Lyapunov Stability of a Class of Discrete Event Systems

Kevin M. Passino, *Member, IEEE,* Anthony N. Michel, *Fellow, IEEE,* and Panos J. Antsaklis, *Fellow, IEEE*

***Abstract*—Discrete event systems (DES) are dynamical systems which evolve in time by the occurrence of events at possibly irregular time intervals. "Logical" DES are a class of discrete time DES with equations of motion that are most often nonlinear and discontinuous with respect to event occurrences. Recently, there has been much interest in studying the stability properties of logical DES and several definitions for stability, and methods for stability analysis have been proposed. Here we introduce a logical DES model and define stability in the sense of Lyapunov and asymptotic stability for logical DES. Then we show that a more conventional analysis of stability which employs appropriate Lyapunov functions can be used for logical DES. We provide a general characterization of the stability properties of automata-theoretic DES models, Petri nets, and finite state systems. Furthermore, the Lyapunov stability analysis approach is illustrated on a manufacturing system that processes batches of N different types of parts according to a priority scheme (to prove properties related to the machine's ability to reorient itself to achieve safe operation) and a load balancing problem in computer networks (to study the ability of the system to achieve a balanced load to minimize underutilization).**

I. INTRODUCTION

DISCRETE event systems (DES) are dynamical systems which evolve in time by the occurrence of events at possibly irregular time intervals. Some examples include flexible manufacturing systems, computer networks, logic circuits, and traffic systems. "Logical" DES are a class of discrete time DES with equations of motion that are most often nonlinear and discontinuous in the occurrence of the events. Recently, there has been much interest in studying the stability properties of logical DES and several definitions for stability, and methods for stability analysis have been proposed. Here we introduce a logical DES model and define *stability in the sense of Lyapunov* and *asymptotic stability* for logical DES. Then we show that the metric space formulation in [1] can be adapted so that a conventional analysis of stability which employs appropriate Lyapunov functions can be used for logical DES. An important advantage of the Lyapunov approach is that it does not require high computational complexity (as do some of the other new approaches), but the difficulty lies in specifying the Lyapunov function.

Manuscript received November 22, 1991; revised February 18, 1993. Paper recommended by Past Associate Editor, D. F. Delchamps. This work was supported in part by the National Science Foundation under Grant ECS88-02924 and Grant IRI-9210332, and in part by the Jet Propulsion Laboratory.

K. Passino is with the Department of Electrical Engineering, Ohio State University, Columbus, OH 43210-1272.

A. N. Michel and P. J. Antsaklis are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556.

IEEE Log Number 9213709.

We provide a general characterization of the stability properties of automata-theoretic DES models such as the "generator" in [2], General and Extended Petri nets [3], and finite state systems. The approach is further illustrated on a manufacturing system that processes batches of N different types of parts according to a priority scheme and a "load balancing problem" in computer networks. It is shown that the manufacturing system is stable is stable in the sense of Lyapunov. Certain "fairness" conditions (constraints allowing fair access to the machine) are provided to ensure that the manufacturing system is asymptotically stable in the large (which illustrates its ability to reorient itself to a safe operating condition). For the load balancing problem we examine both the "continuous" and "discrete" load cases. For each case we provide results on both Lyapunov and asymptotic stability in the large which illustrate the ability of the network to achieve load balancing (in the discrete load case only imperfect balancing can be achieved). This paper is an expanded version of [4], [5].

It has been long known (as shown in e.g., [1]) that a stability theory can be developed in a very broad setting (e.g., a metric space) which is phrased in terms of motions of dynamical systems and which does not require the description of the system under investigation in terms of specific equations (e.g., differential/difference equations, partial differential equations, etc.). Even though this theory is beautiful and powerful, it has thus far not found real-world applications *in its most general form.* We believe that the results in this paper on the use of Lyapunov theory for DES analysis constitute perhaps the *first* application of this general qualitative theory. Furthermore, we believe that the present results eliminate the need for ad hoc "stability definitions" made for specific applications as long as the DES under investigation can be described on a metric space. Thus, we demonstrate that it is possible to develop meaningful and useful qualitative results for DES which are phrased in terms of well-established and time-tested theories (e.g., Lyapunov and Lagrange stability theory).

In summary, some of the contributions of the present paper include the following:

1) perhaps the first application of the Lyanpunov theory in its most general form (developed, e.g., in [1]) to an interesting class of dynamical systems (DES);

2) demonstration that DES (that can be described on metric spaces) can often by analyzed by means of well-established and time-tested theories (Lyapunov theory) and that ad hoc, tailor made, "stability definitions" are often not needed (i.e., the wheel need not be reinvented);

0018-9286/94$04.00 © 1994 IEEE

3) general characterization and analysis of the stability properties of automta-theoretic models, Petri nets, and finite state systems; and

4) application of the results to a new manufacturing system example and an investigation into load balancing properties (as characterized by stability in the sense of Lyapunov and asymptotic stability) for both the continuous and discrete load cases.

The foundations for the study of stability properties of logical DES lie in the areas of general stability theory (the approach used herein) and theoretical computer science (recent DES-theoretic research). In the following paragraphs, we provide an overview of the research from these areas that has focused on the stability of DES and related studies of invariant sets in DES

The two (related) main areas in theoretical computer science that form the foundation for logical DES-theoretic stability studies are temporal logic and automata. Intuitively speaking, in a temporal logic or automata-theoretic framework, a system is considered in some sense *stable* if 1) for some set of initial states the system's state is guaranteed to enter a given set and stay there forever, or 2) for some set of initial states, the system's state is guaranteed to visit a given set of states infinitely often.

In temporal logic, stability characteristics are most often represented with temporal formulas from a linear or branching time language (modal logics) and either a proof system or an effective procedure is used to verify that the temporal formula is satisfied. The fact that the above notions of stability could be studied using temporal logic in a control-theoretic setting was first recognized in [6]. The linear time temporal logic framework of [7], which uses a proof system, is adapted and used to prove stability properties in a DES theoretic framework in [8]. A linear time temporal logic framework where effective procedures are used to mechanically test the satisfaction of formulas describing stability properties is studied in [9], [10]. Both a proof system and efficient algorithms for testing the satisfaction of "real time" temporal formulas are provided in [11]. The branching time temporal logic approach in [12] is adapted to a DES theoretic framework, and efficient algorithms are used to perform some studies of stability properties in [13].

Stability concepts for logical DES such as finite automata have foundations in the study of, for instance, Buchi and Muller automata [14], [15] and how infinite strings are accepted by such automata. This automata theoretic work in computer science has also been adapted for the study of stability of DES. In [16], the authors introduce a special DES model (finite automaton) and use a state-space approach to develop efficient algorithms for the study of the two types of stability described above. They also provide approaches to synthesize stabilizing controllers for DES and to study several other characteristics of logical DES (for more details see [17]). Related studies are given in [18] and [19]. The construction of stabilizing controllers has also been studied in a Petri net framework in [20]. Krogh's approach was based on the Ramadge–Wonham formulation [2]. Certainly, results in the Ramadge–Wonham framework can be utilized for the study of types of stability of logical DES.

Certain general formulations for the study of stability are relevant to the study of stability properties of logical DES. For instance, there have been studies of stability of asynchronous iterative processes in [21]. Tsitsiklis defines a model that can represent logical DES, and, assuming that the DES has certain timing characteristics, he gives constructive methods to study stability of a class of DES. Tsitsiklis identifies the relationship between his work and the use of Lyapunov functions and provides some efficient procedures for testing stability. For an introduction to general stability theory and an overview of such research, see [22]. Finally, in other DES studies, there have been significant advances recently in the study of stability properties of manufacturing systems in [23], [24].

In Section II, we introduce a logical DES model, and in Section III we define stability in the sense of Lyapunov and asymptotic stability for DES and give necessary and sufficient conditions for stability of invariant sets of DES in a metric space. In Section IV, we provide a characterization of the stability properties of systems represented by automata-theoretic models, Petri nets, and finite state models. The manufacturing system and computer network applications are also given in Section IV, and some concluding remarks are given in Section V.

II. A Discrete Event System Model

We will consider stability properties of discrete event systems that can be accurately modeled with

$$G = (\mathcal{X}, \mathcal{E}, f_e, g, \boldsymbol{E}_v) \tag{1}$$

where $\mathcal{X}$ is the set of states, $\mathcal{E}$ is the set of *events*,

$$f_e : \mathcal{X} \longrightarrow \mathcal{X} \tag{2}$$

for $e \in \mathcal{E}$ are operators,

$$g : \mathcal{X} \longrightarrow \mathcal{P}(\mathcal{E}) - \{\emptyset\} \tag{3}$$

is the *enable function*, and $\boldsymbol{E}_v \subset \mathcal{E}^{\blacktriangle}$ is the set of *valid event trajectories*. Here, for an arbitrary set Z, $\mathcal{P}(Z)$ denotes the power set of Z. We only require that $f_e(x)$ be defined when $e \in g(x)$. The inclusion of $\mathcal{P}(\mathcal{E}) - \{\emptyset\}$ in the codomain of g ensures that there will always exist some event that can occur. If, for some physical system, it is possible that at some state there are no events to occur, this can be modeled by appending a *null event* (when it occurs the state stays the same and time advances). In this way systems that can "deadlock" or "terminate" at a state can also be modeled via G and studied in the Lyapunov stability theoretic framework developed here.

We associate "time" indexes with the states and events so that $x_k \in \mathcal{X}$ represents the state at time $k \in \blacktriangle$ and $e_k \in \mathcal{E}$ represents an *enabled* event at time $k \in \blacktriangle$ if $e_k \in g(x_k)$. If at state $x_k \in \mathcal{X}$, event $e_k \in \mathcal{E}$ *occurs* at time $k \in \blacktriangle$ (randomly, not necessarily according to any particular statistics), then the next state x_{k+1} is given by application of the operator f_{e_k}, i.e., $x_{k+1} = f_{e_k}(x_k)$. Note that since $\boldsymbol{E}_v \subset \mathcal{E}^{\blacktriangle}$, if the system is at a state $x \in \mathcal{X}$ and events $g(x)$ are enabled, then eventually one of the events must occur. Events can only occur if they lie on valid event trajectories as we now discuss.

Any sequence $\{x_k\} \in \mathcal{X}^{\blacktriangle}$ such that for all k, $x_{k+1} = f_{e_k}(x_k)$ where $e_k \in g(x_k)$, is a *state trajectory.* The set of all *event trajectories* denoted with $\boldsymbol{E}$ ($\boldsymbol{E} \subset \mathcal{E}^{\blacktriangle}$) is composed of those sequences $\{e_k\} \in \mathcal{E}^{\blacktriangle}$ such that there exists a state trajectory $\{x_k\} \in \mathcal{X}^{\blacktriangle}$ where for all k, $e_k \in g(x_k)$. Hence, to each event trajectory, which specifies the order of the application of the operators f_e, there corresponds a unique state trajectory (but, in general, not vice versa). Define the set of *valid event trajectories* $\boldsymbol{E}_v$, so that $\boldsymbol{E}_v \subset \mathcal{E}^{\blacktriangle}$. The valid event trajectories represent the event trajectories that are *physically possible* in G. Hence, even if $x_k \in \mathcal{X}$ and $e_k \in g(x_k)$ it is not the case that e_k can occur unless it lies on valid event trajectory that ends at x_{k+1}, where $x_{k+1} = f_{e_k}(x_k)$. Hence, using G one normally first models the physical system via $\mathcal{X}$, $\mathcal{E}$, f_e, and g. Then $\boldsymbol{E}_v$ is added to indicate which trajectories are and are not possible in the physical system. When we study the applications we shall see that the use of $\boldsymbol{E}_v$ can facilitate the modeling of many DES and provide flexibility in the study of stability properties. The use of $\boldsymbol{E}_v$ also makes the model G much more flexible than a standard state machine in the sense that it effectively combines the so called "state-based models" with the "path models" of DES.

Let $\boldsymbol{E}_v(x_0) \subset \boldsymbol{E}_v$ denote the set of all possible valid event trajectories that begin from state $x_0 \in \mathcal{X}$. Below, we shall also utilize a special set of *allowed event trajectories* denoted with $\boldsymbol{E}_a$, where $\boldsymbol{E}_a \subset \boldsymbol{E}_v$, and allowed event trajectories that begin at state $x_0 \in \mathcal{X}$ denoted by $\boldsymbol{E}_a(x_0)$. Note that since $\boldsymbol{E}_v(x_0) \subset \boldsymbol{E}_v \subset \boldsymbol{E} \subset \mathcal{E}^{\blacktriangle}$ all such event trajectories must be of infinite length. If one is concerned with the analysis of systems with finite length trajectories, this can be modeled with a null event as it is discussed above.

Let E_k, for fixed $k \in \blacktriangle$, denote an event sequence of k events that have occurred (by definition $E_0 = \varnothing$, the empty sequence). If $E_k = e_0, e_1, \cdots, e_{k-1}$ we let $E_k E \in \boldsymbol{E}_v(x_0)$ denote the *concatenation* of E_k and (the infinite sequence) $E = e_k, e_{k+1}, \cdots$, i.e., $E_k E = e_0, e_1, \cdots, e_{k-1}, e_k, e_{k+1}, \cdots$. The value of the function $X(x_0, E_k, k)$ will be used to denote the state reached at time k from $x_0 \in \mathcal{X}$ by application of event sequence E_k such that $E_k E \in \boldsymbol{E}_v(x_0)$. (By definition, $X(x_0, \varnothing, 0) = x_0$ for all $x_0 \in \mathcal{X}$.) For fixed x_0 and E_k, $X(x_0, E_k, k)$ shall be called a *motion* (which is a function of k). For our model G, we assume that for all $x_0 \in \mathcal{X}$, if $E_k E \in \boldsymbol{E}_v(x_0)$ and $E_{k'} E' \in \boldsymbol{E}_v(X(x_0, E_k, k))$ then $E_k E_{k'} E'' \in \boldsymbol{E}_v(x_0)$; consequently, for all $x_0 \in \mathcal{X}$,

$$X(X(x_0, E_k, k), E_{k'}, k') = X(x_0, E_k E_{k'}, k + k') \qquad (4)$$

for all $k, k' \in \blacktriangle$. This is the standard semigroup property for dynamical systems. (In Remark 2 it is explained how this assumption can be lifted and our results still hold.) This DES model provides a general enough framework to study the stability properties of automata-theoretic models, Petri nets, finite state systems, and a wide class of DES applications (see Section IV).

III. Necessary and Sufficient Conditions for the Stability of Invariant Sets of DES in a Metric Space

The following adapts the formulation developed in [1] to the study of stability properties of systems represented by the logical DES model introduced above. Note that stability of systems defined on normed linear spaces is treated in detail in [25]; however, this framework is inadequate due to the fact that the state spaces for the DES to be studied here cannot even be assumed to be *vector spaces* (e.g., for automata-theoretic models, Petri nets, and the applications in Section IV). Theorems 1 and 2 show that the stability framework in [1] can be extended to the case where for any state there can be an infinite number of possible next states (nondeterminism), and the case where local properties relative to event trajectories need to be studied.

Let $\rho : \mathcal{X} \times \mathcal{X} \to \gtreqless$ denote a *metric* on $\mathcal{X}$, and $\{\mathcal{X}; \rho\}$ a *metric space*. Let $\mathcal{X}_z \subset \mathcal{X}$ and $\rho(x, \mathcal{X}_z) = \inf\{\rho(x, x') : x' \in \mathcal{X}_z\}$ denote the *distance* from point x to the set $\mathcal{X}_z$. By a *functional* we shall mean a mapping from an arbitrary set to $\gtreqless$.

Definition 1: The r*-neighborhood* of an arbitrary set $\mathcal{X}_z \subset \mathcal{X}$ is denoted by the set $S(\mathcal{X}_z; r) = \{x \in \mathcal{X} : 0 < \rho(x, \mathcal{X}_z) < r\}$ where $r > 0$.

Definition 2: The set $\mathcal{X}_m \subset \mathcal{X}$ is called *invariant* with respect to (w.r.t) G if from $x_0 \in \mathcal{X}_m$ it follows that $X(x_0, E_k, k) \in \mathcal{X}_m$ for all E_k such that $E_k E \in \boldsymbol{E}_v(x_0)$ and $k \in \blacktriangle$ where E is an infinite event sequence.

Definition 3: A closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G is called *stable in the sense of Lyapunov w.r.t.* $\boldsymbol{E}_a$ if for any $\varepsilon > 0$ it is possible to find a quantity $\delta > 0$ such that when $\rho(x_0, \mathcal{X}_m) < \delta$ we have $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \varepsilon$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ and $k \in \blacktriangle$ where E is an infinite event sequence. If, furthermore, $\rho(X(x_0, E_k, k), \mathcal{X}_m) \to 0$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ as $k \to \infty$, then the closed invariant set $\mathcal{X}_m$ of G is called *asymptotically stable w.r.t.* $\boldsymbol{E}_a$.

Notice that the invariant set $\mathcal{X}_m$ is automatically closed (with respect to $\{\mathcal{X}; \rho\}$) due to the definition of invariance. As always these properties are *local* stability properties, i.e., with respect to some r-neighborhood. It follows directly from Definition 3 that if the closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G is stable in the sense of Lyapunov (asymptotically stable) w.r.t $\boldsymbol{E}_a$ then it is stable in the sense of Lyapunov (respectively, asymptotically stable) w.r.t all $\boldsymbol{E}'_a$ such that $\boldsymbol{E}'_a \subset \boldsymbol{E}_a$.

Definition 4: A closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G is called *unstable in the sense of Lyapunov w.r.t.* $\boldsymbol{E}_a$ if it is not stable in the sense of Lyapunov w.r.t $\boldsymbol{E}_a$.

Definition 5: If the closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G is asymptotically stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$, then the set $\mathcal{X}_a$ of all states $x_0 \in \mathcal{X}$ having the property $\rho(X(x_0, E_k, k), \mathcal{X}_m) \to 0$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ as $k \to \infty$ is called the *region of asymptotic stability of* $\mathcal{X}_m$ *w.r.t* $\boldsymbol{E}_a$.

Definition 6: The closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G with region of asymptotic stability $\mathcal{X}_a$ w.r.t. $\boldsymbol{E}_a$ is called *asymptotically stable in the large w.r.t.* $\boldsymbol{E}_a$ if $\mathcal{X}_a = \mathcal{X}$.

The above definitions provide a conventional characterization of stability for logical DES. Some more recent studies of various types of stability for logical DES are surveyed in the Introduction.

Remark 1: Let $\mathcal{X}_0$ denote a set of possible initial states and let $\mathcal{X}_m$ contain the elements of all the motions $X(x_0, E_k, k)$ such that $x_0 \in \mathcal{X}_0$ and E_k satisfies $E_kE \in \boldsymbol{E}_a(x_0)$ where E is an infinite event sequence. Studying the stability of this invariant set $\mathcal{X}_m$ is similar to the study of "orbital stability" in [25]. For this invariant set $\mathcal{X}_m$ it could also be assumed that each of these motions visits some prespecified set $\mathcal{X}_s \subset \mathcal{X}_m$ infinitely often or that the motions satisfy some other property. This shows one connection between the work in temporal logic and automata-theoretic studies [10], [16] and Lyapunov stability analysis.

The following theorems, which can be deduced from existing theory (e.g., in [1]), provide necessary and sufficient conditions for Lyapunov and asymptotic stability of the DES defined in (1).

Theorem 1: For a closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G to be stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$, it is necessary and sufficient that in a sufficiently small neighborhood $S(\mathcal{X}_m; r)$ of the set $\mathcal{X}_m$ there exists a specified functional V with the following properties:

i) For all sufficiently small $c_1 > 0$, it is possible to find a $c_2 > 0$ such that $V(x) > c_2$ for $x \in S(\mathcal{X}_m; r)$ and $\rho(x, \mathcal{X}_m) > c_1$.

ii) For any $c_4 > 0$ as small as desired, it is possible to find a $c_3 > 0$ so small that when $\rho(x, \mathcal{X}_m) < c_3$ for $x \in S(\mathcal{X}_m; r)$ we have $V(x) \le c_4$.

iii) $V(X(x_0, E_k, k))$ is a nonincreasing function for $k \in \blacktriangle$, for $x_0 \in S(\mathcal{X}_m; r)$, for all $k \in \blacktriangle$, as long as $X(x_0, E_k, k) \in S(\mathcal{X}_m; r)$ for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$.

Proof: (Necessity) Let the closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ be stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$ for some r-neighborhood of $\mathcal{X}_m$. We show that the conditions of Theorem 1 are satisfied. We choose a certain $\varepsilon > 0$. According to Definition 3 there corresponds to a certain $\delta > 0$ such that when $\rho(x_0, \mathcal{X}_m) < \delta$ we have $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \varepsilon$ for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$ and $k \in \blacktriangle$. Let

$$V(x_0) = \sup\Big\{\rho(X(x_0, E_k, k), \mathcal{X}_m) : \forall\, E_k, E_kE \in \boldsymbol{E}_a(x_0) \text{ and } k \in \blacktriangle\Big\} \quad (5)$$

This defines the functional $V(x_0)$ for $x_0 \in S(\mathcal{X}_m; \delta)$.

1) The functional $V(x_0)$ satisfies i) since $V(x_0) \ge \rho(x_0, \mathcal{X}_m)$, from which it follows that when $\rho(x_0, \mathcal{X}_m) > c_1$, $\rho(x_0, \mathcal{X}_m) < \delta$, and $c_1 = c_2$, we obtain $V(x_0) > c_2$.

2) For the $c_4 > 0$ one can find $c_3 > 0$ such that for $\rho(x_0, \mathcal{X}_m) < c_3$ we have $\rho(X(x_0, E_k, k), \mathcal{X}_m) < c_4$ for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$ and $k \in \blacktriangle$. Hence,

$$\sup\{\rho(X(x_0, E_k, k), \mathcal{X}_m) : \forall\, E_k, E_kE \in \boldsymbol{E}_a(x_0) \text{ and } k \in \blacktriangle\} \le c_4 \quad (6)$$

so $V(x_0) \le c_4$ for $\rho(x_0, \mathcal{X}_m) < c_3$; hence condition ii) is satisfied.

3) Let $x_0 \in S(\mathcal{X}_m; \delta)$, then for all $k \in \blacktriangle$ such that $k \in [0, T)$ (T is the time that the motion escapes the δ-neighborhood and it can be that $T = \infty$) and for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$ we have $X(x_0, E_k, k) \in S(\mathcal{X}_m; \delta)$. Consequently, the value of the functional is defined at any point $X(x_0, E_{k'}, k')$, where $k' \in \blacktriangle$ and $k' \in [0, T)$ for all $E_{k'}$ such that $E_{k'}E \in \boldsymbol{E}_a(x_0)$. Notice that

$$V(X(x_0, E_{k'}, k')) = \sup\{\rho(X(X(x_0, E_{k'}, k'), E_k, k), \mathcal{X}_m) : \forall\, E_k, E_kE \in \boldsymbol{E}_a(X(x_0, E_{k'}, k')), \forall\, k \in \blacktriangle\} \quad (7)$$

and from (4)

$$V(X(x_0, E_{k'}, k')) = \sup\{\rho(X(x_0, E_{k'}E_k, k' + k), \mathcal{X}_m) : \forall\, E_k, E_kE \in \boldsymbol{E}_a(X(x_0, E_{k'}, k')), \forall\, k \in \blacktriangle\}. \quad (8)$$

Notice that

$$V(X(x_0, E_{k'}, k')) \le \sup\{\rho(X(x_0, \overline{E}_k, k), \mathcal{X}_m) : \forall\, \overline{E}_k, \overline{E}_kE \in \boldsymbol{E}_a(x_0) \text{ and } k \in \blacktriangle\} = V(x_0) \quad (9)$$

Hence

$$V(X(x_0, E_{k'}, k')) \le V(x_0) \quad (10)$$

for $k' \in [0, T)$ so that $X(x_0, E_{k'}, k') \in S(\mathcal{X}_m; \delta)$. Hence, for the $\delta > 0$ that exists for every chosen $\varepsilon > 0$, V is a nonincreasing function of k on an r-neighborhood of $\mathcal{X}_m$.

(Sufficiency) Let there exist a specified functional V with properties i), ii), and iii) in a certain neighborhood $S(\mathcal{X}_m; r)$ (assume that $S(\mathcal{X}_m; r)$ is nonvoid, for if it is void then the result holds trivially). We now show that the closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$. Take $\varepsilon > 0$ and $\varepsilon < r$ and let

$$\lambda = \inf\{V(x) : x \in S(\mathcal{X}_m; r), \rho(x, \mathcal{X}_m) \ge \varepsilon\}$$

(Since $S(\mathcal{X}_m; r)$ is nonvoid, it can be assumed that ε is chosen so that $\{V(x) : x \in S(\mathcal{X}_m; r), \rho(x, \mathcal{X}_m) \ge \varepsilon\}$ is a nonvoid set so that λ is well defined.) By i) we have $\lambda > 0$. From ii) it is possible to find for λ, $\delta > 0$ such that for $\rho(x_0, \mathcal{X}_m) < \delta$, $V(x_0) < \lambda$ for $x_0 \in S(\mathcal{X}_m; r)$. We show that $\delta > 0$ thus found corresponds to the chosen $\varepsilon > 0$, i.e., when $\rho(x_0, \mathcal{X}_m) < \delta$ we get $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \varepsilon$ for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$ and $k \in \blacktriangle$. Assume the opposite, namely that there exists a point $x_0 \in S(\mathcal{X}_m; \delta)$ such that for a finite $k' > 0$ and $E_{k'}$ such that $E_{k'}E \in \boldsymbol{E}_a(x_0)$, the inequality $\rho(X(x_0, E_{k'}, k'), \mathcal{X}_m) \ge \varepsilon$ holds. We know that $\rho(X(x_0, E_{k'}, k'), \mathcal{X}_m) \le r$ by condition iii) so that V is defined at $X(x_0, E_{k'}, k')$ and by definition of λ, $V(X(x_0, E_{k'}, k')) \ge \lambda$. But by iii), $V(X(x_0, E_k, k)) \le V(x_0) < \lambda$ for all E_k such that $E_kE \in \boldsymbol{E}_a(x_0)$ and $k \in \blacktriangle$ which is a contradiction; hence the assumption is incorrect, and $\mathcal{X}_m$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$. ■

Theorem 2: For a closed invariant set $\mathcal{X}_m \subset \mathcal{X}$ of G to be asymptotically stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$, it is necessary and sufficient that in a sufficiently small neighborhood $S(\mathcal{X}_m; r)$, of the set $\mathcal{X}_m$ there exists a specified functional V having properties i), ii), and iii) of Theorem 1 and, furthermore, $V(X(x_0, E_k, k)) \to 0$ as $k \to \infty$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ and for all $k \in \blacktriangle$ as long as $X(x_0, E_k, k) \in S(\mathcal{X}_m; r)$.

Proof: (Necessity) Let $\mathcal{X}_m \subset \mathcal{X}$ be asymptotically stable w.r.t. $\boldsymbol{E}_a$. Then $\mathcal{X}_m$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$ and, consequently, in a sufficiently small neighborhood $S(\mathcal{X}_m; r)$, it is possible to construct a functional $V(x_0)$ (as in Theorem 1) which satisfies i), ii), and iii) of Theorem 1. By virtue of the asymptotic stability of $\mathcal{X}_m$ w.r.t. $\boldsymbol{E}_a$, all the motions $X(x_0, E_k, k)$ with E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ and $x_0 \in S(\mathcal{X}_m; \delta)$, remain in $S(\mathcal{X}_m; \delta)$ for all $k \in \blacktriangle$ for some $\delta > 0$. Let $X(x_0, E_k, k)$ be one of these motions. Let us show that $V(X(x_0, E_k, k)) \to 0$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ where $k \to \infty$. For $\varepsilon' > 0$ we can find $T > 0$ such that $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \varepsilon'$ and for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for $k \geq T$. The existence of such T follows from the asymptotic stability. It is clear that

$$V(X(x_0, E_k, k)) = \sup\{\rho(X(x_0, E_k E_{k'}, k + k'), \mathcal{X}_m) : \forall\, E_{k'}, E_{k'}E \in \boldsymbol{E}_a(X(x_0, E_k, k)), \forall\, E_{k'} \in \blacktriangle\}.$$

It follows from $\rho(X(x_0, E_{k+T}, k + T), \mathcal{X}_m) < \varepsilon'$ for $k \in \blacktriangle$ that $V(X(x_0, E_k, k)) \leq \varepsilon'$ for $k \geq T$; consequently, $V(X(x_0, E_k, k)) \to 0$ as $k \to +\infty$.

(Sufficiency) Let the conditions of Theorem 2 be satisfied. Let us prove that the invariant set $\mathcal{X}_m$ is asymptotically stable w.r.t. $\boldsymbol{E}_a$. From the satisfaction of the conditions of Theorem 2, it follows that in the neighborhood $S(\mathcal{X}_m; r)$ there exists $V(x_0)$ satisfying conditions i), ii), and iii) of Theorem 1. Consequently, the set $\mathcal{X}_m$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$, i.e., for any $\varepsilon > 0$ it is possible to find a $\delta > 0$ such that when $\rho(x_0, \mathcal{X}_m) < \delta$, we have $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \varepsilon$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for all $k \in \blacktriangle$. Let us show that this δ can at the same time be chosen so as to make $\rho(X(x_0, E_k, k), \mathcal{X}_m) \to 0$ as $k \to +\infty$ and for $\rho(x_0, \mathcal{X}_m) < \delta$. In fact, for the value of $\delta > 0$ obtained, we construct by means of the process indicated in the proof of Theorem 1 (as for ε) a δ_1 such that when $\rho(x_0, \mathcal{X}_m) < \delta_1$, we have $\rho(X(x_0, E_k, k), \mathcal{X}_m) < \delta$ for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for all $k \in \blacktriangle$. It is clear that $V(X(x_0, E_k, k))$ is defined for $k \in \blacktriangle$ and for all E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for any $x_0 \in S(\mathcal{X}_m; \delta_1)$. Let us show that δ_1 is the one sought. We assume that this is not so, i.e., that there exists at least one point $x_0 \in S(\mathcal{X}_m; \delta_1)$ such that $\rho(X(x_0, E_k, k), \mathcal{X}_m) > c_1 > 0$ for some $c_1 > 0$ for some E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for infinitely many $k \in \blacktriangle$. We then have $V(X(x_0, E_k, k)) > c_2 > 0$ in accordance with property i) for some $c_2 > 0$ for this E_k such that $E_k E \in \boldsymbol{E}_a(x_0)$ for infinitely many $k \in \blacktriangle$ which contradicts the condition $V(X(x_0, E_k, k)) \to 0$ as $k \to +\infty$. ∎

Remark 2: Although Theorems 1 and 2 rely on assuming that the semigroup property (4) holds, it is possible to prove exactly the same results without this assumption. The basic approach follows along the same lines as the above proofs and is based on the results in [1, ch. 4].

IV. Discrete Event System Applications

In this section, we explain the relevance of the Lyapunov framework to automata, Petri nets, and finite state systems. Then we show how to perform conventional Lyapunov stability analysis for two types of DES applications: 1) a manufacturing system that processes batches of N different types of parts according to a priority scheme, and 2) a load balancing problem in computer networks. In each case we specify the logical DES model G and the invariant set $\mathcal{X}_m$, pick the metric ρ, choose the Lyapunov function $V(x)$, then show that $V(x)$ satisfies the appropriate properties. Detailed comparisons to similar applications found in the literature are given throughout.

A. Automata, Petri Nets, and Finite State Systems

In this section, we show how the results of Section III can be used to characterize and analyze the stability properties of systems represented by automata-theoretic models like the "generator" in [2], General and Extended Petri nets [3], and finite state systems. This analysis helps to show 1) the relevance of Lyapunov stability to general logical DES models, and 2) some limitations of the proposed stability analysis approach.

Assume that we have a DES model $G_{\underline{\text{aut}}} = (Q, \Sigma, \delta, \boldsymbol{E})$ where Q is the set of states, Σ is the set of events, $\delta : \Sigma \times Q \to Q$ is the state transition function, and we allow all event trajectories (denoted by $\boldsymbol{E}$) to occur. We emphasize that for $G_{\underline{\text{aut}}}$ we focus on general logical DES models where the state and event sets Q and Σ are nonnumeric, i.e., "symbolic," and there are no particular assumptions about δ. In this general case, even though the "state space" of $G_{\underline{\text{aut}}}$ is completely unstructured, one can still metricize Q with the *discrete metric* ρ_d ($\rho_d(q, q') = 0$ if $q = q'$, and $\rho_d(q, q') = 1$ if $q = q'$). Relative to the metric space $\{Q; \rho_d\}$ any closed invariant set $Q_m \subset Q$ for $G_{\underline{\text{aut}}}$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}$ and asymptotically stable w.r.t. $\boldsymbol{E}$. This is the case since there are *local* properties. For asymptotic stability in the large w.r.t. $\boldsymbol{E}$, we can let $V(q) = \rho_d(q, Q_m)$. Proving that $\rho_d(q_k, Q_m) \to 0$ as $k \to \infty$ for all possible initial states and event trajectories involves showing that for all possible event trajectories and initial states there exists $k' > 0$ such that $\rho_d(q_{k'}, Q_m) = 0$. Hence the Lyapunov framework (for a metric space) offers little in the way of analysis in such general cases (the analysis reduces to the study of invariant sets).

Any system that can be represented with the General and Extended Petri nets [3] can also be represented with our DES model (1). For the Petri net $\mathcal{X} = \blacktriangle^n$ and if $x = [x_1 \cdots x_n]^t$ and $x' = [x_1' \cdots x_n']^t$ then $\rho_1(x, x') = \sum_{i=1}^n |x_i - x_i'|$ is a valid choice for a metric. While any invariant set $\mathcal{X}_m \subset \mathcal{X}$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}$ and asymptotically stable w.r.t. $\boldsymbol{E}$ (relative to the metric space $\{\mathcal{X}; \rho_1\}$), the use of $V = \rho_1$ can sometimes be useful in the analysis of asymptotic stability in the large w.r.t. $\boldsymbol{E}$ (see the results and Petri net

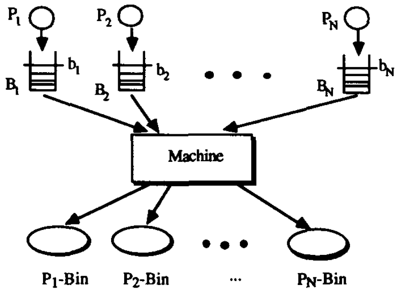


Fig. 1. Manufacturing system with priority batch processing.

applications in [26]). For finite state systems defined on a metric space, it is the case that for all $x, x' \in \mathcal{X}$ there exists $\gamma > 0$ such that $\rho(x, x') > \gamma$. Hence, all G such that $|\mathcal{X}|$ is finite are stable in the sense of Lyapunov and asymptotically stable as in the automata model case. As for the Petri net case, the analysis of asymptotic stability in the large can, in some cases, be facilitated with the Lyapunov framework. For example, in [5] the authors use the Lyapunov framework of Section III to analyze asymptotic stability in the large for Dijkstra's self-stabilizing distributed system [27, 28] that has been studied via a temporal logic framework in [8].

B. Manufacturing System

Consider the manufacturing system shown in Fig. 1 that processes batches of N different types of jobs according to a priority scheme. Here we use the term "job" in a general sense. For us, the completion of a job may mean the processing of a batch of 10 *parts*, the processing of a batch of 5.103 *tasks*, etc. There are N *producers* P_i, where $1 \le i \le N$, of jobs of different types. The producers P_i place batches of their jobs in their respective *buffers* B_i, where $1 \le i \le N$. These buffers B_i have *safe capacity limits* of b_i where $b_i > 0$, $1 \le i \le N$. Let x_i, $1 \le i \le N$, denote the number of jobs in buffer B_i. Let x_i for $N+1 \le i \le 2N$ denote the number of P_{i-N} type jobs in the machine. The machine can safely process less than or equal to M (where $M > 0$) jobs of any type, at any time. As the machine finishes processing batches of P_i type jobs they are placed in their respective *output bins* (P_i-bins). The producers P_i can only place batches of jobs in their buffers B_i if $x_i < b_i$. Also, there is a priority scheme whereby batches of P_i type jobs are only allowed to enter the machine if $x_j = 0$ for all j such that $j < i \le N$, i.e., only if there are no jobs in any buffers to the left of the B_i buffer. Next, we specify the DES model G for the manufacturing system.

Let $\mathcal{X} = \gtreqless^{2N}$ and $\boldsymbol{x}_k \in \mathcal{X}$, where $\boldsymbol{x}_k = [x_1 x_2 \cdots x_N x_{N+1} x_{N+2} \cdots x_{2N}]^t$ (t denotes transpose) denote the state at time k. Let the set of events $\mathcal{E}$ be composed of events e_{pi} for $1 \le i \le N$ (representing the case where producer P_i places a batch of α_{pi} jobs in buffer B_i), events e_{ai} for $1 \le i \le N$ (representing the case where a batch of α_{ai} P_i jobs, from buffer B_i, arrive at the machine for processing), and events e_{di} for $1 \le i \le N$ (representing the case where a batch of α_{ai} P_i jobs depart from the machine after they are processed and are placed in their respective output bins). When we say a e_{pi}, e_{ai}, e_{di} "type" of event we mean an event e_{pi}, e_{ai}, and e_{di} for any α_{pi}, α_{ai}, and α_{di}, respectively. It is assumed that jobs are *infinitely divisible* so that, for example, a batch of 5.23 jobs can be placed into buffer B_i, 2.01 of these jobs can be placed into the machine for processing, then 1.999 of these can be processed. Note, however, that results similar to those below also hold for *discrete jobs* as it was shown in [4], [5]. Let $\gtreqless_+$ denote the set of nonnegative reals and $\gtreqless^+ = \gtreqless_+ \cup \{0\}$. Let $\gamma \in (0, 1]$ denote a fixed parameter. According to the above specifications, the enable function g and event operators f_e for $e \in g(\boldsymbol{x}_k)$ are defined below.

i) If $x_i < b_i$ for some i, $1 \le i \le N$, then $e_{pi} \in g(\boldsymbol{x}_k)$ and $f_{e_{pi}}(\boldsymbol{x}_k) = [x_1 x_2 \cdots x_i + \alpha_{pi} \cdots x_N x_{N+1} x_{N+2} \cdots x_{2N}]^t$, where $\alpha_{pi} \in \gtreqless^+$, $\alpha_{pi} \le |x_i - b_i|$.

ii) If $\sum_{j=N+1}^{2N} < M$, and for some i, $1 \le i \le N$, $x_i > 0$, and $x_l = 0$ for all l, $l \le i \le N$, then $e_{ai} \in g(\boldsymbol{x}_k)$ and $f_{e_{ai}}(\boldsymbol{x}_k) = [x_1 x_2 \cdots x_i - \alpha_{ai} \cdots x_N x_{N+1} x_{N+2} \cdots x_{N+i} + \alpha_{ai} \cdots x_{2N}]^t$, where $\gamma x_i \le \alpha_{ai} \le \min\{x_i | \sum_{j=N+1}^{2N} x_j - M|\}$.

iii) If $x_i > 0$ for any i, $1 \le i \le N$, then $e_{di} \in g(\boldsymbol{x}_k)$ and $f_{e_{di}}(\boldsymbol{x}_k) = [x_1 x_2 \cdots x_N x_{N+1} x_{N+2} \cdots x_{N+i} - \alpha_{di} \cdots x_{2N}]^t$, where $\gamma x_{N+i} \le \alpha_{di} \le x_{N+i}$.

For i) each time an event e_{pi} occurs, some amount of jobs arrive at the buffers but the producers will never overfill the buffers. For ii), the e_{ai} are enabled only if the machine is not too full and the ith buffer has appropriate priority. The number of jobs that can arrive at the machine is limited by the number available in the buffers and by how many the machine can process at once. We require that $\gamma x_i \le \alpha_{ai}$ so that nonneglible batches of jobs arrive when they are allowed to. The constraints on α_{di} in iii) ensure that the number of jobs that can depart the machine is limited by the number of jobs in the machine and that nonegligible amounts of jobs depart from the machine. We let $\boldsymbol{E}_v = \boldsymbol{E}$, i.e., the set of all event trajectories is defined by g and f_e for $e \in g(\boldsymbol{x}_k)$. The system operates in a standard asynchronous fashion.

This manufacturing system is a generalization of computer systems often used in the study of a simple "mutual exclusion problem" in computer science [3], [7] and similar to several applications studied in the DES literature. For instance, if $\boldsymbol{x}_0 = 0$ and $\alpha_{pi} = \alpha_{ai} = \alpha_{di} = 1$ for all i, $1 \le i \le N$, for all times then our manufacturing system is similar to the "Two Class Parts Processing" example in [8] (except they allow an arbitrary finite number of parts to enter their machine and consider only two producers), and the manufacturing system example in [9], [10] (they also consider only two producers).

Let

$$\mathcal{X}_m = \left\{ \boldsymbol{x} \in \mathcal{X} : x_i \le b_i \ \forall\, i, 1 \le i \le N, \text{ and } \sum_{j=N+1}^{2N} x_j \le M \right\} \tag{11}$$

which represents all states for which the manufacturing system is in a safe operating mode. It is easy to see that $\mathcal{X}_m$ is invariant by letting $\boldsymbol{x}_k \in \mathcal{X}_m$ and showing that no matter which event occurs it is the case that the next state $\boldsymbol{x}_{k+1} \in \mathcal{X}_m$. The invariance of $\mathcal{X}_m$ is the property of the manufacturing system that has been studied extensively in similar manufacturing system examples [8]–[10]. Also, if $M = 1$, $N = 2$, $\boldsymbol{x}_0 = 0$, $\alpha_{pi} = \alpha_{ai} = \alpha_{di} = 1$ for all i, $1 \le i \le N$, for all times, and the priority scheme is removed, then the proof of the invariance of $\mathcal{X}_m$ is equivalent to proving the mutual exclusion property often studied in the computer science mentioned above.

Here, we provide a new study of the *stability properties* of the above manufacturing system. Intuitively this will, for instance, show that under certain conditions, if the manufacturing system starts in an unsafe operating mode (too many jobs in a buffer or in the machine, or both), it will eventually return to a safe operating condition. This is more carefully quantified in the following propositions and their proofs. Let $\boldsymbol{x}_k = [x_1 \cdots x_{2N}]^t$, $\boldsymbol{x}_{k+1} = [x_1' \cdots x_{2N}']^t$, $\overline{\boldsymbol{x}} = [\overline{x}_1 \cdots \overline{x}_{2N}]^t$, and $\overline{\boldsymbol{x}}' = [\overline{x}_1' \cdots \overline{x}_{2N}']^t$ (we often omit the "k"). For this manufacturing system example we assume that

$$\rho(\boldsymbol{x}, \mathcal{X}_m) = \inf\left\{ \sum_{j=1}^{2N} |x_j - \overline{x}_j| : \overline{\boldsymbol{x}} \in \mathcal{X}_m \right\}.$$

Proposition 1: For the manufacturing system, the closed invariant set $\mathcal{X}_m$ is stable in the sense of Lyapunov w.r.t $\boldsymbol{E}_v$.

Proof: Choose $V_1(\boldsymbol{x}_k) = \rho(\boldsymbol{x}_k, \mathcal{X}_m)$. We will show that $V_1(\boldsymbol{x}_k)$ satisfies conditions i), ii), and iii) of Theorem 1 for all $\boldsymbol{x}_k \notin \mathcal{X}_m$. Conditions i) and ii) follow directly from the choice of $V_1(\boldsymbol{x}_k)$. For condition iii), we show that $V_1(\boldsymbol{x}_k) \ge V_1(\boldsymbol{x}_{k+1})$ for all $\boldsymbol{x}_k \notin \mathcal{X}_m$, no matter what event $e \in g(\boldsymbol{x}_k)$ occurs causing $\boldsymbol{x}_{k+1} = f_e(\boldsymbol{x}_k)$, as long as it lies on an event trajectory in $\boldsymbol{E}_v$.

a) For $\boldsymbol{x}_k \notin \mathcal{X}_m$ if e_{pi} occurs for some i, $1 \le i \le N$, then we need to show that

$$\inf\left\{ \sum_{j=1}^{2N} |x_j - \overline{x}_j| : \overline{\boldsymbol{x}} \in \mathcal{X}_m \right\} \ge \inf\left\{ \sum_{\substack{j=1\\ j\neq 1}}^{2N} |x_j - \overline{x}_j'| + |x_i + \alpha_{pi} - \overline{x}_i'| : \overline{\boldsymbol{x}}' \in \mathcal{X}_m \right\}. \quad (12)$$

It suffices to show that for all $\overline{\boldsymbol{x}} \in \mathcal{X}_m$ at which the inf is achieved on the left of (7), there exists $\overline{\boldsymbol{x}}' \in \mathcal{X}_m$ such that

$$\sum_{j=1}^{2N} |x_j - \overline{x}_j| \ge \sum_{\substack{j=1\\ j\neq 1}}^{2N} |x_j - \overline{x}_j'| + |x_i + \alpha_{pi} - \overline{x}_i'|. \quad (13)$$

If we choose $\overline{x}_l' = \overline{x}_l$ for all $l \ne i$ then it suffices to show that for all $\overline{x}_i$, $0 \le \overline{x}_i \le b_i$, at which the inf on the left side of (12) is achieved there exists $\overline{x}_i'$, $0 \le \overline{x}_i' \le b_i$, such that

$$|x_i - \overline{x}_i| \ge |x_i + \alpha_{pi} - \overline{x}_i'| \quad (14)$$

where $\alpha_{pi} \le |x_i - b_i|$. Choosing $\overline{x}_i' = x_i + \alpha_{pi}$ so that $0 \le \overline{x}_i' \le b_i$, results in $\overline{\boldsymbol{x}}' \in \mathcal{X}_m$ and the satisfaction of (14).

b) For $\boldsymbol{x}_k \notin \mathcal{X}_m$ if e_{ai} occurs for some i, $1 \le i \le N$, then following the above approach it suffices to show that for all $\overline{\boldsymbol{x}} \in \mathcal{X}_m$ at which the inf is achieved there exists $\overline{\boldsymbol{x}}' \in \mathcal{X}_m$ such that

$$\sum_{j=1}^{2N} |x_j - \overline{x}_j| \ge \sum_{\substack{j=1\\ j\neq 1, N+i}}^{2N} |x_j - \overline{x}_j'| + |x_i - \alpha_{ai} - \overline{x}_i'| + |x_{N+i} + \alpha_{ai} - \overline{x}_{N+i}'|. \quad (15)$$

Choosing $\overline{x}_l' = \overline{x}_l$ for all $l \ne i$, $N+i$ it suffices to show that for all $\overline{x}_i$, $\overline{x}_{N+i}$ there exists $\overline{x}_i'$, $\overline{x}_{N+i}'$ such that both

$$|x_i - \overline{x}_i| \ge |x_i - \alpha_{ai} - \overline{x}_i'| \quad (16)$$

and

$$|x_{N+i} - \overline{x}_{N+i}| \ge |x_{N+i} + \alpha_{ai} - \overline{x}_{N+i}'|. \quad (17)$$

For (16), if $x_i \le b_i$ then the inf is achieved so that $|x_i - \overline{x}_i| = |x_i - \alpha_{ai} - \overline{x}_i'| = 0$, whereas if $x_i > b_i$, the inf is achieved at $\overline{x}_i = b_i$ so clearly $|x_i - b_i| \ge |x_i - \alpha_{ai} - \overline{x}_i'|$ since either $\overline{x}_i' = b_i$ or $\overline{x}_i' = x_i - \alpha_{ai}$. The case for (17) is similar to case a) above. The case for e_{di} is similar to the case for (16). ■

Proposition 2: For the manufacturing system, the closed invariant set $\mathcal{X}_m$ is not asymptotically stable in the large w.r.t. $\boldsymbol{E}_v$.

Proof: We show that for some $\boldsymbol{x}_0 \notin \mathcal{X}_m$ there exists $E_kE \in \boldsymbol{E}_v$ such that it is not the case that $V_1(X(\boldsymbol{x}_0, E_k, k)) \to 0$ as $k \to +\infty$. In fact, we show two reasons why asymptotic stability is not achieved: 1) Consider the case where $x_i > b_i$ for all $1 \le i \le N$ (but where the machine is in a safe operating zone) and $E_kE = e_{a1}, e_{a1}, \cdots, e_{a1}, e_{p1}, e_{a1}, e_{d1}, e_{p1}, e_{a1}, e_{d1}, \cdots$. This allowable event trajectory represents the case where P_1 type jobs enter the machine for processing (and possibly are processed and output) until B_1 is well within in a safe operating zone ($x_1 < b_1$) then each time a P_1 job is produced and put in B_1, it is placed in the machine from B_1 and the machine processes and outputs it, P_1 puts another job in B_1 and repeats the process. For this $E_kE \in \boldsymbol{E}_v$, for all $k \in \blacktriangle$ there exists a $k' \ge k$ for which $X(\boldsymbol{x}_0, E_{k'}, k') \notin \mathcal{X}_m$. By the satisfaction of condition i) of Theorem 1, it is not the case that $V_1(\boldsymbol{x}_k) \to 0$ for the chosen $E_kE \in \boldsymbol{E}_v(\boldsymbol{x}_0)$. 2) Let $x_i > b_i$ for all i, $1 \le i \le N$. Assume that $x_{N+i} > 0$ for some i and that e_{di} occurs to process P_i type jobs and puts them into the P_i-bin. If for each successive time $\alpha_{di} = \gamma x_{N+i}$ it can be the case that $E = e_{di}e_{di}e_{di}\cdots$ (a constant string) where $E \in \boldsymbol{E}_v$. Hence the remainder of the events that occur are to reduce the number of P_i parts in the machine and no events occur to reduce the number of jobs in the buffers resulting in the lack of asymptotic stability in the large w.r.t. $\boldsymbol{E}_v$. ■

Notice that for the counterexamples to asymptotic stability provided in the proof of Proposition 2, case 1) essentially results from the priority ordering of the buffers and 2) results from the fact that jobs are infinitely divisible. Next, we provide an added assumption from which asymptotic stability in the large can be achieved. Let $\boldsymbol{E}_a \subset \boldsymbol{E}_v$ denote the set of event trajectories such that each type of event e_{pi}, e_{ai}, and e_{di},

$1 \le i \le N$, occurs *infinitely often* on each event trajectory $E \in \boldsymbol{E}_a$. If we assume for the manufacturing system that only events which lie on event trajectories in $\boldsymbol{E}_a$ occur, then it is always the case that eventually each type of event (e_{pi}, e_{ai}, and e_{di}, $1 \le i \le N$) will occur.

Proposition 3: For the manufacturing system, the closed invariant set $\mathcal{X}_m$ is asymptotically stable in the large w.r.t. $\boldsymbol{E}_a$ where $\boldsymbol{E}_a \subset \boldsymbol{E}_v$ as defined above.

Proof: By Proposition 1, $\mathcal{X}_m$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$. To show asymptotic stability we show that $V_1(\boldsymbol{x}_k) \to 0$ for all E_k such that $E_k E \in \boldsymbol{E}_a(\boldsymbol{x}_0)$ as $k \to +\infty$ for all $\boldsymbol{x}_k \notin \mathcal{X}_m$. Since $\alpha_{ai} \ge \gamma x_i$ and $\alpha_{di} \ge \gamma x_{N+i}$ where $\gamma \in (0,1]$ if e_{ai} and e_{di} where i, $1 \le i \le N$ occur infinitely often as the restrictions on $\boldsymbol{E}_a$ guarantee, x_i and x_{N+i} will converge so that $V_1(\boldsymbol{x}_k) \to 0$ as $k \to +\infty$ (of course it could be that $V(\boldsymbol{x}_k) = 0$ for some finite k). Hence, if the manufacturing system starts out in an unsafe operating mode, it will eventually enter a safe operating mode. ■

The use of the set $\boldsymbol{E}_a$ for the manufacturing system imposes what is called a "fairness' constraint in computer science (in our example we require that each producer P_i get fair use of the machine) [29]. One can guarantee that the fairness constraint can be met via the use of a mechanism for sequencing access to the machine. Such fairness constraints are also used in the study of temporal logic [7], [12], the mutual exclusion problem in computer science [28], and in [21] when the author studies conditions under which the Lyapunov function can be constructed mechanically for a class of logical DES.

C. Computer Network Load Balancing Problem

Consider a network of computers described by an directed graph (C, A) where $C = \{1, 2, \cdots, N\}$ represents a set of computers that are numbered with $i \in C$, and $A \subset C \times C$ is the set of connections between the computers. We require that if $i \in C$ then there exists $(i,j) \in A$ or $(j,i) \in A$ for some $j \in C$ (i.e., every computer is connected to the network). Also, if $(i,j) \in A$ then $(j,i) \in A$ and if $(i,j) \in A$ $i \neq j$. Each computer has a buffer which holds tasks (load), each of which can be executed by any computer in the network. Let the load of computer $i \in C$ be given by x_i; hence, $x_i \ge 0$. Each connection in the network $(i,j) \in A$ allows for computer i to pass a portion of its load to computer j. It also allows computer i to sense the size of the load of computer j (for any two computers i and j such that $(i,j) \notin A$, i may not pass load directly to j or sense the size of j's load).

We assume that initially the distribution of the load across the computers is uneven and seek to prove properties relating to the system, achieving a more even distribution of tasks so that the computers in the network are more fully utilized. For convenience, we assume that the computers will not begin working on any of the tasks or receive any more to process until the load has been balanced. (Under certain conditions this assumption can be lifted, and our analysis still applies as we discuss below in Remark 4.)

Below we will consider two different cases: 1) continuous load: when the load is infinitely divisible (sometimes called "fluid load"), and 2) discrete load: when the load is in the form of fixed uniform-sized blocks that cannot be subdivided. The two cases are significantly different since, as it is explained below. In the discrete load case there are more severe restrictions on what can be passed so that it is only possible to achieve less than perfect balancing.

Continuous Load: First, we specify the model G. Let $\mathcal{X} = \gtreqless^N$ denote the set of states and $\boldsymbol{x}_k = [x_1 x_2 \cdots x_N]^t$ and $\boldsymbol{x}_{k+1} = [x_1' x_2' \cdots x_N']^t$ denote the state at time k and $k+1$, respectively. Let $e_{\alpha_k}^{ij}$ denote the event that represents the passing of α_k amount of load from computer i to computer j at time k (often we omit the subscript k). If the state is $\boldsymbol{x}_k$, then for some $(i,j) \in A$, $e_{\alpha_k}^{ij}$ occurs to produce the next state $\boldsymbol{x}_{k+1}$. Let $\mathcal{E} = \{e_\alpha^{ij} : (i,j) \in A, \alpha \in \gtreqless_+\}$ denote the (infinite) set of events (notice that all e_0^{ij} such that $(i,j) \in A$ are valid events). Below, when we say "an event of type e_α^{ij}" we mean any event e_α^{ij} (or e_α^{ji}) that represents the passing of load between i and j (i.e., for any $\alpha \ge 0$). For the specification of g and f_e for $e \in g(\boldsymbol{x}_k)$ let $\gamma \in (0, \frac{1}{2}]$:

a) If for any $(i,j) \in A$, $x_i > x_j$, then $e_\alpha^{ij} \in g(\boldsymbol{x}_k)$ and $f_e(\boldsymbol{x}_k) = \boldsymbol{x}_{k+1}$ where $e = e_\alpha^{ij}$, $x_i' := x_i - \alpha$, $x_j' := x_j + \alpha$, $x_k' := x_k$ for all $k \neq i,j$, and $\gamma|x_i - x_j| \le \alpha \le (1/2)|x_1 - x_j|$.

b) if for any $(i,j) \in A$, $x_i = x_j$ then $e_0^{ij} \in g(\boldsymbol{x}_k)$ and $f_e(\boldsymbol{x}_k) = \boldsymbol{x}_k$ where e_0^{ij}.

Let $\boldsymbol{E}_v = \boldsymbol{E}$ and $\mathcal{X}_c = \{\boldsymbol{x} \in \mathcal{X} : x_i = x_j \text{ for all } (i,j) \in A\}$ (representing perfect balancing) which is clearly invariant. Let $\boldsymbol{E}_a \subset \boldsymbol{E}_v$ denote the set of event trajectories such that events of each type e_α^{ij} occur infinitely often on each $E \in \boldsymbol{E}_a$. This fairness constraint is used to ensure that each pair of connected computers will continually try to balance the load between them.

This load balancing problem is similar to the one in [30] except the conditions for load passing here are different: at each time where load is passed from computer i to one of its neighbors j, such that $(i,j) \in A$, it is not required here to pass load to the lightest loaded neighbor. Also, as we shall see below, we guarantee that the load will eventually balance only under a fairness assumption given by $\boldsymbol{E}_a$ and not the "partial asynchronism assumption" in [30]. However, in [30], they allow for the possibility that a computer's information about the load of adjacent computers is outdated and when load is sent to a neighboring computer, there may be a delay in its arrival, and achieve geometric convergence with their partial asynchronism assumption when simultaneous load passing is possible. Various forms of the load balancing problem have also been studied in the DES literature [31] and extensively studied in the computer science literature (See [30]–[32] and the references therein).

The following Proposition and subsequent Remarks provide a new characterization and analysis of the Lyapunov and asymptotic stability of the computer network load balancing problem described above. Let $\overline{\boldsymbol{x}} = [\overline{x}_1 \cdots \overline{x}_N]^t$, $\overline{\boldsymbol{x}}' = [\overline{x}_1' \cdots \overline{x}_N']^t$, and choose

$$\rho(\boldsymbol{x}_k, \mathcal{X}_c) = \inf\{\max\{|x_1 - \overline{x}_1|, \cdots, |x_N - \overline{x}_N|\} : \overline{\boldsymbol{x}} \in \mathcal{X}_c\}.$$

Proposition 4: For the computer network load balancing problem with continuous load, the closed invariant set $\mathcal{X}_c$ is asymptotically stable in the large w.r.t. $\boldsymbol{E}_a$ where $\boldsymbol{E}_a \subset \boldsymbol{E}_v$ as defined above.

Proof: Choose

$$V_2(\boldsymbol{x}_k) = \rho(\boldsymbol{x}_k, \mathcal{X}_c) \tag{18}$$

so that conditions i) and ii) of Theorem 1 are satisfied. For condition iii) of Theorem 1, we must show that for all $\boldsymbol{x}_k \notin \mathcal{X}_c$ and all $e_\alpha^{ij} \in g(\boldsymbol{x}_k)$ when e_α^{ij} occurs $V_2(\boldsymbol{x}_k) \geq V_2(\boldsymbol{x}_{k+1})$, i.e., that

$$\begin{aligned}\inf\{\max\{|x_1-\overline{x}_1|,\cdots,|x_N-\overline{x}_N|\}:\overline{\boldsymbol{x}}\in\mathcal{X}_c\} \geq \\ \inf\{\max\{|x_1-\overline{x}_1'|,\cdots,|x_i-\alpha-\overline{x}_i'|, \\ \cdots,|x_j+\alpha-\overline{x}_j'|,\cdots,|x_N-\overline{x}_N'|:\overline{\boldsymbol{x}}'\in\mathcal{X}_c\}.\end{aligned} \tag{19}$$

Let $\mathcal{X}^* \subset \mathcal{X}_c$ denote the set of points at which the inf on the left of (19) is achieved. It suffices to show that for all $\overline{\boldsymbol{x}} \in \mathcal{X}^*$ there exists $\acute{\overline{\boldsymbol{x}}} \in \mathcal{X}_c$ such that

$$\begin{aligned}\max\{|x_1-\overline{x}_1|,\cdots,|x_N-\overline{x}_N|\} \geq \max\{|x_1-\overline{x}_1'|, \\ \cdots,|x_i-\alpha-\overline{x}_i'|, \\ \cdots,|x_j+\alpha-\overline{x}_j'|, \\ \cdots,|x_N-\overline{x}_N'|\}.\end{aligned} \tag{20}$$

Choose $\overline{x}_l' = \overline{x}_l$ for all $l \neq i, j$. It suffices to show that for all $\overline{\boldsymbol{x}} \in \mathcal{X}^*$ there exist $\overline{x}_i'$, $\overline{x}_j'$ such that

$$\begin{aligned}\max\{|x_i-\overline{x}_i|,|x_j-\overline{x}_j|\} > \max\{|x_i-\alpha-\overline{x}_i'|, \\ |x_j+\alpha-\overline{x}_j'|\}.\end{aligned} \tag{21}$$

For each $\overline{\boldsymbol{x}} \in \mathcal{X}^*$ there exist x^*, $x_* \in \gtreqless_+$ such that $\overline{x}_i = \overline{x}_j = x^*$ and $\overline{x}_i' = \overline{x}_j' = x_*$. Therefore, it suffices to show that for all x^* there exists x_* such that

$$\begin{aligned}\max\{|x_i-x^*|,|x_j-x^*|\} > \max\{|x_i-\alpha-x_*|, \\ |x_j+\alpha-x_*|\}.\end{aligned} \tag{22}$$

The validity of (22) is shown by considering all x_i, x_j such that $x_i > x_j$:

a) If $x_i \geq x^*$ and $x_j \geq x_*$ or $x_i \leq x^*$ and $x_j \leq x_*$ then choosing $x_* = x^*$ results in the satisfaction of (22).

b) If $x_i > x^*$ and $x_j < x_*$ then again choose $x_* = x^*$:

i) Since $2\alpha \leq |x_i - x_j|$, it is the case that $x_j + \alpha \leq x_i - \alpha$, so that $|x_i - \alpha - x_*| \geq |x_j + \alpha - x_*|$.

ii) Since $\alpha \geq \gamma|x_i - x_j|$, it is the case that $x_i > x_i - \alpha$, so that $|x_i - x^*| > |x_i - \alpha - x_*|$ resulting in the satisfaction of (22).

This completes the proof that $\mathcal{X}_c$ is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_a$. Next, we must show that $\mathcal{X}_c$ is asymptotically stable in the large w.r.t. $\boldsymbol{E}_a$. Notice that from the proof of (21) each time e_α^{ij} occurs $(\alpha > 0)$, $|x_m - \overline{x}_m| > |x_m' - \overline{x}_m'|$ where $m = i$ or $m = j$ (or in both cases), and for $l \neq m$, $|x_l - \overline{x}_l| \geq |x_l' - \overline{x}_l'|$. Hence, each time e_α^{ij} occurs $(\alpha > 0)$, definite progress is made towards balancing the load between i and j. Due to the restrictions on $\boldsymbol{E}_a$, events of each type e_α^{ij} will be enabled and occur for all $k \geq 0$ so that from (21) the load that deviates most from balancing (as measured by V_2) must be reduced eventually. Hence, it must always be the case that there exists k such that for some $k' \geq k$, $V_2(\boldsymbol{x}_{k'}) > V_2(\boldsymbol{x}_{k'+1})$ as long as $\boldsymbol{x}_k \notin \mathcal{X}_c$ so $V_2(\boldsymbol{x}_k) \to 0$ as $k \to \infty$ for all E_k such that $E_k E \in \boldsymbol{E}_a$. Hence, the system is asymptotically stable in the large w.r.t. $\boldsymbol{E}_a$. ■

Proposition 5: For the computer network load balancing problem with continuous load $\mathcal{X}_c$

i) is stable in the sense of Lyapunov w.r.t. $\boldsymbol{E}_v$.

ii) is not asymptotically stable w.r.t. $\boldsymbol{E}_v$.

Proof: For i), notice that with $\boldsymbol{E}_v$, we are still guaranteed that $V_2(\boldsymbol{x}_k) \geq V_2(\boldsymbol{x}_{k+1})$ for all $k \geq 0$. For ii), without the fairness restrictions imposed by $\boldsymbol{E}_a$ some $(i,j) \in A$ may try to balance at each time instant so that no other load imbalances can be reduced. ■

Remark 3: If *simultaneous events* are allowed (i.e., i and j, $(i,j) \notin A$ can pass load at the same time instant), Proposition 4 is still valid and this can be shown using

$$V_2(\boldsymbol{x}) = \max_i\left\{\frac{1}{N}\sum_{j=1}^{N} x_j - x_i\right\} \tag{23}$$

as the Lyapunov function (of course appropriate events that represent the simultaneous occurrence of several of the above events must be defined) [33].

Discrete Load: In [4], [5], the authors study a load balancing problem where the load is discrete. In this case, it is assumed that any task can be executed on any computer, but that the tasks cannot be infinitely subdivided. The same graph (C, A) is used to describe the computer network. Discrete loads are quite common in computer networks since it is often the case that "jobs" in such networks can at most be broken down into *bits*, *bytes*, or some other finite block.

It is important to note that the discrete load case is not a special case of the continuous load case for the following reasons: 1) the fairness constraint imposed by $\boldsymbol{E}_a$ can be lifted, 2) for the continuous load case there are, in general, an uncountably infinite number of different events that can occur at each state where the load is not balanced whereas, in any state where the load is not balanced for the discrete load case, there are only a finite number of possible events that can occur, and 3) since the testing of whether or not the load is balanced can only be performed *locally*, and there may not be the proper number of load blocks to achieve perfect balancing, it is the case that only an imperfect type of balancing is possible in the discrete load case. Essentially, since the load is discrete, the system does not have as many ways to perform redistribution so that only imperfect load balancing can be achieved. The exact nature of this problem is more carefully quantified with the following model for the discrete load balancing problem and the subsequent stability analysis.

For the model for the discrete load case we use $G' = (\mathcal{X}', \mathcal{E}', f_e', g', \boldsymbol{E}_v')$ where $\mathcal{X}' = \blacktriangle^N$ and $\mathcal{E}' = \{e_\alpha^{ij} : (i,j) \in A, \alpha \in \blacktriangle - \{0\}\} \cup \{e^0\}$ is the set of events for G' where

e_α^{ij} is defined similar to above (including "types" of e_α^{ij}) and e^0 is a null event. Let $M \in \blacktriangle - \{0\}$ be the amount of load imbalance tolerated between any two computers i and j where $(i,j) \in A$. Next we specify g' and f_e' for $e \in g'(\boldsymbol{x}_k)$:

i) If for any $(i,j) \in A$, $|x_i - x_j| > M$ then if $x_i > x_j$, then $e_\alpha^{ij} \in g'(\boldsymbol{x}_k)$ and $f_e'(\boldsymbol{x}_k) = \boldsymbol{x}_{k+1}$ where $e = e_\alpha^{ij}$, $x_i' := x_i - \alpha$, $x_j' := x_j + \alpha$, $x_k' := x_k$ for all $k \neq i,j$, and $0 < \alpha \leq (1/2)(x_i - x_j)$ for $\alpha \in \blacktriangle$.

ii) If for all $(i,j) \in A$, $|x_i - x_j| \leq M$, then $e^0 \in g(\boldsymbol{x}_k)$ and $f_e(\boldsymbol{x}_k) = \boldsymbol{x}_k$ where $e = e^0$.

Let $\boldsymbol{E}_v' = \boldsymbol{E}'$ and $\mathcal{X}_d = \{\boldsymbol{x} \in \mathcal{X}' : |x_i - x_j| \leq M \text{ for all } (i,j) \in A\}$ which is clearly invariant and which represents less than perfect balancing.

Note that as in Section IV.A in the study of automata, Petri nets, and finite state systems, for the discrete load balancing problem $\mathcal{X}_d$ is trivially stable in the sense of Lyapunov and asymptotically stable w.r.t. $\boldsymbol{E}_v'$; but these are only local properties. The following result shows the utility of the Lyapunov approach for DES stability analysis for systems with a discrete metric space by studying asymptotic stability in the large, i.e., a *nonlocal* stability property.

Proposition 6: For the computer network load balancing problem with discrete load, the closed invariant set $\mathcal{X}_d$ is asymptotically stable in the large w.r.t. $\boldsymbol{E}_v'$.

Proof: For stability in the sense of Lyapunov, the proof is similar to that of Proposition 4 except that an extended case analysis is needed to show the validity of (21). We omit the proof in the interest of saving space. The same metric and Lyapunov function can be used and the details are given in [5]. Next, we must show that $\mathcal{X}_d$ is asymptotically stable in the large w.r.t. $\boldsymbol{E}_v'$. The proof is similar to that for Proposition 4 but now we are always guaranteed that the lightest loaded computer will receive more load to process in a finite amount of time until the load is balanced. ■

Proposition 6 shows that the use of discrete load restricts the passing of load (there are fewer enabled events at each state) so that, in general, less than perfect balancing can be achieved. It is important to note that the necessary use of M to quantify the tolerable imbalance between i and j, $(i,j) \in A$ can propagate through a large network (C, A). Hence, when the load is balanced in the discrete load case there may be a large difference between the loads in two unconnected computers (e.g., each successive set of arcs in a path in (C, A) can allow for another M amount of imbalance). Also note that due to the discrete load assumption no restrictions are needed on $\boldsymbol{E}_v'$ (as for the continuous load case for $\boldsymbol{E}_a$) to ensure that asymptotic stability in the large is achieved.

Remark 4: If, for either the discrete or continuous load cases, tasks enter the computer network or get processed by one of the computers $i \in C$ we let a new initial state $\boldsymbol{x}_0$ reflect the increased or decreased load, and the above stability analysis shows that the load will still eventually balance provided that new tasks arrive and tasks depart sufficiently slower than the load is balanced. (This characteristic was also discussed in [30].) In fact, for the discrete load case, if the total amount of load is finite then it will take a finite amount of time for the load to become balanced.

V. Conclusions

It has been shown that it is possible to define and study Lyapunov stability of a wide class of logical DES by adapting the metric space formulation in [1]. Hence, logical DES, which have recently received much attention in the literature, are amenable to conventional stability analysis via the choosing of appropriate Lyapunov functions. Other notions of stability and more recent stability analysis techniques based on methods from theoretical computer science (surveyed in the Introduction) are often prohibitive due to problems with computational complexity. Here, we avoid these problems with computational complexity but instead rely on the specification of Lyapunov functions that satisfy certain properties. We have provided a general characterization of the stability properties of automata-theoretic models such as the "generator" in [2], General and Extended Petri nets, and finite state systems. Furthermore, we have shown that it is not difficult to specify Lyapunov functions for two types of DES applications: a manufacturing system that processes batches of N different types of parts according to a priority scheme and a load balancing problem in computer networks. Our characterization and analysis of stability of DES in a traditional stability-theoretic framework will, in the future, allow researchers to use the vast body of concepts from the field of Lyapunov stability theory to study properties of DES.

References

[1] V. I. Zubov, *Methods of A.M. Lyapunov and Their Applications.* The Netherlands: Noordhoff, 1964.

[2] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM J. Contr. Optimiz.*, vol. 25, no. 1, pp. 206–230, Jan. 1987.

[3] T. Murata, "Petri nets: Properties, analysis, and applications," in *Proc. IEEE*, vol. 77, no. 4, pp. 541–580, Apr. 1989.

[4] K. M. Passino, A. N. Michel, and P. J. Antsaklis, "Stability analysis of discrete event systems," in *Proc. 28th Allerton Conf. Commun., Contr., Comput.*, Univ. Illinois, Urbana-Champaign, IL, 1990, pp. 487–496.

[5] ——, "Lyapunov stability of a class of discrete event systems," in *Proc. Amer. Contr. Conf.*, Boston, MA, 1991, pp. 2911–2916.

[6] A. Fusaoka, H. Seki, and K. Takahashi, "A description and reasoning of plant controllers in temporal logic," in *Proc. 8th Int. Joint Conf. Artificial Intelligence*, Aug. 1983, pp. 405–408.

[7] Z. Manna and A. Pnueli, "Verification of concurrent programs: A temporal proof system," Dept. Comput. Sci., Stanford Univ., Rep. no. STAN-CS-83-967, 1983.

[8] J. G. Thistle and W. M. Wonham, "Control problems in a temporal logic framework," *Int. J. Contr.*, vol. 44, no. 4, pp. 943–976, 1986.

[9] J. F. Knight and K. M. Passino, "Decidability for temporal logic in control theory," in *Proc. Allerton Conf. Commun., Contr., Comput.*, Univ. Illinois, Urbana-Champaign, IL, 1987, pp. 335–344.

[10] ——, "Decidability for a temporal logic used in discrete event system analysis," *Int. J. Contr.*, vol. 52, no. 6, pp. 1489–1506, 1990.

[11] J. S. Ostroff, "Real-time computer control of discrete systems modelled by extended state machines: a temporal logic approach," Ph.D. dissertation, Dept. of Elect. Eng., Univ. of Toronto, Canada, Rep. No. 8618, 1987.

[12] E. M. Clarke, M. C. Browne, E. A. Emerson, and A. P. Sistla, "Using temporal logic for automatic verification of finite state systems," in *Logics and Models of Concurrent Systems*, K. R. Apt, Ed. NY: Springer Verlag, 1985, pp. 3–25.

[13] K. M. Passino and P. J. Antsaklis, "Branching time temporal logic for discrete event system analysis," in *Proc. 26th Allerton Conf. Commun., Cont., Comput.*, Univ. Illinois, Urbana-Champaign, IL, 1988, pp. 1160–1169.

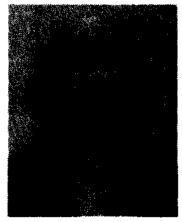
[14] J. R. Buchi, "On a decision method in restricted second order arithmetic," in *Proc. Int. Congress on Logic, mathematics, and Phil. Sci., 1960.* Stanford, CA: Stanford Univ. Press, 1962, pp. 1–11.

[15] D.E. Muller, "Infinite sequences and finite machines," in *Switching Circuit Theory and Logic Design,* in *Proc. Fourth Annual Symp., IEEE,* Chicago, IL, 1963, pp. 3–16.
[16] C.M. Ozveren, A.S. Willsky, and P.J. Antsaklis, "Stability and Stabilizability of Discrete Event Dynamic Systems," MIT, Cambridge, MA, LIDS Rep. LIDS-P-1853, Feb. 1989; and *J. ACM,* to be published.
[17] C.M. Ozveren "Anaysis and control of discrete event dynamic systems: A state space approach," Ph. D. dissertation, MIT, Cambridge, MA, LIDS Rep. LIDS-TH-1907, Aug. 1989.
[18] Y. Brave and M. Heymann, "On stabilization of discrete event processes," in *Proc. 28th Conf. Decision and Contr.,* Tampa, FL, 1989, pp. 2737–2742.
[19] P.J. Ramadge, "Some tractable supervisory control problems for discrete-event systems modeled by Buchi automata," *IEEE Trans. Automat. Contr.,* vol. 34, no. 1, pp. 10–19, 1989.
[20] B.H. Krogh, "Controlled petri nets and maximally permissive feedback logic," in *Proc. 25th Allerton Conf.,* Univ. Illinois, Urbana-Champaign, IL, 1987, pp. 317–326.
[21] J.N. Tsitsiklis, "On the stability of asychronous iterative processes," *Math. Sys. Theory,* vol. 20, pp. 137–153, 1987.
[22] A.N. Michel and R.K. Miller, *Qualitative Analysis of Large Scale Dynamical Systems.* New York: Academic, 1977.
[23] J.R. Perkins and P.R. Kumar, "Stable, distributed, real-time scheduling of flexible manufacturing/assembly/disassembly systems," *IEEE Trans. Automat. Contr.,* vol. 34, no. 2, pp. 139–148, Feb. 1989.
[24] P.R. Kumar and T.I. Seidman, "Dynamic instabilities and stabilization methods in distributed real-time scheduling of manufacturing systems," *IEEE Trans. Automat. Contr.,* vol. 35, no. 3, pp. 289–298, Mar. 1990.
[25] W. Hahn, *Stability of Motion.* New York: Springer Verlag, 1967.
[26] K.M. Passino and A.N. Michel, "Stability and boundedness analysis of discrete event systems," in *Proc. Amer. Contr. Conf.,* Chicago, IL, 1992, pp. 3201–3205.
[27] E.W. Dijkstra, "Self-stabilizing systems in spite of distributed control," *Commun. The ACM,* vol. 17, no. 11, pp. 643–644, Nov. 1974.
[28] M. Raynal, *Algorithms for Mutual Exclusion.* Cambridge, MA: MIT Press, 1986.
[29] N. Francez, *Fairness.* New York: Springer-Verlag, 1986.
[30] D.P. Bertsekas and J.N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall, 1989.
[31] R.K. Boel and J.H. van Schuppen, "Distributed routing for load balancing," *Proc. IEEE,* vol. 77, no. 1, pp. 210–221, 1989.
[32] G. Cybenko, "Dynamic load balancing for distributed memory multiprocessors," Tufts Univ., Medford, MA, Dept. of Comput. Sci. Tech. Rep. 87-1, 1987; and *J. Parallel and Distributed Comput.,* to be published.
[33] K.L. Burgess and K.M. Passino, "Stability analysis of load balancing systems," in *Proc. of the Amer. Contr. Conf.,* San Francisco, CA, June 1992, pp. 2415–2419.

Panos J. Antsaklis (S'74–M'76–SM'86–F'91) received the Diploma in mechanical and electrical engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1972 and the M.S. and Ph.D. degrees in electrical engineering from Brown University, Providence, RI, in 1974 and 1977, respectively.

He is currently Professor of Electrical Engineering at the University of Notre Dame, Notre Dame, IN. He has held faculty positions at Brown University, Rice University, and Imperial College, University of London. During sabbatical leaves, he has been Senior Visiting Scientist at LIDS of M.I.T. in 1987 and at Imperial College in 1992; he also was Visiting Professor at NTUA in 1992. His research interests are in multivariable system and control theory and more recently in autonomous intelligent control systems, and in particular in hybrid and discrete event systems and in neural networks. He is co-editor (with K. M. Passino) of *An Introduction to Intelligent and Autonomous Control* (New York: Kluwer Academic, 1993).

Dr. Antsaklis was the Program Chair of the 30th IEEE CDC in England in 1991, the Guest Editor of the 1990 and 1992 special Issues on Neural Networks in Control Systems in the IEEE CONTROL SYSTEMS MAGAZINE and he is the Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS and the General Chair of the 1993 8th IEEE International Symposium on Intelligent Control.



Anthony N. Michel (S'55–M'59–SM'79–F'82) received the Ph.D. degree in electrical engineering from Marquette University and the D.Sc. degree in applied mathematics from the Technical University of Graz, Austria.

He has seven years of industrial experience. From 1968 to 1984, he was at Iowa State University, Ames, IA. From 1984 to 1988, he was Frank M. Freimann Professor of Engineering and Chairman of the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN. Currently, he is Frank M. Freimann Professor of Engineering and Dean of the College of Engineering at the University of Notre Dame. He is author and coauthor of three texts and several other publications.

Dr. Michel received the 1978 Best Transactions Paper Award of the IEEE Control Systems Society (with R.D. Rasmussen), the 1984 Guillemin–Cauer Prize Paper Award from the IEEE Circuits and Systems Society (with R.K. Miller and B.H. Nam), and a IEEE Centennial Medal. He is a former Associate Editor and a former Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS and a former Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He was the Program Chairman of the 1985 IEEE Conference on Decision and Control. He was President of the IEEE Circuits and Systems Society in 1989. He was Co-Chairman of the 1990 IEEE International Symposium on Circuits and Systems. He was an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS and an Associate Editor at Large for the IEEE AUTOMATIC CONTROL. He was a Fulbright Scholar in 1992 (at the Technical University of Vienna, Austria). He is a Foreign Member of the Academy of Engineering of the Russian Federation.

Kevin M. Passino (S'79–M'89) received his Ph.D. in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 1989.

He has worked in the Control Systems Group at Magnavox Electronic Systems Co., Ft. Wayne, IN, and at McDonnell Aircraft Co., St. Louis, MO, on research in intelligent flight control. He spent a year at Notre Dame as a Visiting Assistant Professor and is currently an Assistant Professor in the Department of Electrical Engineering at Ohio State University, Columbus, OH. His research interests include intelligent and autonomous control, discrete event systems, and stability theory.

Dr. Passino serves as the Chairman of Student Activities for the IEEE Control Systems Society and is on the Editorial Board of the *International Journal for Engineering Applications of Artificial Intelligence.* He was the Guest Editor for the June 1993 SPECIAL ISSUE ON INTELLIGENT CONTROL of the IEEE CONTROL SYSTEMS MAGAZINE. He is Program Cochairman for the 1993 Eight IEEE INTERNATIONAL SYMPOSIUM ON INTELLIGENT CONTROL. He is Coeditor (with P.J. Antsaklis) of the book *An Introduction to Intelligent and Autonomous Control* (New York: Kluwer Academic).

Correction to "Lyapunov Stability of a Class of Discrete-Event Systems"[1]

K. M. Passino, A. N. Michel, and P. J. Antsaklis

In the above article,[1] several errors were made due to font incompatibilities with the typographer. The following corrections apply.

1) The symbol ▲ used throughout the paper should be N, the set of natural numbers ($\mathsf{N} = \{0, 1, 2, 3, \cdots\}$).
2) The symbol = used in (3) and throughout the paper should be "**P**" and hence $\mathsf{P}(Z)$ denotes the power set of Z.
3) The symbol $\gtreqless$ first used in the second paragraph of Section III should be **R**, the set of real numbers.

[1] K. M. Passino, A. N. Michel, and P. J. Antsaklis, *IEEE Trans. Automat. Contr.*, vol. 39, no. 2, pp. 269–279, 1994.