition at $M \in R(\mathcal{N}, M_0)$, then $M_0 \notin R(\mathcal{N}, M)$.*

[5] While the notion of empty siphon has been very useful for explaining non-liveness in many ordinary PN classes, including the existence of (partial) deadlock in RAS classes where acquisition of each resource type is limited to one unit at a time, it fails to effectively characterize deadlock in the considered C/D-RAS class, since the operational characteristic of multiple resource acquisitions allows a partial deadlock to occur even if there is no empty siphon. As it was mentioned in the Introduction, the works presented in [4, 19] have also attempted to generalize the notion of empty siphon to non-ordinary Petri nets. Specifically, the work in [4] proposes the concept of the *max-marked siphon*: Given a marked Petri net $\mathcal{N} = (P, T, W, M_0)$, a siphon $S \subseteq P$ is said to be *max-marked* at $M \in R(\mathcal{N}, M_0)$ iff $\exists p \in S$ s.t. $M(p) \geq max_{p^\bullet}$, where $max_{p^\bullet} = max_{t \in p^\bullet} W(p, t)$. Furthermore, a marked PN $\mathcal{N}$ is said to satisfy the *max-cs property* if $\forall M \in R(\mathcal{N}, M_0)$, every siphon is max-marked. More recently, the work presented in [3] has established that S^3PGR^2 nets that satisfy the max-cs property, are live. However, it can be shown that the max-cs property of [4], is only a sufficient condition for the liveness of S^3PGR^2. The work in [19] attempts to extend the notion of empty siphon only for the class of S^3PGR^2 nets (or, in the authors' terminology, S^4PR nets). Hence, [19] introduces a siphon-based necessary condition for the presence of dead transitions in the net dynamics (and therefore, deadlock in the dynamics of the underlying RAS), but it fails to establish that this is a complete (i.e., necessary and sufficient) characterization of C/D-RAS deadlock. Furthermore, none of the aforementioned works provides a polynomial-complexity solution to the control problem of C/D-RAS deadlock avoidance.

On the other hand, in this section we show that the concept of the deadly marked siphon introduced in Definition 3 provides the basis for a *sufficient and necessary condition* for the liveness of the considered net class. Furthermore, in the next section, this novel characterization of liveness facilitates also the development of an (computationally) efficient deadlock avoidance policy for the considered class of C/D-RAS. Finally, Section 5 exploits the same characterization in order to develop an algebraic liveness test for PN's modeling C/D-RAS, which can further function as an "automatic" correctness verification tool for a large class of DAP's proposed for these RAS.

Proof: Let t be a dead transition at M. Suppose $M_0 \in R(\mathcal{N}, M)$. Then, from Lemma 2, there exists $M' \in R(\mathcal{N}, M_0)$ s.t. $M'[t\rangle$, which is a contradiction. $\diamond$

The next lemma relates the notion of total deadlock in well-marked S^3PGR^2 nets to the notion of deadly marked siphon.

Lemma 4 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well-marked S^3PGR^2 net. If marking $M \in R(\mathcal{N}, M_0)$ is a total deadlock, then there exists a deadly marked siphon at M.*

Proof: Since all transitions are dead at M, $\forall t \in T$, t is disabled by some $p \in P$. Let S be the set of disabling places. Since $S^\bullet = T$, ${}^\bullet S \subseteq S^\bullet$. Therefore, S is a siphon. It is clear that S is deadly marked from the construction. $\diamond$

Although the notion of deadly marked siphon is adequate for characterizing total deadlocks developing in S^3PGR^2 nets and the underlying C/D-RAS, the siphon-based characterization of *partial* deadlocks developing in these systems, requires a more elaborate siphon construct. The next lemma and the ensuing theorem establish this result.

Lemma 5 [6] *Let $\mathcal{N} = (P, T, W, M_0)$ be a well-marked S^3PGR^2 net. If there exists a dead transition at $M \in R(\mathcal{N}, M_0)$, then there exists a marking $M' \in R(\mathcal{N}, M)$ with its modified marking $\overline{M'}$ containing a deadly marked siphon, S, such that (i) $S \cap P_R \neq \emptyset$, and (ii) every place in $S \cap P_R$ is a disabling place.*

Proof: From the fact that there exists a dead transition at M, it is easy to see that, by firing only transitions in $T - P_0^\bullet$, $\mathcal{N}$ can reach $M' \in R(\mathcal{N}, M)$ at which $\forall p \in P_S$ s.t. $M'(p) \geq 1$, $\forall t \in p^\bullet$, t is dead (otherwise, M_0 can be reached from M by firing non-dead transitions, which contradicts Lemma 3). Let $B = \{p \in P_S \mid \overline{M'}(p) \geq 1\}$. $B \neq \emptyset$ (otherwise, $M' = M_0$). Consider the set of places $S = S_R \cup S_{P1} \cup S_{P2}$, where $S_R = \bigcup_{p \in B} \bigcup_{t \in p^\bullet} \{r \in {}^\bullet t \cap P_R \mid \overline{M'}(r) < W(r,t)\}$, $S_{P1} = \{p \in P_S \mid p \in H(S_R) \wedge \overline{M'}(p) = 0\}$, and $S_{P2} = \bigcup_{\{i: \exists t \in {}^\bullet p, p \in (P_{S_i} \cap S_{P1}) \wedge {}^\bullet t \cap S_{P1} = \emptyset \wedge \forall r \in {}^\bullet t \cap S_R, \overline{M'} \geq W(r,t)\}} P_{S_i} \cup \{p_{0_i}\}$. The definitions of M', B and S_R, together with the fact that $\forall t$, $|{}^\bullet t \cap P_s| \leq 1$, imply that the set of disabling resource places $S_R \neq \emptyset$. Next we show that S is a deadly marked siphon, by considering the following cases:

Case 1: $t \in {}^\bullet S_R$. Let $r \in t^\bullet \cap S_R$. The net purity and the selection of t also imply that ${}^\bullet t \cap P_0 = \emptyset$. Let $\{p\} = {}^\bullet t \cap P_S$. Obviously, $p \in H(r)$. We consider two sub-cases: (i) $\overline{\mathbf{M'}}(\mathbf{p}) = \mathbf{0}$: Then, $p \in S_{P1}$, $t \in S_{P1}^\bullet \subseteq S^\bullet$, and t is disabled by $p \in S$. (ii) $\overline{\mathbf{M'}}(\mathbf{p}) \geq \mathbf{1}$: From the definition of $\overline{M'}$, t is dead. It follows that $\exists r'$ s.t. $\overline{M'}(r') < W(r', t)$. But then, $r' \in S_R$, $t \in S_R^\bullet \subseteq S^\bullet$, and t is disabled by $r' \in S$.

Case 2: $t \in {}^\bullet S_{P1}$. Let $p \in t^\bullet \cap S_{P1}$. We consider three sub-cases: (i) $\nexists \mathbf{r} \in \mathbf{S_R}$ **s.t.** $\mathbf{t} \in \mathbf{r}^\bullet$: $\exists p'$ s.t. $\overline{M'}(p') = 0 \wedge t \in p'^\bullet$ (otherwise, $\overline{M'}(p') \geq 1$, which contradicts the deadness of t).

[6] We would like to thank Drs. Tricas and Ezpeleta for pointing out a problem in the original version of this result.

Furthermore, $p \in S_{P1}$ implies $\exists r'$ s.t. $p \in H(r') \wedge r' \in S_R$, and by the sub-case assumptions, $t \notin r'^{\bullet}$. Therefore, $p' \in H(r')$, which implies that $p' \in S_{P1}$. It follows, then, that $t \in S_{P1}^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $p' \in S$. (ii) $\exists \mathbf{r} \in \mathbf{S_R}$ **s.t.** $\mathbf{t} \in \mathbf{r}^{\bullet} \wedge \overline{\mathbf{M}'}(\mathbf{r}) < \mathbf{W}(\mathbf{r}, \mathbf{t}) : t \in S_R^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $r \in S$. (iii) $\forall \mathbf{r} \in {}^{\bullet}\mathbf{t} \cap \mathbf{S_R}$, $\overline{\mathbf{M}'}(\mathbf{r}) \geq \mathbf{W}(\mathbf{r}, \mathbf{t})$: If $\exists p' \in S_{P1}$ s.t. $t \in p'^{\bullet}$, $t \in S_{P1}^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $p' \in S$. Otherwise, $\exists p' \in S_{P2}$ s.t. $t \in p'^{\bullet}$ (from the definition of S_{P2}). Furthermore, $\overline{M'}(p') = 0$ (otherwise, $\exists r \in S_R$ s.t. $t \in r^{\bullet}$ which contradicts the sub-case assumption). Therefore, $t \in S_{P2}^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $p' \in S$.

Case 3: $t \in {}^{\bullet}(S_{P2} - S_{P1})$. Let $p \in t^{\bullet} \cap S_{P2}$. We consider two sub-cases: (i) $\overline{\mathbf{M}'}(\mathbf{p}) = \mathbf{0}$: $p \notin H(S_R)$. Therefore, $\exists p' \in S_{P2}$ s.t. $\overline{M'}(p') = 0 \wedge t \in p'^{\bullet}$ (otherwise, $\exists r \in S_R$ s.t. $t \in r^{\bullet} \wedge \overline{M'}(r) < W(r,t)$, which implies $p \in H(S_R)$). It follows that $t \in S_{P2}^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $p' \in S$. (ii) $\overline{\mathbf{M}'}(\mathbf{p}) \geq \mathbf{1}$: If $\exists p' \in S_{P2}$ s.t. $t \in p'^{\bullet} \wedge \overline{M'}(p') = 0$, $t \in S_{P2}^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $p' \in S$. Otherwise, $\exists r \in S_R$ s.t. $t \in r^{\bullet} \wedge \overline{M'}(r) < W(r,t)$. It follows that $t \in S_R^{\bullet} \subseteq S^{\bullet}$, and t is disabled by $r \in S$.

Finally, the fact that every resource place in S disables some transition in $\mathcal{N}$ results immediately from the definition of S_R, S_{P1} and S_{P2}. $\diamond$

Theorem 1 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well-marked S^3PGR^2 net. The net is* live *iff the space of modified reachable markings, $\overline{R(\mathcal{N}, M_0)}$, contains no deadly marked siphon such that (i) $S \cap P_R \neq \emptyset$, and (ii) every place in $S \cap P_R$ is a disabling place.*

Proof: (i) To show the necessity part, suppose that there exists a marking $M \in R(\mathcal{N}, M_0)$, with its modified marking, $\overline{M}$, containing a deadly marked siphon, S, such that $S \cap P_R \neq \emptyset$ and every place in $S \cap P_R$ disables some transition. Let $r \in S \cap P_R$ be one of the disabling resource places, and consider $t \in r^{\bullet}$ s.t. $\overline{M}(r) < W(r,t)$. Lemma 1 implies that $\forall t' \in {}^{\bullet}r$, t' is dead in $R(\mathcal{N}, \overline{M})$. From the definition of $\overline{M}$, it follows that $\forall M' \in R(\mathcal{N}, M)$, $M'(r) \leq M(r)$. Therefore, t is a dead transition at M, which contradicts the assumption of the net liveness.

(ii) To show the sufficiency for liveness of the condition stated in Theorem 1, suppose $\mathcal{N}$ is not live. Then, $\exists\ M \in R(\mathcal{N}, M_0)$ and $t \in T$ s.t. t is dead at M. But then, Lemma 5 implies that there exists a marking $\overline{M'} \in \overline{R(\mathcal{N}, M)} \subseteq \overline{R(\mathcal{N}, M_0)}$, containing a deadly marked siphon such that $S \cap P_R \neq \emptyset$ and every place in $S \cap P_R$ disables some transition. $\diamond$

We remark that in the class of S^3PGR^2 nets, weak liveness does not imply liveness. This is due to the existence of partial deadlocks in the underlying C/D-RAS, which might not lead to total deadlock.

4 A Polynomial-Complexity DAP for C/D-RAS

Deadlock avoidance in sequential RAS In this section we consider the problem of controlling the resource allocation taking place in any given C/D-RAS configuration, in a way that

the behavior of the controlled system is deadlock and induced deadlock-free.[7] In its broader statement, this control problem is formally known as *deadlock avoidance* in sequential RAS, and it has been formally characterized in [16, 17], by means of the topological structure of the state transition diagram (STD) of the automata modeling the behavioral space generated by these systems. Furthermore, the works of [16, 17] have established that, given a general sequential RAS configuration, the implementation of the *optimal* deadlock avoidance policy, that establishes deadlock-free operation by imposing the minimal restriction on the system operation, is NP-hard [9], and therefore, real-time implementable solutions will necessarily be suboptimal – i.e., in certain cases, they might constrain unnecessarily the system operation, in their effort to establish deadlock-free behavior.

The rest of this section develops such a suboptimal polynomial-complexity deadlock avoidance policy (DAP) that is appropriate for the class of C/D-RAS. The policy can be perceived as a generalization of the RUN DAP, originally developed in [16] for SU-RAS configurations; for that reason, we call the policy C/D-RUN. In the subsequent development, first we provide the formal policy definition, and demonstrate its implementation through an example. Next, we show that its implementation in the PN formalism of Section 2 is of polynomial complexity with respect to the size of the original system configuration. Finally, we establish the policy correctness, by establishing the liveness of the PN modeling the controlled system behavior.

C/D-RUN Consider a C/D-RAS, as defined in Section 2, and let $o_i \equiv O(R_i),\ O() : \mathcal{R} \rightarrow \{1, \ldots, m\}$ be any partial order imposed on the resource set $\mathcal{R}$. Given $p \in P_S$, let $\rho_p^{max} = \max\{o_i \mid a_p[i] > 0, i = 1, \ldots, m\}$ and $\rho_p^{min} = \min\{o_i \mid a_p[i] > 0, i = 1, \ldots, m\}$. Also, let $L_p = \{q \mid q \in (p^\bullet)^\bullet \cap P_S \wedge \rho_q^{max} = \min_{v \in (p^\bullet)^\bullet \cap P_S} \rho_v^{max}\}$; by convention, $L_p = \emptyset$ if $(p^\bullet)^\bullet \cap P_0 \neq \emptyset$. Then:

i. The *neighborhood* set N_p of $p \in P_S$ is defined recursively by the following equation:

$$N_p = \{p\} \cup \{q \mid q \in \bigcup_{v \in L_p} N_v \ \wedge \ \rho_p^{min} \leq \rho_q^{max}\} \tag{2}$$

ii. The *adjusted* resource allocation requirement $\hat{a}_p$ for $p \in P_S \cup P_0$, defined by C/D-RUN implementation under partial ordering $O()$, is given by: $\forall i = 1, \ldots, m$,

$$\hat{a}_p[i] = \begin{cases} \max\{a_q[i] \mid q \in N_p\} & \text{if } p \in P_S \wedge o_i \geq \rho_p^{min} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

iii. Finally, the policy-imposed constraint on the system operation is expressed by the requirement that *no resource is over-allocated with respect to the adjusted stage requirements specified by Equation 3.* $\diamond$

[7] An *induced* deadlock occurs when the policy logic itself blocks the further advancement of any job processed through the system.

Example 2: Let us consider the implementation of the C/D-RUN DAP on the S^3PGR^2 net of Figure 1, under the resource ordering $o_1 = 2$, $o_2 = 3$, $o_3 = 1$. The neighborhood sets associated with the various processing stages $p \in P_S$ can be efficiently computed starting from the terminal stages in the partially ordered sets corresponding to the various system job types, and proceeding backwards. Hence, working on the stages of job type J_1, first, we obtain: $\rho_{p_{15}}^{max} = 3$, $L_{p_{15}} = \emptyset$, $N_{p_{15}} = \{p_{15}\}$. Subsequently, we have that $\rho_{p_{13}}^{max} = 2$, $\rho_{p_{13}}^{min} = 1$, $L_{p_{13}} = \{p_{15}\}$, and since $\rho_{p_{13}}^{min} = 3 < 5 = \rho_{p_{15}}^{max}$, $N_{p_{13}} = \{p_{13}, p_{15}\}$. Similarly, $\rho_{p_{14}}^{max} = \rho_{p_{14}}^{min} = 2$, $L_{p_{14}} = \{p_{15}\}$, and $N_{p_{14}} = \{p_{14}, p_{15}\}$. Continuing in the same way with the remaining stages of job type J_1, we also get: $N_{p_{12}} = \{p_{12}, p_{13}, p_{14}, p_{15}\}$ and $N_{p_{11}} = \{p_{11}, p_{12}, p_{13}, p_{14}, p_{15}\}$. For the terminal stages of job type J_2, we have: $\rho_{p_{23}}^{max} = 2$, $L_{p_{23}} = \emptyset$, $N_{p_{23}} = \{p_{23}\}$ and $\rho_{p_{24}}^{max} = 3$, $L_{p_{24}} = \emptyset$, $N_{p_{24}} = \{p_{24}\}$. Subsequently, for stage p_{22}, we obtain: $L_{p_{22}} = \{p_{23}\}$, $\rho_{p_{22}}^{min} = 3 > 2 = \rho_{p_{23}}^{max}$, and therefore, $N_{p_{22}} = \{p_{22}\}$. Finally, for stage p_{21}, we have: $L_{p_{21}} = \{p_{22}\}$ and $\rho_{p_{21}}^{min} = 3 = \rho_{p_{22}}^{max}$, and therefore, $N_{p_{21}} = \{p_{21}, p_{22}\}$.

Once the stage neighborhood sets have been computed, the stage adjusted resource allocation requirements are obtained directly from Equation 3, as follows: $\hat{a}_{p_{11}} = (4, 1, 0)$, $\hat{a}_{p_{12}} = (4, 1, 0)$, $\hat{a}_{p_{13}} = (1, 1, 1)$, $\hat{a}_{p_{14}} = (4, 1, 0)$, $\hat{a}_{p_{15}} = (0, 1, 0)$, $\hat{a}_{p_{21}} = (0, 3, 0)$, $\hat{a}_{p_{22}} = (0, 3, 0)$, $\hat{a}_{p_{23}} = (1, 0, 0)$, and $\hat{a}_{p_{24}} = (0, 1, 1)$.

Finally, given the above stage adjusted resource allocation requirements, the constraint imposed by the policy on the system resource allocation can be expressed by the following set of linear inequalities in vector M_S, i.e., the projection of the PN marking M modeling the RAS state, on the sub-space defined by the stage place set, P_S.

$$\begin{bmatrix} 4 & 4 & 1 & 4 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 3 & 3 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot M_S \leq \begin{bmatrix} 4 \\ 4 \\ 2 \end{bmatrix} \tag{4}$$

$\diamond$

The ability to express the resource allocation constraints imposed by any C/D-RUN implementation in the form of Equation 4, gives the policy the characterization of *algebraic* DAP. Next, we show that this algebraic form of the policy constraints allows the modeling of the controlled system behavior by another PN, obtained from the S^3PGR^2 characterization of the original system behavior through the addition of a control subnet superimposing the policy-defining logic. In a later section we also show that this PN-based characterization of the controlled system behavior also allows the relaxation of the policy imposed constraints through PN structural analysis.

CS^3PGR^2 nets: a PN-based implementation of C/D-RUN Given an S^3PGR^2 net modeling a C/D-RAS configuration, the control logic corresponding to the C/D-RUN instantiation resulting from any resource (partial) ordering, can be encoded by the super-imposition

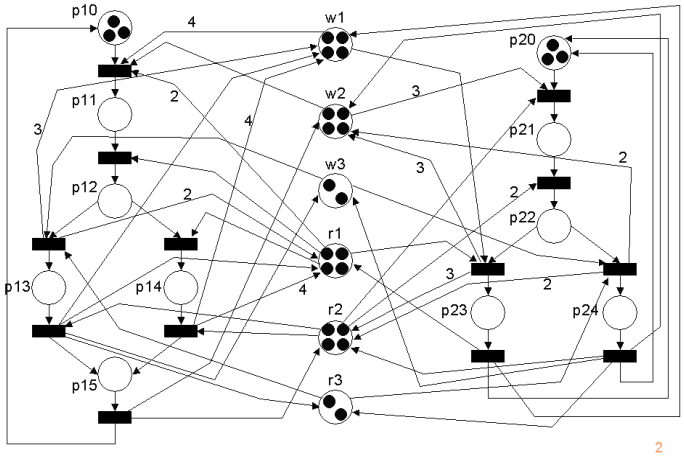

Figure 2: The CS^3PGR^2 net corresponding to the C/D-RUN implementation on the C/D-RAS of Figure 1, under the resource ordering $o_1 = 2$, $o_2 = 3$, $o_3 = 1$

to the original net of a *control subnet*, that traces the allocation of the system resources with respect to the stage adjusted resource requirements specified by Equation 3 (similar to the approach of [10, 23]). This control subnet is constructed as follows: (i) It contains a control place w_i for every resource place r_i, with initial marking $M_0(w_i) = C_i$, $\forall i$. Let $P_W = \{w_1, w_2, \ldots, w_m\}$ (ii) The introduced control places are connected to the original net transitions according to the following logic: $\forall t \in T$, let $\{p\} = {}^\bullet t \cap (P_S \cup P_0)$, and $\{q\} = t^\bullet \cap (P_S \cup P_0)$. Then, $\forall w_i \in P_W$, $W(w_i, t) = \hat{a}_q[i] - \hat{a}_p[i]$, if $\hat{a}_q[i] - \hat{a}_p[i] > 0$; $W(t, w_i) = \hat{a}_p[i] - \hat{a}_q[i]$, if $\hat{a}_q[i] - \hat{a}_p[i] < 0$; $W(w_i, t) = W(t, w_i) = 0$, otherwise. The resulting net is characterized as CS^3PGR^2: *Controlled System of Simple Sequential Processes with General Resource Requirements*. It should be easy to see that the CS^3PGR^2 net structure still belongs to the class of S^3PGR^2, with control places w_i playing the role of additional resources. We state this effect in the following lemma.

Lemma 6 *Let $\mathcal{N} = (P_s \cup P_0 \cup P_R \cup P_W, T, W, M_0)$ be a (well-marked) CS^3PGR^2 net corresponding to a C/D-RUN implementation. Then, $\forall w_i \in P_W$, $\exists$ a unique minimal p-semiflow y_{w_i} s.t. $\|y_{w_i}\| \cap P_W = \{w_i\}$, $\|y_{w_i}\| \cap P_R = \emptyset$, $\|y_{w_i}\| \cap P_0 = \emptyset$, $\|y_{w_i}\| \cap P_S \neq \emptyset$, and $y_{w_i}(w_i) = 1$. Furthermore, $\sum_{p \in \{w_i\} \cup H(w_i)} y_{w_i}(p) \cdot M(p) = M_0(w_i) \equiv C_i$, where $H(w_i)$ extends the notion of resource holders to control places.*

The CS^3PGR^2 net corresponding to the C/D-RUN implementation of Example 2 is given in Figure 2. From this figure and the discussion in Example 2, it should be clear that the policy implementation on any given C/D-RAS configuration is of polynomial complexity with respect to the size of the original system (defined by the number of places, $|P|$, in its S^3PGR^2

representation). The next theorem formally states and proves this complexity result.

Theorem 2 *Given an S^3PGR^2 net $\mathcal{N} = (P_S \cup P_0 \cup P_R, T, W, M_0)$, the control subnet imposing an instantiation of C/D-RUN on the net operation is of size $O(|P_S|+|P_R|)$, while the evaluation of the flow relation $W_W : (P_W \times T) \cup (T \times P_W) \rightarrow Z^+$ is of complexity no higher than $O(n \cdot |P_R| \cdot \max_{j=1}^{n} |P_{S_j}|^2)$, where n is the number of the supported job types.*

Proof: It can be seen from the CS^3PGR^2 net construction, that the control subnet consists of $|P_R|$ control places, each connected to $O(|T|)$ transitions, in the worst case. Hence, the result regarding the size of the control net follows from the fact that $|T|$ is of $O(|P_S|)$.

The complexity of the evaluation of the flow relation $W_W : (P_W \times T) \cup (T \times P_W) \rightarrow Z^+$, is established from the following observations: Given a job type J_j, (i) the evaluation of ρ_p^{max} and ρ_p^{min} for all places $p \in P_{S_j}$ requires $O(|P_{S_j}| \cdot |P_R|)$ operations. (ii) Subsequently, the construction of the neighborhood sets requires, in the worst case, $1+2+\ldots+|S_j| \Rightarrow O(|P_{S_j}|^2)$ operations. (iii) Finally, the evaluation of the adjusted resource allocation requirements requires $O(|P_R| \cdot |P_{S_j}|^2)$ operations. (iv) The conclusion follows by observing also that $|T|$ is of $O(|P_S|)$. $\diamond$

Next, we prove the correctness of the proposed C/D-RUN DAP, by showing that it negates the development of deadly marked siphons of the type specified in Theorem 1.

Lemma 7 *Let $\mathcal{N} = (P_s \cup P_0 \cup P_R \cup P_W, T, W, M_0)$ be a (well-marked) CS^3PGR^2 net corresponding to a C/D-RUN implementation. Then, $\overline{R(\mathcal{N}, M_0)}$ contains no deadly marked siphon S such that (i) $S \cap (P_R \cup P_W) \neq \emptyset$, and (ii) every place $p \in S \cap (P_R \cup P_W)$ is disabling.*

Proof: We prove the above result by contradiction. Hence, for the sake of the argument, suppose that there exists $\overline{M'} \in \overline{R(\mathcal{N}, M_0)}$ containing a deadly marked siphon S such that $S \cap (P_R \cup P_W) \neq \emptyset$ and every place in $S \cap (P_R \cup P_W)$ disables some transition of $\mathcal{N}$. Then, from Lemma 1, and working as in the proof of Lemma 5, we can construct a reachable marking $M \in R(\mathcal{N}, M')$, s.t. (i) $\overline{M}$ contains the deadly marked siphon S, (ii) $\{p \in P_S : \overline{M}(p) \geq 1\} \neq \emptyset$, (iii) $\forall p \in P_S$ s.t. $\overline{M}(p) \geq 1$, $\forall t \in p^\bullet$, t is dead in $R(\mathcal{N}, M)$, and (iv) $(S_R \cup S_W) \neq \emptyset$, where S_R (resp., S_W) $= \{p \in P_R$ (resp., $P_W) : M(p) < M_0(p)\}$. Consider $q_1 \in S_R \cup S_W$. From the construction of marking M and the definition of S_R and S_W, $\exists p_1 \in P_S$ s.t. $M(p_1) \geq 1 \wedge p_1 \in H(q_1)$, and $\forall t \in p_1^\bullet$, t is dead. We select $t_1 \in p_1^\bullet$ s.t. $t_1 \in {}^\bullet s_1$, where $s_1 \in L_{p_1}$. Since t_1 is dead, and $M(p_1) \geq 1$, $\exists q_2 \in S_R \cup S_W$ disabling t_1. Repeating the above argument on place q_2, and considering the finiteness of the set $S_R \cup S_W$, we conclude that there exists a set $\{q_1, q_2, \ldots, q_k\} \subseteq S_R \cup S_W$, and a corresponding set $\{p_1, p_2, \ldots, p_k\} \subseteq P_S$ s.t. $p_i, i = 1, \ldots, k$, is a marked place belonging to $H(q_i)$. Furthermore, $t_i \in p_i^\bullet$ is disabled by q_{i+1}, $i = 1, \ldots, k-1$, and t_k is disabled by some q_i, $i \in \{1, \ldots, k\}$. Next, consider a place $q_{i^*} \in \{q_1, q_2, \ldots, q_k\}$ where i^*

$= \arg\min_{i=1,\ldots,k}\{o(q_i)\}$, and $o(q_i)$ is the partial ordering used in the C/D-RUN implementation, extended to $P_R \cup P_W$ by imposing the same order to a resource, r_i, and its corresponding control place, w_i. Defining p_{i^*} and s_{i^*} as in the above discussion, we consider four cases:

Case 1: $p_{i^*} \in H(r_{i^*})$ and $s_{i^*} \in H(r_{i^*+1})$. Since $\rho^{min}_{p_{i^*}} \leq o_{i^*} \leq o_{i^*+1} \leq \rho^{max}_{s_{i^*}}$ and $s_{i^*} \in L_{p_{i^*}}$, $s_{i^*} \in N_{p_{i^*}}$. Furthermore, since $\rho^{min}_{p_{i^*}} \leq o_{i^*+1}$, $p_{i^*} \in H(w_{i^*+1})$, where w_{i^*+1} is the control place corresponding to resource r_{i^*+1}.

Case 2: $p_{i^*} \notin H(r_{i^*})$ and $s_{i^*} \in H(r_{i^*+1})$. Then, there exists $v \in D_{p_{i^*}}$ s.t. $v \in N_{p_{i^*}}$, and $a_v[r_{i^*}] > 0$. Also, from the definition of C/D-RUN DAP, $\rho^{min}_{p_{i^*}} \leq o_{i^*}$. Hence, it follows that $\rho^{min}_{p_{i^*}} \leq o_{i^*} \leq o_{i^*+1} \leq \rho^{max}_{s_{i^*}}$. Therefore, from $s_{i^*} \in L_{p_{i^*}}$, $s_{i^*} \in N_{p_{i^*}}$. Since $\rho^{min}_{p_{i^*}} \leq o_{i^*+1}$, $p_{i^*} \in H(w_{i^*+1})$, where w_{i^*+1} is the control place corresponding to resource r_{i^*+1}.

Case 3: $p_{i^*} \in H(r_{i^*})$ and $s_{i^*} \notin H(r_{i^*+1})$. There exists $v \in D_{s_{i^*}}$ such that $v \in N_{s_{i^*}}$, and $a_v[r_{i^*+1}] > 0$. Furthermore, from the definition of C/D-RUN DAP, $\rho^{min}_{s_{i^*}} \leq o_{i^*+1}$. Since $v \in N_{s_{i^*}}$, $\rho^{min}_{p_{i^*}} \leq o_{i^*} \leq o_{i^*+1} \leq \rho^{max}_{v}$, and $s_{i^*} \in L_{p_{i^*}}$, it follows that $v \in N_{p_{i^*}}$. Since $\rho^{min}_{p_{i^*}} \leq o_{i^*+1}$, $p_{i^*} \in H(w_{i^*+1})$, where $w_{i^*+1} \equiv q_{i^*+1}$.

Case 4: $p_{i^*} \notin H(r_{i^*})$ and $s_{i^*} \notin H(r_{i^*+1})$. There exists $v \in D_{s_{i^*}}$ such that $v \in N_{s_{i^*}}$, and $a_v[r_{i^*+1}] > 0$. Also, from the definition of C/D-RUN DAP, $\rho^{min}_{p_{i^*}} \leq o_{i^*} \leq o_{i^*+1} \leq \rho^{max}_{v}$. The last set of inequalities, combined with the fact that $v \in N_{s_{i^*}}$ and $s_{i^*} \in L_{p_{i^*}}$, imply that $v \in N_{p_{i^*}}$. Since $\rho^{min}_{p_{i^*}} \leq o_{i^*+1}$, it follows that $p_{i^*} \in H(w_{i^*+1})$, where $w_{i^*+1} \equiv q_{i^*+1}$.

So, we have established that in all four cases, $p_{i^*} \in H(w_{i^*+1})$. By the policy definition, $\hat{a}_{p_{i^*}}[r_{i^*+1}] \geq \hat{a}_{s_{i^*}}[r_{i^*+1}]$, which further implies that $W(w_{i^*+1}, t_{i^*}) \equiv 0$. This establishes the contradiction for Cases (3) and (4), above. For the remaining Cases (1) and (2), we have $q_{i^*+1} \in S_R$, which combined with the fact that $p_{i^*} \in H(w_{i^*+1})$, imply that $y_{w_{i^*+1}}(p_{i^*}) - y_{r_{i^*+1}}(p_{i^*}) \geq W(r_{i^*+1}, t_{i^*})$. Furthermore, since r_{i^*+1} disables t_{i^*} in marking M, $0 \leq M(r_{i^*+1}) < W(r_{i^*+1}, t_{i^*})$. The last three inequalities, combined with the facts that $M(w_{i^*+1}) \geq 0$, $M(p_{i^*}) \geq 1$, $H(w_{i^*+1}) \supseteq H(r_{i^*+1})$ and $\forall p,\ y_{w_{i^*+1}}(p) \geq y_{r_{i^*+1}}(p)$, imply that

$$\sum_{p \in \{w_{i^*+1}\} \cup H(w_{i^*+1})} y_{w_{i^*+1}}(p) M(p) \ - \sum_{p \in \{r_{i^*+1}\} \cup H(r_{i^*+1})} y_{r_{i^*+1}}(p) M(p) \ \geq$$
$$\Big(y_{w_{i^*+1}}(p_{i^*}) - y_{r_{i^*+1}}(p_{i^*})\Big) M(p_{i^*}) + M(w_{i^*+1}) - M(r_{i^*+1}) \ > \ 0$$

But from Lemma 6,

$$\sum_{p \in \{w_{i^*+1}\} \cup H(w_{i^*+1})} y_{w_{i^*+1}}(p) M(p) \ = \sum_{p \in \{r_{i^*+1}\} \cup H(r_{i^*+1})} y_{r_{i^*+1}}(p) M(p) \ = \ C_{i^*+1}$$

which establishes the contradiction for Cases (1) and (2), above, and concludes, thus, the proof. $\diamond$

Theorem 3 *Any (well-marked) CS^3PGR^2 net corresponding to a C/D-RUN implementation, is live.*

Proof: Since $CS^3PGR^2 \subseteq S^3PGR^2$, this follows directly from Lemma 7 and Theorem 1. $\diamond$

5 Algebraic Liveness and DAP Correctness Verification Tests for C/D-RAS

Although the parameterization of C/D-RUN w.r.t. the ordering imposed on the resource set during the policy implementation essentially provides a *set* of policies for any given C/D-RUN configuration, from the DAP design standpoint, it is still a *point* solution to the underlying control problem. The results presented in this section allow the development of additional DAP's for this RAS class, by providing an algebraic correctness verification test that can be applied to a large class of DAP's tentatively proposed/synthesized for any given C/D-RAS configuration. It is also shown that, in addition to their theoretical significance of enriching the space of effectively computable and computationally efficient policies for the considered RAS class, the presented results also hold the practical potential of enhancing the flexibility of the C/D-RUN implementations presented in Section 4.

The starting point for these developments is the observation that for the class of structurally bounded non-ordinary PN's, the presence of deadly marked siphons in any given net marking can be effectively tested by means of an *Integer Programming (IP)* [21] formulation. This observation, when combined with the siphon-based structural characterization of C/D-RAS liveness provided in Theorem 1, lead to a sufficiency test for C/D-RAS liveness, that takes the convenient form of a *Mixed Integer Programming (MIP)* [21] formulation. Finally, it is easy to see that in the case of C/D-RAS DAP's for which the controlled behavior can be still modeled by a S^3PGR^2 net,[8] the aforementioned liveness sufficiency condition essentially provides an "automatic" DAP correctness verification tool. The algebraic nature of this tool allows its easy integration to higher-level analytical formulations and/or search-based techniques, aiming at the optimization of some (performance-related) aspect of the developed policy. This last possibility is briefly demonstrated by showing how the proposed DAP correctness verification tool can enhance the operational flexibility of any C/D-RUN implementation on a given C/D-RAS configuration.

Testing the existence of deadly marked siphons in a given PN marking As mentioned above, the first part of the work presented in this section develops an IP formulation that tests the presence of a deadly marked siphon in a given PN marking, M. This test is constructive, in the sense that it seeks to compute the *maximal* deadly marked siphon in the considered marking M, according to the algorithm of Theorem 4, presented below. Subsequently, Theorem 5 establishes that for the case of structurally bounded PN's, the algorithm of Theorem 4 can be effectively expressed as an IP formulation, that is of polynomial size w.r.t. the underlying

[8] As is, for instance, the case for the C/D-RUN DAP, and more generally, for all those *algebraic* DAP's that can be expressed as a set of linear inequalities in the underlying RAS state M_S, $A_P \cdot M_S \leq \mathbf{f}_P$.

marked net structure.

Theorem 4 *Given a marked PN $\mathcal{N} = (P, T, W, M_0)$ and a marking $M \in R(\mathcal{N}, M_0)$, an algorithm for computing the maximal deadly marked siphon S in M is as follows:*

Algorithm for computing the maximal deadly marked siphon in a given PN marking M

i. $S := P; \quad \mathcal{N}' := \mathcal{N}$

ii. **while** $\exists\ t \in T$ *such that* t *is fireable in the modified net* $\mathcal{N}'$ **do**

(a) Remove t *from* $\mathcal{N}'$

(b) Remove $t^\bullet$ *from* $\mathcal{N}'$

(c) $S := S - \{t^\bullet\}$

endwhile

iii. **Return** S

Proof: Consider a transition $t \in {}^\bullet S$, where S is the place set returned by the algorithm of Theorem 4. Then, t does not belong to the set of transitions removed from the net $\mathcal{N}$ during the algorithm execution, since, otherwise, $t^\bullet \notin S$. But then, t is a transition not fireable in $\mathcal{N}$, due to insufficient marking of some place(s) $p \in S$. Hence, $t \in S^\bullet$, and S is a siphon that is deadly marked in M.

To show that S is the *maximal* deadly marked siphon in M, we use contradiction. Hence, suppose that S' is another deadly marked siphon in M with $S' \backslash S \neq \emptyset$, and let $p \in S' \backslash S$. Then, since p is removed from net $\mathcal{N}$ by the considered algorithm, $\exists\ t \in {}^\bullet p \subseteq {}^\bullet S'$ that either (i) is fireable in M, or (ii) it has all of its input places removed from $\mathcal{N}$ during the algorithm execution. Case (i) violates Lemma 1, and therefore S' cannot be a deadly marked siphon. The contradiction for case (ii) is established as follows: Since S' is a siphon and $t \in {}^\bullet S'$, there exists $p' \in {}^\bullet t$ such that $p' \in S' \backslash S$ and p' has been removed from net $\mathcal{N}$ by the algorithm. Then, repeating the entire argument above on place p', and recognizing the finiteness of set $S' \backslash S$, we shall eventually identify a place $p^* \in S' \backslash S$ for which case (i) in the above argument applies. $\diamond$

In the case of *structurally bounded* nets, the algorithm of Theorem 4 can be converted to an IP formulation as follows: First, let $SB(p)$ denote a structural bound for the markings of place $p \in P$. Furthermore, let v_p, z_t and f_{tp} be *binary indicator* variables, respectively denoting the following conditions:

$$
\begin{aligned}
v_p = 1 \quad &\Longleftrightarrow \quad \text{place } p \text{ is removed during the algorithm execution,} \quad \forall p \in P \\
z_t = 1 \quad &\Longleftrightarrow \quad \text{transition } t \text{ is removed during the algorithm execution,} \quad \forall t \in T \\
f_{pt} = 1 \quad &\Longleftrightarrow \quad M(p) \geq W(p,t) \ \vee \ v_p = 1 \quad \forall W(p,t) > 0
\end{aligned}
$$

Then, we have the following theorem:

Theorem 5 *Let $\mathcal{N} = (P, T, W, M_0)$ be a structurally bounded marked PN. Then, given a marking $M \in R(\mathcal{N}, M_0)$, the maximal deadly marked siphon S contained in M is determined by:*

$$S = \{p \in P \mid v_p = 0\} \tag{5}$$

where $v_p,\ p \in P$, is obtained through the following IP formulation:

$$G(M) = \min \sum_{p \in P} v_p \tag{6}$$

s.t.

$$f_{pt} \geq \tfrac{M(p) - W(p,t) + 1}{SB(p)}, \quad \forall W(p,t) > 0 \tag{7}$$

$$f_{pt} \geq v_p, \quad \forall W(p,t) > 0 \tag{8}$$

$$z_t \geq \textstyle\sum_{p \in {}^\bullet t} f_{pt} - |{}^\bullet t| + 1, \quad \forall t \in T \tag{9}$$

$$v_p \geq z_t, \quad \forall W(t,p) > 0 \tag{10}$$

$$v_p, z_t, f_{pt} \in \{0, 1\},\ \forall p \in P,\ \forall t \in T \tag{11}$$

Proof: First we argue that places p and transitions t that are eliminated during the execution of the algorithm in Theorem 4, have $v_p = 1$ and $z_t = 1$ in the optimal solution of the IP formulation of Theorem 5. Indeed, Equation 9 together with Equation 7 imply that all transitions z_t fireable in marking M will have $z_t = 1$. Furthermore, Equation 10 implies that all places $p \in t^\bullet$ for some t with $z_t = 1$ will have $v_p = 1$, which is in agreement with the algorithm's logic regarding the elimination of the output places of removed transitions. Finally, Equation 8 combined with Equation 9 also force $z_t = 1$ for all transitions t with $v_p = 1,\ \forall p \in {}^\bullet t$. The fact that no additional place p (resp., transition t) has $v_p = 1$ (resp., $z_t = 1$), is guaranteed by the specification of the objective function in the above formulation (cf. Equation 6). $\diamond$

Sufficient liveness and DAP correctness verification tests for C/D-RAS A specialization of the key result of Theorem 5 to deadly marked siphons related to the presence of deadlock in C/D-RAS (c.f., Theorem 1) is as follows:

Theorem 6 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well marked $(C)S^3PGR^2$ net. Then, given a marking $\overline{M} \in \overline{R(\mathcal{N}, M_0)}$, the maximal deadly marked siphon S such that (i) $S \cap (P_R \cup P_W) \neq \emptyset$ and (ii) every place in $S \cap (P_R \cup P_W)$ is a disabling place at $\overline{M}$, is determined by:*

$$S = \{p \in P \mid v_p = 0\} \tag{12}$$

where $v_p,\ p \in P$, is obtained through the following IP formulation:

$$G(M) = \min \sum_{p \in P} v_p \tag{13}$$

s.t.

$$f_{pt} \geq \frac{\overline{M}(p) - W(p,t) + 1}{SB(p)}, \quad \forall W(p,t) > 0 \tag{14}$$

$$f_{pt} \geq v_p, \quad \forall W(p,t) > 0 \tag{15}$$

$$z_t \geq \textstyle\sum_{p \in {}^\bullet t} f_{pt} - |{}^\bullet t| + 1, \quad \forall t \in T \tag{16}$$

$$v_p \geq z_t, \quad \forall W(t,p) > 0 \tag{17}$$

$$\textstyle\sum_{r \in P_R \cup P_W} v_r \leq |P_R \cup P_W| - 1 \tag{18}$$

$$\textstyle\sum_{t \in r^\bullet} f_{rt} - |r^\bullet| + 1 \leq v_r, \quad \forall r \in P_R \cup P_W \tag{19}$$

$$v_p, z_t, f_{pt} \in \{0,1\},\ \forall p \in P,\ \forall t \in T \tag{20}$$

Proof: From Theorem 5 above, the set of places $S' = \{p \in P \mid v_p = 0\}$ satisfying Equations 14 $-$ 17 and 20, is a maximal deadly marked siphon at $\overline{M}$. Next, we show that Equation 19 further eliminates from the set S' all places $r \in S' \cap (P_R \cup P_W)$ that do not disable any transition $t \in r^\bullet$, while the remaining set of places, S, maintains the deadly marked siphon property. Indeed, a place $r \in S' \cap (P_R \cup P_W)$ for which $v_r = 0$ in the solution computed by the IP formulation of Theorem 5, will have $v_r = 1$ under the addition of Constraint 19, only if $\sum_{t \in r^\bullet} f_{rt} = |r^\bullet|$ (i.e., only if $\overline{M}(r)$ disables no output transition of r). Furthermore, the marking of such a place r by $v_r = 1$ does not incur the marking of any additional places and transitions, because all the transitions t with $z_t = 0$ in the solution of the IP formulation of Theorem 5, are disabled in marking $\overline{M}$; i.e., for every $t \in r^\bullet$ there still exists $p \in {}^\bullet t \backslash \{r\}$ such that $\overline{M}(p,t) < W(p,t)$. The last observation also implies that the remaining set of unmarked places $S \equiv S' \backslash \{r :\ r \in S' \cap (P_R \cup P_W) \wedge v_r = 1$ by Equation 19$\}$ is a deadly marked siphon, since $\forall t \in {}^\bullet S,\ \exists p \in S$ s.t. $t \in p^\bullet$ and p disables t at $\overline{M}$. In addition, since S' is maximal, S is also maximal. Finally, Equation 18 requires that $S \cap (P_R \cup P_W) \neq \emptyset$. $\diamond$

The next corollary is an immediate consequence of Theorem 6.

Corollary 1 *Given a reachable marking M of a $(C)S^3PGR^2$ net $\mathcal{N} = (P, T, W, M_0)$, $\overline{M}$ contains no deadly marked siphon S such that $S \cap (P_R \cup P_W) \neq \emptyset$ and all the places in $S \cap (P_R \cup P_W)$ are disabling places iff the integer program of Theorem 6 is infeasible.*

Hence, Corollary 1 provides a necessary and sufficient condition for the non-existence of deadly marked siphons with $S \cap (P_R \cup P_W) \neq \emptyset$ and all the places in $S \cap (P_R \cup P_W)$ being disabling places, in any given marking $\overline{M} \in \overline{R(\mathcal{N}, M_0)}$ of a CS^3PGR^2 net $\mathcal{N}$, modeling the controlled behavior of a system under some tentative DAP expressed by a set of invariant-imposing "control" places. The test of Corollary 1 can be extended, in principle, to a test for the non-existence of such deadly marked siphons over the entire space $\overline{R(\mathcal{N}, M_0)}$ of such a $(C)S^3PGR^2$ net $\mathcal{N} = (P, T, W, M_0)$, by: (i) turning marking vector $\overline{M}$ in the IP formulation of Theorem 6 into a variable, (ii) introducing an additional set of variables, M, representing the net reachable markings, and (iii) adding two additional sets of constraints, the first one linking variables M and $\overline{M}$ according to the logic of Equation 1, and the second one ensuring that the set of feasible values for the variable vector M is equivalent to the PN reachability space $R(\mathcal{N}, M_0)$. Unfortunately, however, any system of linear inequalities exactly characterizing the set $R(\mathcal{N}, M_0)$ is of exponential complexity with respect to the net size [18]. On the other hand, a superset of the reachability space $R(\mathcal{N}, M_0)$ is provided by the system *state equation* [6]:

$$M = M_0 + \Theta \bar{x} \tag{21}$$

$$M \geq 0,\ \bar{x} \in Z^+ \tag{22}$$

The above remarks lead to a *sufficient* condition for the non-existence of deadly marked siphons S, such that $S \cap (P_R \cup P_W) \neq \emptyset$ and all places in $S \cap (P_R \cup P_W)$ are disabling places, in the entire space $\overline{R(\mathcal{N}, M_0)}$ of a given $(C)S^3PGR^2$ net $\mathcal{N}$. Furthermore, in the light of Theorem 1, this condition constitutes a *sufficient* condition for liveness of $(C)S^3PGR^2$ nets, and therefore, a convenient correctness verification tool for any arbitrarily synthesized DAP that results in a controlled system behavior that can be modeled by the class of CS^3PGR^2 nets.

Corollary 2 *Let $\mathcal{N} = (P, T, W, M_0)$ be a well marked $(C)S^3PGR^2$ net. Then, if the mixed integer program defined by Equations 13–22 and Equation 1 is infeasible, $\mathcal{N}$ is live.*

Flexibility Enhancement of C/D-RUN DAP implementations Finally, we show how the liveness test of Corollary 2 can lead to the systematic enhancement of the operational flexibility allowed by any C/D-RUN implementation on a given C/D-RAS configuration. The underlying idea is to use the aforementioned test in order to search for maximal elements in the space of *"meaningful"* rhs vectors, $\mathbf{f}$, that can relax the original policy implementation, while maintaining its correctness. Specifically, given a S^3PGR^2 net $\mathcal{N} = (P, T, W, M_0)$, controlled by a C/D-RUN implementation that is expressed by the system of linear inequalities $\mathbf{A}_{C/D-RUN} \cdot M_S \leq \mathbf{f}_0 (\equiv \mathbf{C})$, the search space is defined by the lattice $\{\mathbf{f} \in (Z^+)^m \mid \mathbf{f}_0 \leq \mathbf{f} \leq \bar{\mathbf{f}}\}$, where the (not necessarily tight) upper bound $\bar{\mathbf{f}}$ is computed by the following IP's:

$$\forall i \in \{1,\ldots,m\},\quad \bar{f}[i] \;=\; \max_{\{x_p \,:\, p\in P_S \wedge \hat{a}_p[i]>0\}} \sum_{\{p\in P_S \;\mid\; \hat{a}_p[i]>0\}} \hat{a}_p[i]x_p$$

$$\text{s.t.}$$

$$\sum_{\{p\in P_S \;\mid\; \hat{a}_p[i]>0\}} a_p[j]x_p \le C_j,\quad \forall j \in \{1,\ldots,m\}$$

$$x_p \in Z^+,\quad \forall p \in P_S :\; \hat{a}_p[i] > 0 \qquad (23)$$

The search algorithm that identifies in the lattice defined above the *maximal* elements that lead to correct policy implementations, is based on the fact that any particular selection for the rhs vector $\mathbf{f}$ essentially defines the initial marking of the control places, $w_i \in P_W$, in the CS^3PGR^2 net modeling the controlled system behavior. A brief statement of this algorithm is as follows: Starting from the upper bound $\bar{\mathbf{f}}$, generate the arborescence of the elements defined by the '$\le$' order; at every generated node solve the corresponding MIP formulation of Equations 13–22 and 1; terminate the search along each path when an element $\mathbf{f}^* \in \{\mathbf{f} \in (Z^+)^m \mid \mathbf{f}_0 \le \mathbf{f} \le \bar{\mathbf{f}}\}$ satisfying the condition of Corollary 2, is identified. The next example applies this policy improvement algorithm on the C/D-RUN implementation that was developed in Example 2.

Example 3: Consider the C/D-RUN implementation of Example 2, which is represented by the system of linear inequalities given in Equation 4. Application of the IP formulation of Equation 23 to this policy implementation gives $\bar{\mathbf{f}} = (8, 15, 2)^T$. Subsequently, the application of the search algorithm outlined above results in the unique maximal element $\mathbf{f}^* = \{(7, 8, 2)^T\}$.[9] Hence, a correct relaxed policy implementation is defined by the following set of constraints on the system state:

$$\begin{bmatrix} 4 & 4 & 1 & 4 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 3 & 3 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot M_S \;\le\; \begin{bmatrix} 7 \\ 8 \\ 2 \end{bmatrix} \qquad (24)$$

It should be noted that the policy represented by Equation 24 admits all RAS states admitted by the policy of Equation 4, and furthermore, it admits state $M_S = (0, 1, 0, 0, 0, 1, 1, 0, 0)^T$ (equivalently, marking $M = (2, 0, 1, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 2, 3, 1, 2)^T$ of the CS^3PGR^2 net[10]) which is not admitted by the policy of Equation 4. Therefore, the new relaxed policy is more

[9] It is interesting to notice that $\mathbf{f}^*[2] < 9$ since the deadlocked state $M_S = (0, 0, 0, 0, 0, 3, 0, 0, 0)$ is admitted otherwise. Observations like this can drastically reduce the search for maximal elements performed by the proposed algorithm.

[10] The ordering of places is that used in the discussion of Example 1, with the control places P_W appended at the end.

permissive than the original one, and we can conclude that the proposed scheme provides an effective method to enhance the permissiveness of the original C/D-RUN definition. $\diamond$

6 Conclusions

This paper studied the deadlock avoidance problem for the class of C/D-RAS in the taxonomy of [17], that allows for multiple resource acquisitions and flexible routings. A new PN sub-class, the S^3PGR^2 net, was presented to effectively model the class of C/D-RAS, and some important liveness properties were derived. In particular, it was established that for the considered PN class, (non-)liveness can be effectively characterized in a modified reachability space that constitutes a projection of the original net reachability space to the subspace defined by an appropriately selected set of places. Under this new representation of the net dynamics, non-liveness can be interpreted through the development of a particular type of siphon, formally described as deadly marked siphon with a nonempty subset of resource places and with every such resource place disabling some transition. Subsequently, this siphon-based characterization of the net (non-)liveness provided the analytical framework for the development of C/D-RUN DAP, an efficiently (polynomially) computable deadlock avoidance policy for the class of C/D-RAS.

It should be noticed that the proposed concept of deadly marked siphon has broader significance/implications for the analysis of non-ordinary PN's, since it constitutes the effective generalization to this class of systems of the notion of empty siphon, that has been very instrumental in the structural analysis of ordinary PN's. Hence, generalizing the work of [5], an algorithm for computing the maximal deadly marked siphon in any given non-ordinary PN marking was also developed, and it was shown that for structurally bounded nets, it takes the convenient form of a MIP formulation. In the class of S^3PGR^2 nets, this MIP formulation provided the starting point for the development of a sufficiency test for the liveness of the underlying C/D-RAS, and for the correctness of any tentative algebraic DAP proposed for these systems. Furthermore, in the context of C/D-RUN DAP, the aforementioned test led to a systematic procedure for enhancing the flexibility of any given policy implementation.

Future work will seek the extension of the derived structural characterizations of deadlock and liveness to PN classes modeling RAS with more complex / additional behavioral features, like reworks, assembly/disassembly operations, and uncontrollable resource acquisitions.

References

[1] T. Araki, Y. Sugiyama, and T. Kasami. Complexity of the deadlock avoidance problem. In *2nd IBM Symp. Math. Found. Computer Sci.*, pages 229–257, 1977.

[2] Z. A. Banaszak and B. H. Krogh. Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows. *IEEE Transactions on Robotics & Automation*, 6(6):724–734, 1990.

[3] K. Barkaoui, A. Chaoui, and B. Zouari. Supervisory control of discrete event systems based on structure theory of petri nets. In *IEEE International Conference on Systems, Man, & Cybernetics*, pages 3750–3755. IEEE, 1997.

[4] K. Barkaoui and J.-F. Pradat-Peyre. On liveness and controlled siphons in petri nets. In *17th International Conference on Application and Theory of Petri Nets*, pages 57–72, 1996.

[5] F. Chu and X-L. Xie. Deadlock analysis of petri nets using siphons and mathematical programming. *IEEE Transactions on Robotics & Automation*, 13(6):793–804, 1997.

[6] J. Desel and W. Reisig. Place/transition petri nets. In W. Reisig and G. Rozenberg, editors, *Lectures on Petri Nets I: Basic Models*, volume 1491 of *Lecture Notes in Computer Science*, pages 122–173. Springer-Verlag, Berlin, 1998.

[7] J. Ezpeleta, J. M. Colom, and J. Martinez. A petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Transactions on Robotics & Automation*, 11:173–184, 1995.

[8] M. P. Fanti, B. Maione, S. Mascolo, and B. Turchiano. Event-based feedback control for deadlock avoidance in flexible production systems. *IEEE Transactions on Robotics & Automation*, 13:347–363, 1997.

[9] M. R. Garey and D. S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.

[10] A. Giua, F. DiCesare, and M. Silva. Generalized mutual exclusion constraints on nets with uncontrollable transitions. In *Proceedings of the 1992 IEEE International Conference on Systems, Man, & Cybernetics*, pages 974–979. IEEE, 1992.

[11] M. Lawley and S. Reveliotis. Optimal deadlock avoidance in sequential resource allocation systems : Hard and easy cases. *International Journal of Flexible Manufacturing Systems*, 13(4), 2001.

[12] M. Lawley, S. Reveliotis, and P. Ferreira. A correct and scalable deadlock avoidance policy for flexible manufacturing systems. *IEEE Transactions on Robotics & Automation*, 14(5):796–809, 1998.

[13] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[14] J. Park and S. Reveliotis. Algebraic synthesis of efficient deadlock avoidance policies for sequential resource allocation systems. *IEEE Transactions on Robotics & Automation*, 16(2):190–195, 2000.

[15] 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*, 45(1):79–98, 2001.

[16] S. A. Reveliotis and P. M. Ferreira. Deadlock avoidance policies for automated manufacturing cells. *IEEE Transactions on Robotics & Automation*, 12(6):845–857, 1996.

[17] S. A. Reveliotis, M. A. Lawley, and P. M. Ferreira. Polynomial complexity deadlock avoidance policies for sequential resource allocation systems. *IEEE Transactions on Automatic Control*, 42(10):1344–1357, 1997.

[18] M. Silva, E. Teruel, and J. M. Colom. Linear algebraic and linear programming techniques for the analysis of place/transition net systems. In W. Reisig and G. Rozenberg, editors, *Lectures on Petri Nets I: Basic Models*, volume 1491 of *Lecture Notes in Computer Science*, pages 309–373. Springer-Verlag, Berlin, 1998.

[19] F. Tricas, J. M. Colom, and J. Ezpeleta. A solution to the problem of deadlocks in concurrent systems using petri nets and integer linear programming. In *Proceedings of the 11th European Simulation Symposium*, pages 542–546, 1999.

[20] F. Tricas, F. García-Vallés, J. M. Colom, and J. Ezpeleta. A structural approach to the problem of deadlock prevention in processes with resources. In *Proceedings of the 4th Workshop on Discrete Event Systems*, pages 273–278. IEE, 1998.

[21] L. A. Wolsey. *Integer Programming*. John Wiley & Sons, 1998.

[22] K. Y. Xing, B. S. Hu, and H. X. Chen. Deadlock avoidance policy for petri net modeling of flexible manufacturing systems with shared resources. *IEEE Transactions on Automatic Control*, 41:289–295, 1996.

[23] E. Yamalidou, J. O. Moody, P. J. Antsaklis, and M. D. Lemmon. Feedback control of petri nets based on place invariants. *Automatica*, 32(1):15–28, 1996.

Jonghun Park is presently a post-doctoral fellow at School of Industrial and Systems Engineering, Georgia Institute of Technology. He received the B.S. and M.S. degrees in industrial engineering from Seoul National University in 1990 and 1992, respectively, and the Ph.D. degree in industrial and systems engineering with minor in computer science from Georgia Institute of Technology in 2000. In 1990-1997, he held researcher positions at Daewoo Motors Co., and Engineering Research Center for Advanced Control and Instrumentation. His research interests include discrete event systems, concurrency, distributed algorithms, and their applications to industrial information systems design. He will join School of Information Sciences and Technology, Pennsylvania State University as an assistant professor in August, 2001.

Spyros A. Reveliotis received the M.Sc. degree in computer systems engineering from Northeastern University, Boston, MA, in 1992, and the Ph.D. degree in industrial engineering from the University of Illinois at Urbana-Champaign in 1996. Currently, he is an Assistant Professor with the School of Industrial and Systems Engineering, Georgia Institute of Technology. His research focuses on Discrete Event System theory and its application in the modeling, analysis and control of large-scale contemporary technological environments, including flexibly automated production systems, intelligent transportation systems, and workflow management systems. Dr. Reveliotis currently serves as Associate Editor of the IEEE Transactions on Robotics and Automation. He is also a member of the IIE. He has been the recipient of a number of distinctions, including the 1998 International Conference on Robotics and Automation Kayamori Best Paper Award.