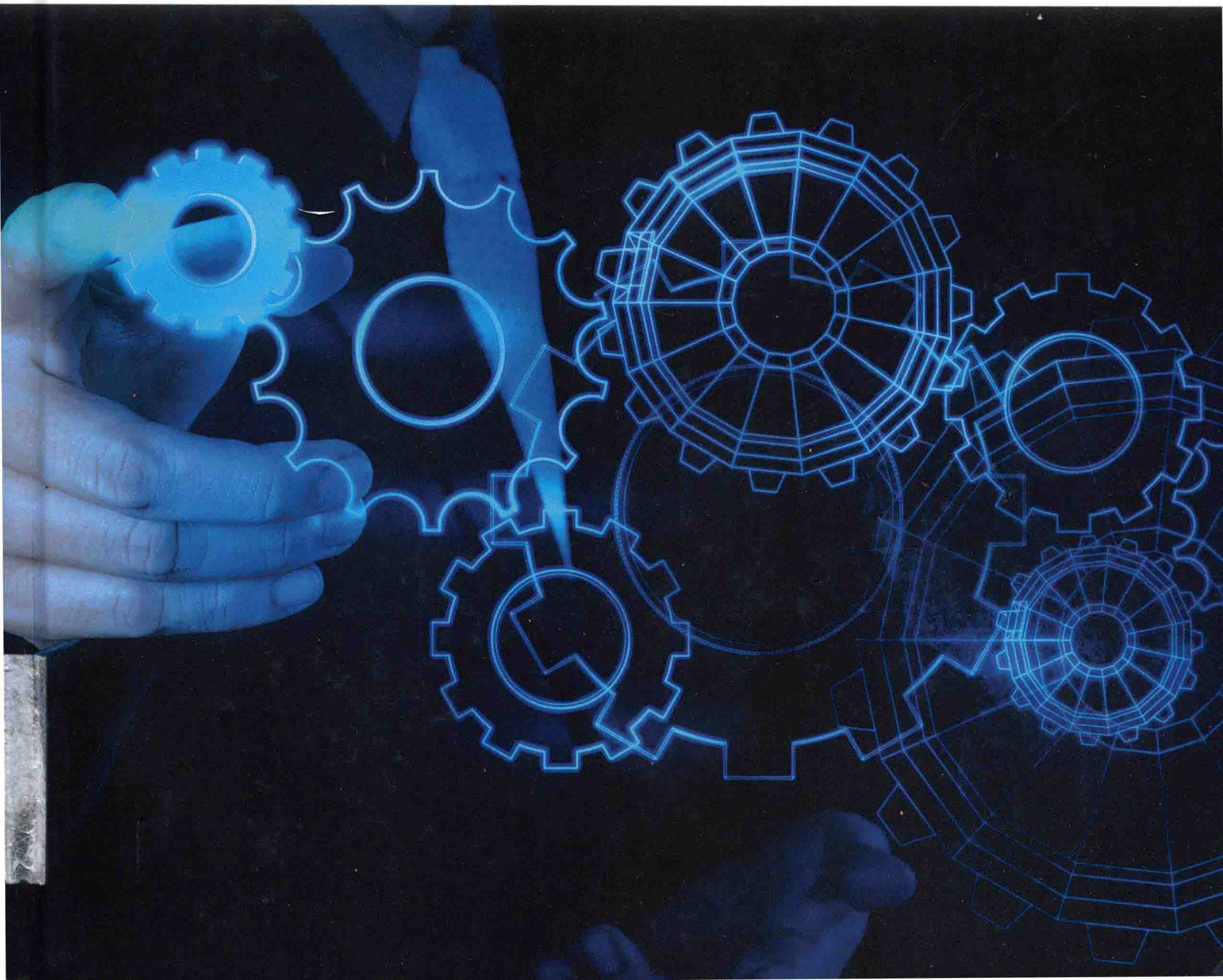
Handbook of Control Science and Engineering

Marques Vang

Volume II



Handbook of Control Science and Engineering Volume II

Edited by **Marques Vang**

Clanrye International
New Jersey

Published by Clanrye International,
55 Van Reypen Street,
Jersey City, NJ 07306, USA
www.clanryeinternational.com

Handbook of Control Science and Engineering: Volume II
Edited by Marques Vang

International Standard Book Number: 978-1-63240-265-3 (Hardback)

Printed in China.

Handbook of Control Science and Engineering

Volume II

Preface

Control science or control systems or control engineering is a field of study and engineering science that integrates automata theory and control theory along with the system designing techniques to build systems with the desired functionality. These systems are designed for self-automation, and the sensors are used to initiate input processing, the measurement converted after the processing of input are further used to initiate output measurement and output controls to be achieved. When a designed system is used without human involvement, they are called automatic control systems. The basic approach to build such systems is mathematical modeling.

Control engineering is playing a large role in designing of control systems, and examples of these control systems can vary from microwave oven to high impact submarines. For such designing, the inputs, outputs, and other important functional components use mathematical modeling to develop controllers for complex systems, and then come the most important task of integrating these controllers with the physical systems using available technological tools. These systems can be of various ranges from mechanical to biological or even financial accounting, they all use control theory in time domains as well as frequency domains, depending on the designing and functional problem.

I especially wish to acknowledge the contributing authors, without whom a work of this magnitude would clearly not be realizable. We thank them for allocating much of their very scarce time to this project. Not only do I appreciate their participation but also their adherence as a group to the time parameters set for this publication. I also thank my publisher for considering me for this project and giving me this incredible opportunity.

Editor

Contents

1

Adaptive Control for a Class of Nonlinear System with Redistributed Models

Haisen Ke and Jiang Li

College of Mechanical and Electrical Engineering, China Jiliang University, Zhejiang, Hangzhou 310018, China

Correspondence should be addressed to Haisen Ke, ksea2001@126.com

Academic Editor: Chengyu Cao

Multiple model adaptive control has been investigated extensively during the last ten years in which the "switching" or "switching and tuning" have emerged as the mainly approaches. It is the "switching" that can improve the transient performance to some extent and also make it difficult to analyze the stability of the system with multiple models adaptive controller. Towards this goal, this paper develops a novel multiple models adaptive controller for a class of nonlinear system in parameter-strict-feedback form which not only improves the transient performance significantly, but also guarantees the stability of all the states of the closed-loop system. A simulation example is proposed to illustrate the effectiveness of the developed multiple models adaptive controller.

1. Introductions

The multiple model adaptive control was introduced to cope with the large parametric uncertainty [1] which always results in large and oscillatory responses or even instable when using the classical adaptive control methods. The multiple models adaptive control [1–7] employing both fixed model and adaptive model have been used to identify the characteristics of the plants, and numerous methods are currently available for controlling such plant satisfactorily. However, the methods mainly focus on the linear time invariant plant [1, 2, 4–6]. The multiple models adaptive controller for nonlinear system is firstly considered in [8], which uses a direct parameter update law to guarantee the stability of the closed-loop system. Then, Ciliz and Cezayirli [9] proposes a different nonlinear multiple models adaptive control which require the condition of persistence of excitation, so that the unknown parameter can be evaluated at the very beginning. Recently, an indirect multiple models adaptive control was developed in [7] which also demonstrated the global asymptotic stability of the closed-loop switching system.

As illustrated in the literature that the "switching" (to the closest model) based on the index of performance results in fast response, and tuning (from the closet model) improves the identification and control errors on a slower time scale, which have the assumption that there are abundant models available. Otherwise, the results may be improved less if the number of the identification models is not adequate to achieve the satisfactory response.

In this paper, a novel multiple models adaptive control was considered for the nonlinear system in parameter-strict-feedback form, which retains the advantages of the multiple models adaptive controller, meanwhile facilitate the procedure to analyze and synthesize the controller of the closed-loop system. The approach developed here in which the multiple models adaptive controller are used to play a significantly larger role in the decision making role, results in substantial improvement in performance. Besides, we also reduce the number of the identification models by redistributing the candidate models even as the system is in operation.

2. Problem Formulation

Consider the multiple models adaptive control of the following nonlinear parameter-strict-feedback (PSF) system:

$$\begin{aligned} \dot{x}_i &= x_{i+1} + \boldsymbol{\varphi}_i^T(\overline{\mathbf{x}}_i)\boldsymbol{\theta}, \quad 1 \le i \le n-1, \\ \dot{x}_n &= \beta(\mathbf{x})u + \boldsymbol{\varphi}_n^T(\mathbf{x})\boldsymbol{\theta}, \\ y &= x_1, \end{aligned} \tag{1}$$

where $\bar{\mathbf{x}}_i = [x_1, \ldots, x_i]^T \in R^i$ and $\mathbf{x} \in R^n$ are the state, $u \in R$ is the control input, $\boldsymbol{\theta} \in R^p$ is an unknown parameter vector belonging to a known compact set S. The functions $\boldsymbol{\varphi}_i(\bar{\mathbf{x}}_i)$ and $\beta(\mathbf{x})$ are known smooth functions with $\beta(\mathbf{x}) \neq 0$, for all $\mathbf{x} \in R^n$. The focus of this paper is to improve the transient performance in the presence of large parametric uncertainties.

One easly way to improve the transient performance may be choosing sufficiently large high-frequency parameters in the conventional backstepping adaptive control design. Unfortunately, the control efforts can also be very large simultaneously [7]. Alternately, in cope with these difficulties, "switching" or "switching and tuning" have emerged as the leading methods during the last decade.

3. Multiple Models Adaptive Controller Design

In order to ensure the stability and transient performance of the system with larger parametric uncertainty, and consequently the boundedness of the state $\mathbf{x}(t)$, the well-established results from the classical adaptive control cannot be used directly. Our multiple models adaptive controller contains N parallel operating identification models on which the control law and the adaptive law are based. For improving the transient performance, it is necessary to distribute the initial estimate values of the unknown parameter $\{\hat{\boldsymbol{\theta}}_j(0)\}_{j=1}^N$ uniformly in the compact set S to which the unknown parameter $\boldsymbol{\theta}$ belongs. Therefore, at least one $\hat{\boldsymbol{\theta}}_j(0)$ is close to $\boldsymbol{\theta}$, consequently there must exists one or more identification models in its neighborhood. Since adaptive control can perform well when parametric errors are small, it is naturally that the controller developed on the jth identification model can stabilize the system with satisfactorily transient performance.

3.1. Multiple Identification Models. We will run in parallel N identification models with the same structure which take the different initial parameter estimate values $\{\hat{\boldsymbol{\theta}}_j(0)\}_{j=1}^N$ uniformly distributed in the compact set S to which the unknown parameter belongs. We first introduce the filters as follows:

$$\dot{\boldsymbol{\xi}}_0 = \left(\mathbf{A}_0 - \lambda \Xi(\mathbf{x})\Xi^T(\mathbf{x})\mathbf{P}\right)(\boldsymbol{\xi}_0 - \mathbf{x}) + f(\mathbf{x}, u), \quad \boldsymbol{\xi}_0 \in R^n, \tag{2}$$

$$\dot{\boldsymbol{\xi}} = \left(\mathbf{A}_0 - \lambda \Xi(\mathbf{x})\Xi^T(\mathbf{x})\mathbf{P}\right)\boldsymbol{\xi} + \Xi(\mathbf{x}), \quad \boldsymbol{\xi} \in R^{n\times p}, \tag{3}$$

where

$$\begin{aligned} f(\mathbf{x}, u) &= \begin{bmatrix} x_2 & \ldots & x_n & \beta(\mathbf{x})u \end{bmatrix}^T, \\ \Xi(\mathbf{x}) &= \begin{bmatrix} \varphi_1(x_1) & \ldots & \varphi_n(\mathbf{x}) \end{bmatrix}^T. \end{aligned} \tag{4}$$

$\lambda > 0$, and $\mathbf{A}_0$ is a Hurwitz matrix such that the Lyapunov equation: $\mathbf{P}\mathbf{A}_0 + \mathbf{A}_0^T\mathbf{P} = -\mathbf{I}$ has a positive definite solution P.

Define

$$\tilde{\mathbf{e}} = \mathbf{x} - \boldsymbol{\xi}_0 - \boldsymbol{\xi}\boldsymbol{\theta}, \tag{5}$$

$$\mathbf{e}_j = \mathbf{x} - \boldsymbol{\xi}_0 - \boldsymbol{\xi}\hat{\boldsymbol{\theta}}_j, \quad j = 1, \ldots, N, \tag{6}$$

$$\tilde{\boldsymbol{\theta}}_j = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_j, \quad j = 1, \ldots, N. \tag{7}$$

It can be derived from (1)–(7) that

$$\dot{\tilde{\mathbf{e}}} = \left(\mathbf{A}_0 - \lambda \Xi(\mathbf{x})\Xi^T(\mathbf{x})\mathbf{P}\right)\tilde{\mathbf{e}}, \tag{8}$$

$$\mathbf{e}_j = \xi\tilde{\boldsymbol{\theta}}_j + \tilde{\mathbf{e}}, \quad j = 1, \ldots, N. \tag{9}$$

Since $\tilde{\mathbf{e}}$ converges to zero exponentially, (9) are called identification error equations.

3.2. Controller Design. The controller design involves N models at total and is developed as [10], which can guarantee the asymptotic tracking when there is not identification error and avoid the finite time escape phenomenon when there exists bounded identification error. Now, the first identification model's adaptive controller is given by

$$u_1 = \frac{\left[\alpha_{1,n}\left(\mathbf{x}, \hat{\boldsymbol{\theta}}_1, y_r, \ldots, y_r^n\right)\right]}{\beta(\mathbf{x})}, \tag{10}$$

where y_r is the reference signal to be tracked and $\alpha_{1,n}$ can be recursively designed by

$$z_i = x_i - \alpha_{1,i-1}\left(x_1, \ldots, x_{i-1}, \hat{\boldsymbol{\theta}}_1, y_r, \ldots y_r^{i-1}\right), \tag{11}$$

$$\begin{aligned} \alpha_{1,i} = {} & -z_{i-1} - c_{1,i}z_i - w_{1,i}^T\hat{\boldsymbol{\theta}}_1 + y_r^i - s_{1,i}z_i \\ & + \sum_{k=1}^{i-1}\left(\frac{\partial \alpha_{1,i-1}}{\partial \mathbf{x}_k}\mathbf{x}_{k+1} + \frac{\partial \alpha_{1,i-1}}{\partial y_r^{k-1}}y_r^k\right), \end{aligned} \tag{12}$$

$$w_{1,i}\left(x_1, \ldots x_i, \hat{\boldsymbol{\theta}}_1, y_r, \ldots, y_r^{i-1}\right) = \boldsymbol{\varphi}_i - \sum_{k=1}^{i-1}\frac{\partial \alpha_{1,i-1}}{\partial x_k}\boldsymbol{\varphi}_k, \tag{13}$$

$$s_{1,i} = k_{1,i}\left|w_{1,i}\right|^2 + g_{1,i}\left|\frac{\partial \alpha_{1,i-1}}{\partial \hat{\boldsymbol{\theta}}_1}\right|^2, \tag{14}$$

We choose

$$V_1 = \frac{1}{2}\sum_{i=1}^n z_i^2. \tag{15}$$

The time derivative of V_1, computed with (10)–(14), is given by

$$\begin{aligned}\dot{V}_1 &= -\sum_{i=1}^{n} c_{1,i} z_i^2 + \sum_{i=1}^{n}\left(w_{1,i}^T \tilde{\boldsymbol{\theta}}_1 - \frac{\partial \alpha_{1,i-1}}{\partial \hat{\boldsymbol{\theta}}_1}\dot{\hat{\boldsymbol{\theta}}}_1\right) z_i \\ &\quad - \sum_{i=1}^{n}\left(k_{1,i}\left|w_{1,i}\right|^2 + g_{1,i}\left|\frac{\partial \alpha_{1,i-1}}{\partial \hat{\boldsymbol{\theta}}_1}\right|^2\right) z_i^2 \\ &\le -\sum_{i=1}^{n} c_{1,i} z_i^2 - \sum_{i=1}^{n} k_{1,i}\left|w_{1,i} z_i - \frac{1}{2k_{1,i}}\tilde{\boldsymbol{\theta}}_1\right|^2 \\ &\quad - \sum_{i=1}^{n} g_{1,i}\left|\frac{\partial \alpha_{1,i-1}}{\partial \hat{\boldsymbol{\theta}}_1} z_i - \frac{1}{2g_{1,i}}\dot{\hat{\boldsymbol{\theta}}}_1\right|^2 \\ &\quad + \sum_{i=1}^{n}\frac{1}{4k_{1,i}}\left|\tilde{\boldsymbol{\theta}}_1\right|^2 + \sum_{i=1}^{n}\frac{1}{4g_{1,i}}\left|\dot{\hat{\boldsymbol{\theta}}}_1\right|^2 \\ &\le -\sum_{i=1}^{n} c_{1,i} z_i^2 + \sum_{i=1}^{n}\frac{1}{4k_{1,i}}\left|\tilde{\boldsymbol{\theta}}_1\right|^2 + \sum_{i=1}^{n}\frac{1}{4g_{1,i}}\left|\dot{\hat{\boldsymbol{\theta}}}_1\right|^2,\end{aligned} \tag{16}$$

with $c_{1,i}$, $k_{1,i}$, $g_{1,i}$ being designed parameters. Equation (16) implies the boundedness of the states of z_i, $1 \le i \le n$, and which in turn indicates the boundedness of the states of x_i, $1 \le i \le n$ and control u_1 on the conditions of $\tilde{\boldsymbol{\theta}}$ and $\dot{\hat{\boldsymbol{\theta}}}$ are bounded which will be proved later. The rest of N-1 controllers can be designed and analyzed similarly which can also guarantee the boundedness of the states of z_i, $1 \le i \le n$, and which in turn indicates the boundedness of the states of x_i, $1 \le i \le n$ and control u_j, $j = 1, \ldots, N$.

3.3. Construction of Equivalent Control. In this section, the crucial point is that the transient performance can be improved significantly, and at the same time the switching between the identification models can be avoided. Besides, the information provided by all the identification models is to be utilized efficiently. For the complement of the goals mentioned, instead of using the estimate values of the model with the minimum of performance criterion to reinitial an adaptive controller, a convex combination of all the N models is used to generate the control of the plant as

$$u = \sum_{j=1}^{N} \gamma_j u_j, \tag{17}$$

and the adaptive update law as

$$\dot{\hat{\boldsymbol{\theta}}}_j = \boldsymbol{\Gamma}\frac{\boldsymbol{\xi}^T \mathbf{e}_j}{1 + \nu|\boldsymbol{\xi}|^2}, \quad \boldsymbol{\Gamma} = \boldsymbol{\Gamma}^T > 0,\ \nu > 0,\ j = 1, \ldots, N, \tag{18}$$

where γ_j are nonnegative values satisfying $\sum_{j=1}^{N} \gamma_j = 1$, and γ_j can be calculated from

$$\gamma_j = \frac{(1/J_j)}{\sum_{j=1}^{N}(1/J_j)}, \tag{19}$$

where J_j is the performance indices of the form:

$$J_j(t) = \alpha e_j^2(t) + \beta\int_{t_0}^{t} e_j^2(\tau)\,d\tau, \quad a \ge 0,\ \beta > 0, \tag{20}$$

with t_0 can be reset when the identification models is redistributed.

3.4. Redistribution of the Identification Models. In this section, the goal is that the transient performance can be improved significantly as far smaller numbers of the identification models as possible. As is illustrated in the literature, the classical adaptive control can cope with the control of linear time invariant system with unknown parameters and achieve satisfactory closed-loop objective only if the plant parametric uncertainty is small. So if the number of the identification models that can be used is abundantly large, the "switching" or "switching and tuning" scheme may act on satisfactorily. Otherwise, the multiple models adaptive control cannot work as expected when the numbers of identification models available is relatively smaller compared with the size of the uncertainty region. Inspired by the "switching" techniques [11–13], we consider the method in which the location of the identification models can be redistributed. From (8) and (9), it can be concluded that the $\tilde{\mathbf{e}} = 0$ can be achieved by choosing the initial values of ξ_0 and ξ as long as the initial state x_0 is known or there exists $T > 0$ such that

$$\mathbf{e}_j = \boldsymbol{\xi}\tilde{\boldsymbol{\theta}}_j, \quad j = 1, \ldots, N,\ t > T. \tag{21}$$

It is obviously that the errors $\mathbf{e}_j$ and $\tilde{\boldsymbol{\theta}}_j$, $j = 1, \ldots, N$ are linearly related. This implies that the index of the performance $J_j(t)$ is a quadratic function of the unknown parameter vector $\tilde{\boldsymbol{\theta}}_j$. Since $\xi^T\xi$ is not negative definite, it follows that the performance indices of all the models are merely points on a time-varying quadratic surface, whose minimum corresponds to the plant indicating the mostly closet identification model M_j (corresponds to the parameter $\tilde{\boldsymbol{\theta}}_j$). So we can redistribute the other $(N-1)$ models M_k $(k \ne j)$ as

$$\overline{\boldsymbol{\theta}}_k = \frac{\sqrt{J_k}}{\sqrt{J_k} + \sqrt{J_j}}\boldsymbol{\theta}_j + \frac{\sqrt{J_j}}{\sqrt{J_k} + \sqrt{J_j}}\boldsymbol{\theta}_k. \tag{22}$$

By introducing the minimum of interval time $T_{\min}$ into our switching scheme to ensure a finite number of switching.

4. Stability Analysis

Theorem 1. *Suppose the multiple models adaptive controller (17) and adaptive law (18) presented in this paper is applied to system (1). Then, for all initial conditions, all closed-loop states are bounded on $[0, \infty)$, and asymptotic tracking can be achieved, that is, $Lim_{t\to\infty} z(t) = 0$ or $y(t) = y_r(t)$ as $t \to \infty$.*

Proof. Since all N models are identical structure and only with different initial estimate parameters, it follows that each controller acts on the system is only different from each other at the weight (each of the controllers can be designed with the same structure and designed parameters).

When we choose the whole candidate Lyapunov function as

$$V = \frac{1}{2}\sum_{i=1}^{n} z_i^2. \tag{23}$$

It is obvious that (23) can be divided into

$$V = \frac{1}{2}\sum_{i=1}^{n} z_i^2 = \sum_{j=1}^{N} \gamma_j \frac{1}{2}\sum_{i=1}^{n} z_i^2 = \sum_{j=1}^{N} \gamma_j V_j. \tag{24}$$

As illustrated by (16), each of the controllers can guarantee the boundedness of the states of z_i, $1 \le i \le n$ at its portion, which accompanied with the control (17), and weighting coefficient (19) can establish the boundedness of the states of z_i, $1 \le i \le n$.

Next, we prove the controller (17) and adaptive law (18) can also guarantee the asymptotic tracking of the closed-loop system states. It can be computed from (3) that

$$\begin{aligned}\frac{d}{dt}\left(\xi \mathbf{P} \xi^T\right) &= -\xi\xi^T - 2\lambda \xi \mathbf{P} \Xi^T \Xi \mathbf{P} \xi^T + \xi \mathbf{P} \Xi^T + \Xi \mathbf{P} \xi^T \\ &= -\xi\xi^T - 2\lambda \left(\Xi \mathbf{P} \xi^T - \frac{1}{2\lambda}\mathbf{I}\right)^T \left(\Xi \mathbf{P} \xi^T - \frac{1}{2\lambda}\mathbf{I}\right) + \frac{1}{2\lambda},\end{aligned} \tag{25}$$

which shows $\boldsymbol{\xi}$ is bounded regardless of the state $\mathbf{x}$. Let $V_j = (\widetilde{\boldsymbol{\theta}}_j^T \boldsymbol{\Gamma}^{-1} \widetilde{\boldsymbol{\theta}}_j + \widetilde{\mathbf{e}}^T \widetilde{\mathbf{e}})/2$, it can be derived that

$$\begin{aligned}\dot{V}_j &= -\frac{\mathbf{e}_j^T \left(\mathbf{e}_j - \widetilde{\mathbf{e}}\right)}{1 + v|\boldsymbol{\xi}|^2} + \widetilde{\mathbf{e}}^T \left(\mathbf{A}_0 - \lambda \Xi(\mathbf{x}) \Xi^T(\mathbf{x}) \mathbf{P}\right) \widetilde{\mathbf{e}} \\ &\le -\frac{3}{4} \frac{\mathbf{e}_j^T \mathbf{e}_j}{1 + v|\boldsymbol{\xi}|^2},\end{aligned} \tag{26}$$

without loss of generality, we can design the parameter satisfies $\mathbf{A}_0 - \lambda \Xi(\mathbf{x}) \Xi^T(\mathbf{x}) \mathbf{P} > I$, I is a unit matrix. Therefore, $\widetilde{\mathbf{e}}$, $\widetilde{\boldsymbol{\theta}}_j$, $j = 1, \ldots, N$ are all bounded, which companied with the boundedness of $\boldsymbol{\xi}$, further yields from $\mathbf{e}_j = \boldsymbol{\xi} \widetilde{\boldsymbol{\theta}}_j + \widetilde{\mathbf{e}}$ that $\mathbf{e}_j$ is bounded. It can be also concluded from (26) that $\mathbf{e}_j$ is squarely integrable on $[0, \infty)$. Furthermore, we can also conclude from (18) that $\dot{\hat{\boldsymbol{\theta}}}_j$ is bounded, which can accomplish the assumption that it is bounded. We can now give the asymptotically tracking control analysis.

The time derivate of identification error is given by

$$\dot{\mathbf{e}}_j = \left(\mathbf{A}_0 - \lambda \Xi \Xi^T \mathbf{P}\right) \mathbf{e}_j + \Xi \widetilde{\boldsymbol{\theta}}_j - \boldsymbol{\xi} \dot{\hat{\boldsymbol{\theta}}}_j. \tag{27}$$

Due to the boundedness of all the closed-loop system states $\dot{\mathbf{e}}_j$, $\ddot{\mathbf{e}}_j$, $j = 1, \ldots, N$ are also bounded, so by Barbalat's lemma, we must have $\mathrm{Lim}_{t\to\infty} \mathbf{e}_j(t) = 0$ and since $\mathrm{Lim}_{t\to\infty} \int_{t_1}^{t} \dot{\mathbf{e}}_j(\tau) d\tau = \lim_{t\to\infty} \mathbf{e}_j(t) - \mathbf{e}_j(t_1) < \infty$, we further have $\mathrm{Lim}_{t\to\infty} \dot{\mathbf{e}}_j(t) = 0$. Then, it can be concluded from (18) that $\mathrm{Lim}_{t\to\infty} \dot{\hat{\boldsymbol{\theta}}}_j = 0$ which accompanied with (27) implies $\mathrm{Lim}_{t\to\infty} \Xi \widetilde{\boldsymbol{\theta}}_j = 0$ and in turn leads to

$$\mathrm{Lim}_{t\to\infty} N_j\left(z, \hat{\boldsymbol{\theta}}, y_r\right) \,\Xi\, \widetilde{\boldsymbol{\theta}}_j = 0, \tag{28}$$

where

$$N_j\left(z, \hat{\boldsymbol{\theta}}_j, y_r\right) = \begin{bmatrix} 1 & 0 & & 0 \\ -\dfrac{\partial \alpha_{j,1}}{\partial x_1} & 1 & 0 & \\ \vdots & & \ddots & 0 \\ -\dfrac{\partial \alpha_{j,n-1}}{\partial x_1} & -\dfrac{\partial \alpha_{j,n-1}}{\partial x_2} & & 1 \end{bmatrix}. \tag{29}$$

By direct calculating, the differentiation of z with parametric controller u_j can be described, in z coordination, by

$$\begin{aligned}\dot{z} = &-\sum_{i=1}^{n} c_{j,i} z_i + N_j\left(z, \hat{\boldsymbol{\theta}}, y_r\right) \Xi \widetilde{\boldsymbol{\theta}}_j - \frac{\partial \alpha_{j,i-1}}{\partial \hat{\boldsymbol{\theta}}_j} \dot{\hat{\boldsymbol{\theta}}}_j \\ &- \sum_{i=1}^{n} \left(k_{j,i} \left| w_{j,i} \right|^2 + g_{j,i} \left| \frac{\partial \alpha_{j,i-1}}{\partial \hat{\boldsymbol{\theta}}_j} \right|^2 \right) z_i.\end{aligned} \tag{30}$$

From (28), accompanied with $\mathrm{Lim}_{t\to\infty} \dot{\hat{\boldsymbol{\theta}}}_j = 0$ and the designed parameters are all positive, it can be easily concluded that $\mathrm{Lim}_{t\to\infty} z(t) = 0$, and thus $\mathrm{Lim}_{t\to\infty} \mathbf{z}_1(t) = \mathrm{Lim}_{t\to\infty}(y(t) - y_r(t)) = 0$. The proof is completed. □

5. Simulation

Consider the following second-order nonlinear system:

$$\begin{aligned}\dot{x}_1 &= x_2 + \theta_1 x_1 + \theta_2 x_1^2, \\ \dot{x}_2 &= u, \\ y(t) &= x_1(t),\end{aligned} \tag{31}$$

where $\theta_1 \in [1, 5]$ and $\theta_2 \in [1, 40]$ are unknown parameters. The output $y(t) = x_1(t)$ is to asymptotically track the reference signal $y_r(t) = \sin 2t$.

In simulation, the parametric controller is developed as (10)–(14) and (17)–(19) with $v = 0$, $\Gamma = 5$, $c_{j,1} = c_{j,2} = 4$, $k_{j,1} = k_{j,2} = g_{j,2} = 0.1$, $j = 1, \ldots, N$, $\alpha = \beta = 1$, $T_{\min}$ is 5 units of time. Since in (31), the unknown parameter appears only in the first equation, the filter can be constructed as [1] to reduce filter dynamic order:

$$\begin{aligned}\dot{\xi}_0 &= -c(\xi_0 - \mathbf{x}) + x_2, \quad \xi_0 \in R^1, \\ \dot{\boldsymbol{\xi}} &= -c\boldsymbol{\xi} + [x_1, x_1^2], \quad \boldsymbol{\xi} \in R^{1\times 2},\end{aligned} \tag{32}$$

where $c = 10$.

The unknown parameter is $[\theta_1, \theta_2] = [4.4, 38.5]$; the number of the multiple identification models is $N = 4$; for convenience to comparison with [7], the initial plant state is $[x_1(0), x_2(0)] = [0.5, -10]$; the same initial filter states are $\xi_0 = 0.5$, $\xi = \begin{bmatrix} 0 & 0 \end{bmatrix}$, and the initial estimate parameters for model 1, model 2, model 3, and model 4 are $\hat{\boldsymbol{\theta}}_1(0) = [1, 1]^T$, $\hat{\boldsymbol{\theta}}_2(0) = [1, 5]^T$, $\hat{\boldsymbol{\theta}}_3(0) = [5, 1]^T$, $\hat{\boldsymbol{\theta}}_4(0) = [5, 40]^T$, respectively. Figures 1–4 depict the simulation results.

These simulation results clearly showed that the multiple models adaptive controller presented in this paper guarantees the boundedness of all the states in the closed-loop system and achieves the asymptotic tracking of the output.

Figure 1 is the output $y(t)$, which demonstrates that the multiple models adaptive controller developed in this paper has the similar property as shown in [7] and is significantly better than using the classical adaptive control. Figures 2 and 3 are the control inputs which show that the multiple model adaptive control can reduce the maximum control input dramatically. Besides, it seems to conclude that the multiple model adaptive control proposed in this paper has the similar property and so the trajectory is nearly to overlap

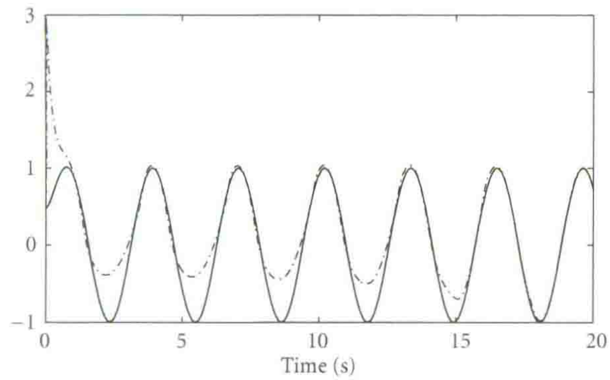


FIGURE 1: Output $y(t)$. dash-dotted line for the classical adaptive control, dashed line for the multiple model case ($N = 200$) as in [7], and solid line for the multiple identification model developed in this paper.

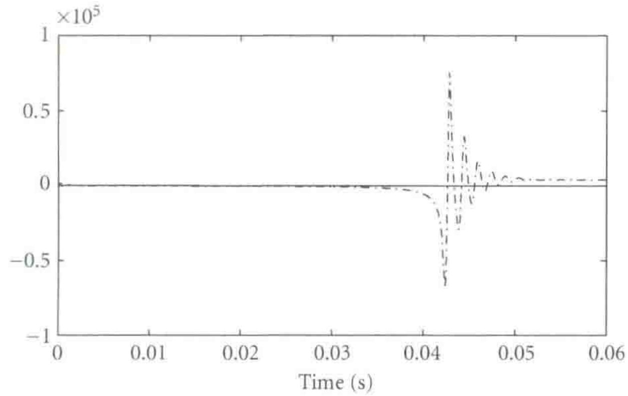


FIGURE 2: Control $u(t)$ on the time interval [0, 0.06]. dash-dotted line for the classical adaptive control, dashed line for the multiple model case ($N = 200$) as in [7], and solid line for the multiple identification model developed in this paper.

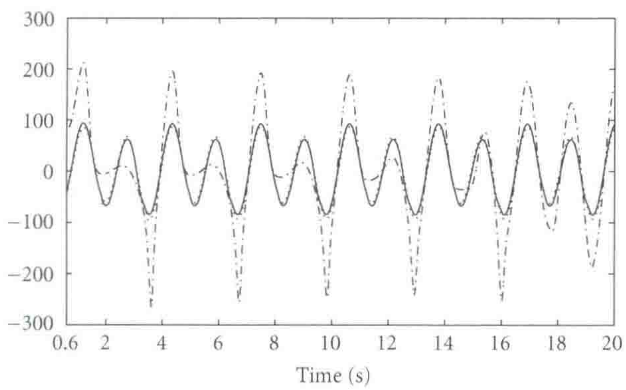


FIGURE 3: Control $u(t)$ on the time interval [0.6, 20]. dash-dotted line for the classical adaptive control, dashed line for the multiple model case ($N = 200$) as in [7], and solid line for the multiple identification model in this paper.

from Figures 1–3 because the method in [7] uses more identification models than ours. Figure 4 is the trajectory of the redistribution of the identification models which can find the most suitable identification model and enhance the transient performance.

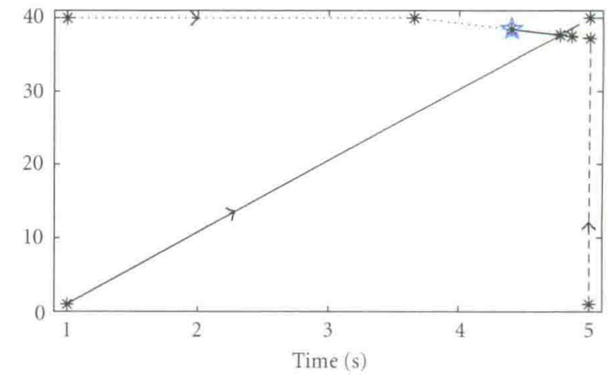


FIGURE 4: The redistribution trajectory of the identification models.

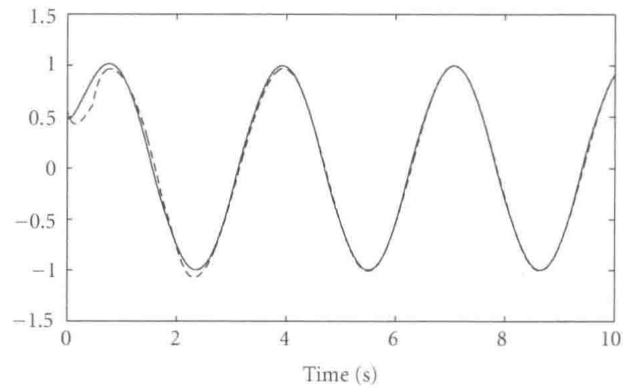


FIGURE 5: Output $y(t)$. dashed line for the multiple model case ($N = 4$) as in [7], solid line for the multiple identification model developed in this paper.

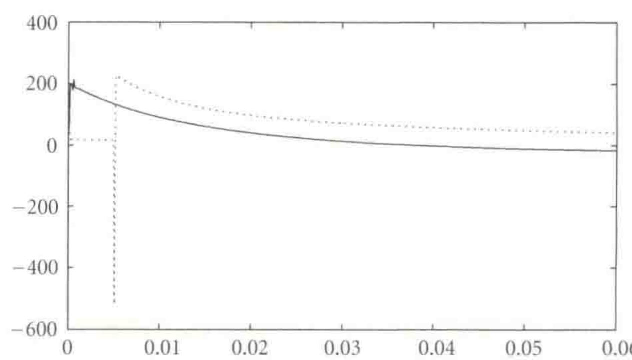


FIGURE 6: Control $u(t)$ on the time interval [0, 0.06]. dashed line for the multiple model case ($N = 4$) in [7], and solid line for the multiple model developed in this paper.

Next, we can compare the approach presented in this paper with the method developed in [7] with the multiple identification models ($N = 200$) is set to ($N = 4$), which is the same identification models used in our approach. Figures 5–7 depict the simulation results.

Figure 5 is the output $y(t)$ with the multiple models adaptive controller, which shows the approach developed in this paper is superior to the method presented in [7]. Figures 6 and 7 are the control inputs which show that the multiple model adaptive control developed in this paper has

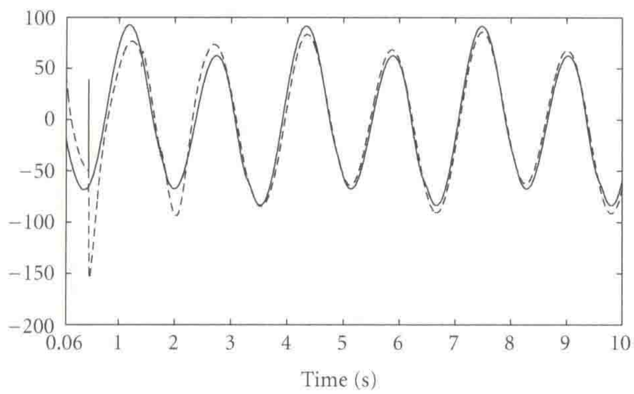


FIGURE 7: Control $u(t)$ on the time interval [0.06, 10]. dashed line for the multiple model case ($N = 4$) in [7], and solid line for the multiple model developed in this paper.

better properties than the method presented in [7] which has switching and larger control input.

6. Conclusions

In this paper, a novel multiple models adaptive controller was developed for a class of nonlinear systems. The multiple models technique was used to describe the most appropriate model at different environments. If the number of the identification models that can be used is abundantly large, the "switching" or "switching and tuning" scheme may act on satisfactorily. Otherwise, the multiple models adaptive control cannot work as expected when the number of identification models available is relatively small compared with the size of the uncertainty region. So we consider the method in which the location of the identification models can be redistributed. Unlike previous results, we do not require a switching scheme to guarantee the most appropriate model to be switched into the controller design which can simplify the analysis of the stability of the closed-loop system.

Acknowledgments

The authors would like to thank the reviewers and the editors for their helpful and insightful comments for further improvement of the quality of this work. The work is supported by the National Natural Science Foundation of China (nos. 60674023 and 60905034) and the Zhejiang Provincial Natural Science Foundation of China (no. Y12F030119).

References

[1] K. S. Narendra and J. Balakrishnan, "Improving transient response of adaptive control systems using multiple models and switching," *IEEE Transactions on Automatic Control*, vol. 39, no. 9, pp. 1861–1866, 1994.

[2] K. S. Narendra and J. Balakrishnan, "Adaptive control using multiple models," *IEEE Transactions on Automatic Control*, vol. 42, no. 2, pp. 171–187, 1997.

[3] J. P. Hespanha, D. Liberzon, and A. S. Morse, "Overcoming the limitations of adaptive control by means of logic-based switching," *Systems and Control Letters*, vol. 49, no. 1, pp. 49–65, 2003.

[4] Z. Han and K. S. Narendra, "Multiple adaptive models for control," in *2010 49th IEEE Conference on Decision and Control, CDC 2010*, pp. 60–65, usa, December 2010.

[5] K. S. Narendra and Z. Han, "The changing face of adaptive control: the use of multiple models," *Annual Reviews in Control*, vol. 35, no. 1, pp. 1–12, 2011.

[6] K. S. Narendra and K. George, "Adaptive control of simple nonlinear systems using multiple models," in *2002 American Control Conference*, pp. 1779–1784, usa, May 2002.

[7] X. D. Ye, "Nonlinear adaptive control using multiple identification models," *Systems and Control Letters*, vol. 57, no. 7, pp. 578–584, 2008.

[8] K. S. Narendra and K. George, "Adaptive control of simple nonlinear systems using multiple models," in *2002 American Control Conference*, pp. 1779–1784, usa, May 2002.

[9] M. K. Ciliz and A. Cezayirli, "Increased transient performance for the adaptive control of feedback linearizable systems using multiple models," *International Journal of Control*, vol. 79, no. 10, pp. 1205–1215, 2006.

[10] M. Krstic and P. V. Kokotovic, "Adaptive nonlinear design with controller-identifier separation and swapping," *IEEE Transactions on Automatic Control*, vol. 40, no. 3, pp. 426–440, 1995.

[11] H. S. Ke and X. D. Ye, "Robust adaptive controller design for a class of nonlinear systems with unknown high frequency gains," *Journal of Zhejiang University*, vol. 7, no. 3, pp. 315–320, 2006.

[12] P. Bashivan and A. Fatehi, "Improved switching for multiple model adaptive controller in noisy environment," *Journal of Process Control*, vol. 22, no. 2, pp. 390–396, 2012.

[13] H. H. Xiong and S. Y. Li, "Satisfying optimal control of switching multiple models based on mixed logic dynamics," *International Journal of Modelling, Identification and Control*, vol. 10, no. 1-2, pp. 175–180, 2010.

2

Process Completing Sequences for Resource Allocation Systems with Synchronization

Song Foh Chew,[1] Shengyong Wang,[2] and Mark A. Lawley[3]

[1] *Department of Mathematics and Statistics, Southern Illinois University, Edwardsville, IL 62026, USA*
[2] *Department of Mechanical Engineering, University of Akron, Akron, OH 44325, USA*
[3] *School of Biomedical Engineering, Purdue University, West Lafayette, IN 47907, USA*

Correspondence should be addressed to Mark A. Lawley, malawley09@yahoo.com

Academic Editor: Jin-Shyan Lee

This paper considers the problem of establishing live resource allocation in workflows with synchronization stages. Establishing live resource allocation in this class of systems is challenging since deciding whether a given level of resource capacities is sufficient to complete a single process is NP-complete. In this paper, we develop two necessary conditions and one sufficient condition that provide quickly computable tests for the existence of process completing sequences. The necessary conditions are based on the sequence of completions of n subprocesses that merge together at a synchronization. Although the worst case complexity is $O(2^n)$, we expect the number of subprocesses combined at any synchronization will be sufficiently small so that total computation time remains manageable. The sufficient condition uses a reduction scheme that computes a sufficient capacity level of each resource type to complete and merge all n subprocesses. The worst case complexity is $O(n \cdot m)$, where m is the number of synchronizations. Finally, the paper develops capacity bounds and polynomial methods for generating feasible resource allocation sequences for merging systems with single unit allocation. This method is based on single step look-ahead for deadly marked siphons and is $O(2^n)$. Throughout the paper, we use a class of Petri nets called Generalized Augmented Marked Graphs to represent our resource allocation systems.

1. Introduction

In recent years, liveness-enforcing supervisory control has been an active area of research for resource allocation systems characterized by processes with highly ordered, linear workflows. This research has been motivated to a large degree by the need to control resource allocation in large, highly automated manufacturing systems, where process workflow is highly sequential and is typically prespecified in a product's process plan. In brief, a sequential resource allocation system (RAS) consists of a set of resources, each available at a finite level, and a set of processes that progresses through sequences of processing stages, with each stage requiring a predetermined set of the system resources. Furthermore, a process instance is allowed to advance to its next stage only when it has been granted the complete set of required resources and only then will it release the currently held resources that are not required for the following stage.

Because the resource allocation schemes discussed above are embedded in the operation of many technologically advanced systems, a complete understanding of their worst case behaviors is essential when devising operating logic for their control. Indeed, if resource allocation is not properly constrained, the sequential RAS will attain resource allocation states from which additional allocation-deallocation of some subset of resources is not possible. This situation is highly undesirable, because resource allocation stalls, the involved processes and the resources they hold are idle, and outside intervention to resolve and reset the system is required. Liveness enforcing supervision seeks to avoid these situations and maintain completely smooth operation by imposing an appropriate supervisory control policy.

Reveliotis et al. [1] present a taxonomy for sequential RAS based on the structure of the allocation requests associated with various processing stages. This taxonomy includes (i) single-unit (SU) RAS, which admits only linearly ordered process sequences with resource requests corresponding to standard unit vectors, (ii) conjunctive (C) RAS, which admits linearly ordered process flows with arbitrary resource requests, and (iii) disjunctive/conjunctive (D/C) RAS, which allows the process to use alternative workflow sequences. Lower-numbered classes in the taxonomy are specializations of the higher-numbered and therefore present simpler behaviors which are more easily analyzed and controlled. Indeed, many results on RAS liveness and the synthesis of tractable liveness enforcing supervisors (LES) have been developed for the SU-RAS class, see, for example, [2, 3] for seminal papers. Researchers have also addressed the problem in the context of the more general classes of D-RAS, C-RAS, and D/C-RAS, see [4, 5] for early results. An interesting discussion that provides a unifying perspective for many of these results, and also highlights the currently prevailing issues in the area, can be found in [6]. Additional recent reviews are provided in [7, 8].

In [9], Reveliotis et al. extends the taxonomy of [1] to include RAS with process synchronizations, that is, RAS where a process may consist of several subprocesses operating independently until some synchronization stage is attained, at which point subprocesses recombine through merging and splitting and then continue as a new set. We shall refer to this class of RAS as A/D-RAS (assembly/disassembly RAS), since, in the case of manufacturing, this class covers products with both assembly and disassembly in their specified workflow. We notice, however, that synchronization also commonly occurs in project management and business workflow scenarios where finite resources must be allocated to competing tasks, which must eventually merge and spawn successor tasks.

From the perspective of logical analysis and control, a major difference between the A/D-RAS and those addressed in the taxonomy of [1] is that we can no longer quickly be sure that the given level of resource capacities is sufficient to complete even a single process. More specifically, since a single process may consist of several concurrent and independently operating subprocesses, each requesting, using, and holding resources, there is no guarantee that resources are of sufficient capacity to allow these subprocesses to attain required synchronization states. In this paper, we refer to this issue as the "quasi-liveness" problem since, by definition, an underlying Petri net model of the A/D-RAS will be quasi-live if, for every transition of the net (including those representing synchronizations), there exists a sequence of transition firings (resource allocations) that enables that transition. In [9], it is established that the lack of quasi-liveness in the A/D-RAS can be explained by the presence of a particular type of deadly marked siphon in the underlying net dynamics and that testing quasi-liveness, a rather easy task for nets modeling the D/C-RAS, now becomes an NP-complete problem (cf. also [10] for a formal proof on the NP-completeness of the quasi-liveness problem in the considered RAS class). Thus, assessing process quasi-liveness raises important and novel research problems to be addressed for this RAS. For quasi-live processes, an additional issue is identifying sequences of resource allocations that enable the involved process synchronizations. Once such sequences have been identified, standard D/C-RAS deadlock avoidance policies can be implemented to control concurrent allocation of resources across several concurrently operating processes.

We note that in [11], Xie and Jeng also study resource allocation in systems with synchronizations by analyzing a class of ordinary Petri nets called extended resource control nets (ERCN). More specifically, they develop structural characterizations for the ERCN quasi-liveness and liveness that are based on the notion of empty siphons. In other work, Wu et al. [12] model assembly/disassembly processes using resource-oriented Petri nets. Based on the models, a deadlock control policy is proposed and proved to be computationally efficient and less conservative than the existing policies in the literature. Hsieh [13] develops a subclass of Petri net models called nonordinary controlled flexible assembly Petri nets with uncertainties for assembly systems and studies their robustness to resource failure. Hu et al. [14] proposes a class of Petri nets to study automated manufacturing systems with either flexible routes or assembly operations. Using structural analysis, the authors show that liveness of such systems can be attributed to the absence of under-marked siphons.

Our work, on the other hand, places more emphasis on the associated design and control problems, seeking first to find resource levels that guarantee quasi-liveness and then to find resource allocation sequences that enable synchronization transitions. In [15, 16], we model the A/D-RAS using a subclass of Petri nets known as Generalized Augmented Marked Graphs (G-AMG). Based upon the notion of reachability graph, we present an algorithm that determines the quasi-liveness of a process subnet by enumerating all execution sequences that are resource-enabled under the considered resource availability; if the net is quasi-live, there will be at least one sequence that leads to process completion. For a quasi-live process, the reachability graph provides complete information about the resource allocation sequences that can be used. Since the graph is exponential in size, it is generally necessary to select a smaller subset of sequences to use for supervision. Based on the work presented in [15, 16], Choi [17] develops a mixed integer program that selects a small subset of process completing sequences for the development of liveness enforcing supervisors. This defines a manageable set of realizable behaviors the system can exhibit. The subset is selected such that a performance controller, posed as a Markov decision process, has the greatest potential to optimize system performance.

In this paper, we seek to develop more tractable methods of identifying process completing sequences for certain subclasses. More specifically, we define a special case of G-AMG, called G-AMG_A, which models a RAS comprising only "assembly" or merging operations. For RAS modeled by G-AMG_A's, we develop two necessary conditions for quasi-liveness which provide quick tests. We also develop a polynomial net reduction algorithm that can be used to compute resource levels sufficient to assure quasi-liveness.