systems considered is assumed to have the defined Lyapunov detectability property and a dividability condition. If those conditions are satisfied, a switching observer can be constructed so that the state variables can be estimated without prior knowledge of the parameters or the structure of the systems. This type of estimation is particularly useful in the cases where there is no detailed system model, where there exist potential component failures, or where some of the system structure or properties are manually shifted. Whenever those happen, the proposed observer can adaptively detect changes in the system and adaptively switch to a new configuration. The overall observer constructed is proved to be an asymptotic observer with bounded estimation error. This makes the proposed observer suitable in many practical control designs with fault tolerance or gain-scheduling. It should be noted that the main contribution of the paper lies on the claim of the possibility of constructing an overall observer based on individual observers, rather than each individual observer design.

A key restriction of this scheme is due to the assumption that each plant in the set is Lyapunov detectable. The successfully constructed switching observer requires the successful output injection design for each individual system in the system set. It is noted that the Lyapunov direct method-based observer design for nonlinear systems is in general a difficult problem. Therefore, the usefulness of the proposed method may be limited to some special applications.

References

[1] A. J. Van der Schaft, "On nonlinear observers," *IEEE Trans. Automat. Contr.*, vol. AC-30, 1986.

[2] M. Y. Fu and B. R. Barmish, *IEEE Trans. Automat. Contr.*, vol. AC-31, 1986.

[3] W. Hahn, *Stability of Motion*. New York: Springer-Verlag, 1967.

[4] P. V. Kokotovic, "The joy of feedback: Nonlinear and adaptive," *IEEE Contr. Syst. Mag.*, vol. 12, no. 3, 1992.

[5] S. R. Kou, D. L. Elliott, and T. J. Tarn, "Exponential observers for nonlinear dynamic systems," *Inform. Contr.*, vol. 29, 1975.

[6] Y. Liu, M. Cheng, and W. B. Gao, "A new control method and its robustness for a class of nonlinear systems," *J. Optimization Theory Appl.*, vol. 50, no. 1, 1988.

[7] Y. Liu and W. B. Gao, "Switching stabilization control for a set of time-varying nonlinear systems," in *Proc. IFAC Nonlinear Contr. Syst. Design Symp.*, Capri, Italy, 1989.

[8] D. E. Miller, "Adaptive stabilization using a nonlinear time varying controller," *IEEE Trans. Automat. Contr.*, vol. 39, p. 94.

[9] A. S. Morse, "Global stability of parameter-adaptive control systems," *IEEE Trans. Automat. Contr.*, vol. AC-25, no. 3, p. 433, 1980.

[10] K. Narendra, Y. H. Lin, and L. S. Valavani, "Stable adaptive controller design—Part II: Proof of stability," *IEEE Trans. Automat. Contr.*, vol. AC-25, p. 440, Mar. 1980.

[11] M. Vidyasagar, *Nonlinear Systems Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1978.

[12] K.-K. D. Young, "Design of variable structure model following control system," *IEEE Trans. Automat. Contr.*, vol. AC-23, p. 1079, 1978.

Optimal Scheduling with Deadline Constraints in Tree Networks

Partha P. Bhattacharya, Leandros Tassiulas, and Anthony Ephremides

***Abstract*—The problem of scheduling time-critical messages over a tree network is considered. Messages arrive at any of the nodes and have to reach the root node before their deadlines expire, else they are considered lost. The network is assumed to be operating in discrete time and the messages need one time unit for transmission from one node to the next along its path. The arrival and deadline processes are arbitrary. The policy which transmits messages with smallest extinction (arrival + deadline) time at every link is shown to minimize the number of lost messages over all time intervals and for every sample path.**

***Index Terms*— Optimal scheduling, real time communications, tree networks.**

I. Introduction

There are several applications of packet switched communication networks where a high variability in end-to-end packet delivery delay is undesirable. In packetized voice communication systems for example, the quality of the speech signal degrades considerably when the end-to-end delay exceeds a prescribed threshold. In networks carrying control information, a packet incurring a delay larger than the time within which the system state changes becomes useless for control purposes. An important problem in these systems concerns the design of network controls so as to minimize the number of packets reaching the destination after a prescribed threshold.

Queueing systems with impatient customers are appropriate models for communication networks with time-critical messages. Such systems have received moderate attention in the literature. In [1] and [2], the authors consider a first-come/first-served (FCFS) single server queue with customers that have deadlines on their waiting times. If the waiting time of a customer exceeds its deadline, the customer departs from the system and is considered lost. Performance measures such as the steady-state probability of loss of a customer and expected time until the rejection of the first customer were computed. In [3], [5], [7], and [8] the problem of optimal scheduling in a multiserver queue was considered, and the policy that serves the customer with smallest extinction time (arrival time + deadline) was shown to minimize the number of lost customers. In [6], the authors consider in-forest networks where customers are not lost and receive service even when their deadlines expire. The performance metric of interest in this case is customer lateness defined by the difference between departure time and deadline. The authors characterize policies that minimize customer lateness. Similar results for networks where customers are lost upon deadline expiration do not seem to be available in the literature.

In this work, we consider the problem of optimal scheduling of time-critical messages in a tree communication network. Messages are not transmitted further when their deadlines expire. Tree topologies arise often in subnetworks which are used for exchange of

Manuscript received November, 15, 1995.

P. P. Bhattacharya is with the IBM T. J. Watson Research Center, Hawthorne, NY 10532 USA.

L. Tassiulas is with the Electrical Engineering Department, University of Maryland, College Park, MD 20742 USA.

A. Ephremides is with the Electrical Engineering Department, University of Maryland, College Park, MD 20742 USA.

Publisher Item Identifier S 0018-9286(97)08833-8.

control and topology information among the nodes as well as for the concentration of this information to a central controller node. An important special case of such a network is of course a tandem connection. From the point of view of queueing theory, networks with tree topology are a first step toward an effort to generalize results for single queues. In Section II, the model of the tree network is described and the problem of optimal scheduling formulated. The optimality result is given in Section III.

II. Transmission Scheduling in a Tree Network

Consider a tree network with root node D, with L links between each pair of directly connected nodes. Messages arrive at any of the nodes and are destined for node D. There is a deadline and an extinction time (arrival time + deadline) associated with each message, and a message has to reach the destination before its extinction time expires. The deadline of the message becomes known upon arrival and can be used for scheduling its transmission. If the extinction time of a message expires while it is waiting or is being transmitted at an intermediate node, then the message is considered lost and need not be transmitted by the downstream nodes.

The system is assumed to be slotted. A message arriving during a slot is available for transmission only at the beginning of the next slot. The transmission time of a message is assumed to be one slot. The message arrival patterns and the deadlines are arbitrary.

We wish to determine a policy for scheduling the transmission of messages that minimizes the total number of lost messages. We assume that a node knows its distance (in number of hops) from the root node. It is evident that at slot t the optimal policy would never transmit a message with extinction time strictly less than $t + k$ at a node that is k hops away from D, as this message would be lost in any case. We say that a message at a node k which hops away from D is *eligible* for transmission at time t if its extinction time is at least $t + k$. Our result is that the policy which transmits the eligible messages with the shortest extinction time at every node minimizes the number of lost messages over any time interval. Following earlier work [5], [7], we will refer to this policy as the Shortest Time to Extinction (STE) policy. The policy can be trivially implemented in a distributed manner once a node knows its distance from the root node D. The optimality is proven in the next section.

III. Optimality Result

Let $L^{\pi}(t)$ denote the number of messages lost by time t when a message transmission scheduling policy π is applied. Fix arbitrary message arrival and deadline patterns.

Theorem 1: For every scheduling policy π

$$L^{\text{ste}}(t) \leq L^{\pi}(t) \qquad \forall t \geq 0.$$

The two assumptions made in the paper, namely the uniformity of the number of links across the network and identical destination D for every message, may appear too strict and it is natural to ask whether these assumptions can be relaxed. Before proceeding to the proof of the theorem, we present two examples that show the necessity of these assumptions for the theorem to hold.

Example 1: This example shows the necessity of the requirement that the number of links between any two (directly connected) nodes of the network be identical. Consider the tandem network of Fig. 1. Initially, there are two messages with extinction times two and three units at node A. Messages arrive to the system only in slot [0, 1); one to node A with extinction time three units and two to node B, each with extinction time two units. Let π be the policy which transmits the message with extinction time three units at node A at

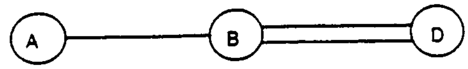

Fig. 1. A simple tandem network.

time zero and schedules according to the STE policy at all other times. It is easy to check that π loses one fewer message than STE.

Example 2: Suppose messages can have arbitrary destinations and the distance of the destination is known at every node and can be used for scheduling purposes. For a message with extinction time where e waiting at a node k hops away from its destination, define its residual time by $e - k$. A natural generalization of STE policy is then simply the policy that transmits messages with the smallest residual time (among the ones with nonnegative values) at every node. Denote this policy also by STE. The following example shows that STE need not be optimal. Consider again the tandem network of Fig. 1, but now there is one link between both A, B and B, D. Initially there are two messages at A, both destined for D with extinction times two and three. There are two arrivals in slot [0, 1): one to A with destination B and extinction time two and one to B with destination D and extinction time two. Let π denote the policy that transmits the message with extinction time three at node A at time zero and follows STE at all other times. It is easy to check that π loses one fewer message than STE.

Proof of the Theorem: The proof is based on an induction on the height of the tree and the idea of a dominating system. Denote the given tree by T. We say that a node of T is at *level* k if it is k hops away from the destination D. Let T_k denote the tree of height k obtained from T by deleting all the nodes at levels strictly larger than k. We will show by an induction on k that STE is optimal for every node in T_k. For the basis step, $k = 1$, observe that T_1 is nothing but a collection of independent links and therefore a collection of independent discrete-time multiserver queues with deterministic and identical service times but arbitrary arrivals and deadlines. Arguments identical to those in [3] and [5] can be given to show that STE is optimal for T_1.

For the induction step, suppose that STE is optimal for T_k. We show that STE is optimal for T_{k+1}. It follows from the induction hypothesis that STE is optimal at every node at level $1, 2, \cdots, k$ of T_{k+1} since the hypothesis holds for arbitrary arrivals and deadlines. It remains to show that STE is optimal at every node at level $k + 1$ in T_{k+1} when messages are being scheduled according to the STE rule at every node at level $1, 2, \cdots, k$. Further, it suffices to show this at any one node (say j) at level $k + 1$. Let i be the parent of node j.

Let A_t denote the number of arrivals to node j during slot $[t, t+1)$. Construct another tree $\tilde{T}_{k+1}$ that is identical to T_{k+1}, except that in $\tilde{T}_{k+1}$ node j is deleted and node i has an additional arrival process $\{B_t, t \geq 0\}$ where $B_0 = 0$, $B_t = A_{t-1}$, $t \geq 1$ (see Fig. 2). This simply means that arrivals to node j in T_{k+1} during slot $[t, t + 1)$ arrive directly to node i in $\tilde{T}_{k+1}$ during the next slot. In other words, $\tilde{T}_{k+1}$ is simply T_{k+1} with infinite links between nodes i and j. The scheduling policy at a node in $\tilde{T}_{k+1}$ is identical to that at the corresponding node in T_{k+1}. It is evident that $\tilde{T}_{k+1}$ dominates T_{k+1} in the sense that for any scheduling policy at node j in T_{k+1}, fewer messages are lost, over any time interval, in $\tilde{T}_{k+1}$ than in T_{k+1}. Therefore, the proof will be complete once we show that the number of lost messages in T_{k+1} equals that in $\tilde{T}_{k+1}$ when the STE policy is followed at node j in T_{k+1}.

Let S_t (respectively, $\tilde{S}_t$) denote the set of messages from node j (respectively, the arrival stream $\{B_t\}$) selected for transmission at time t at node i in T_{k+1} (respectively, $\tilde{T}_{k+1}$). We claim that $S_t = \tilde{S}_t$

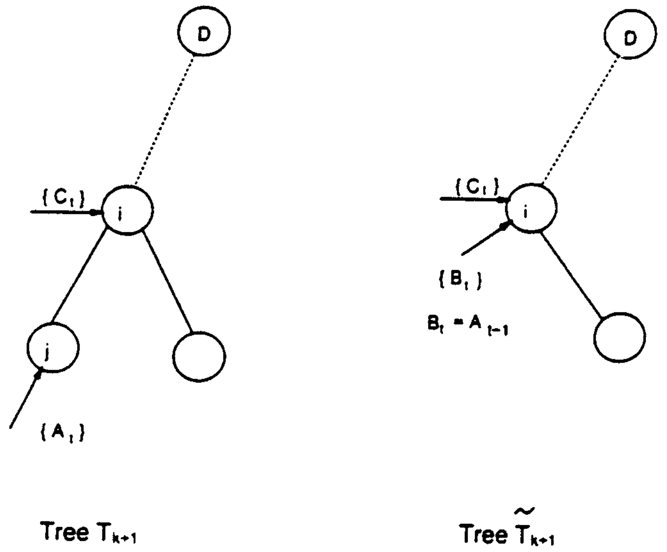

Fig. 2. Illustration of the proof.

for every $t \geq 0$ when STE is followed at node j in T_{k+1}. If the claim holds then the behavior of T_{k+1} and $\tilde{T}_{k+1}$ is identical and the theorem follows.

The proof of the claim is by contradiction. Let τ be the first time at which these two sets $\mathcal{S}_t$ and $\tilde{\mathcal{S}}_t$ differ. Observe that the set of messages from $\{B_t\}$ available for transmission at node i in $\tilde{T}_{k+1}$ is a superset, at all times, of those from node j available at node i in T_{k+1}. Therefore, the only way that $\mathcal{S}_\tau$ and $\tilde{\mathcal{S}}_\tau$ could differ is that there exists a message belonging to $\tilde{\mathcal{S}}_\tau \setminus \mathcal{S}_\tau$ that resides in node j in T_{k+1} at time τ. Let us denote this message by its extinction time e^*. Let t^* denote the arrival time of e^* in T_{k+1}. Since STE is followed at node j, for every t, $t^* \leq t \leq \tau - 1$, the messages transmitted at node j in T_{k+1} have extinction times e which satisfy $e - t \geq k + 1$ and $e < e^*$. Specifically, at time $\tau - 1$, L messages were transmitted, the extinction times, e, of which satisfy $e - \tau \geq k$ and $e < e^*$. However, all these messages must be waiting for transmission at time τ in $\tilde{T}_{k+1}$ since the sets $\mathcal{S}_t$ and $\tilde{\mathcal{S}}_t$ agree up to time $\tau - 1$. Since messages are scheduled according to the STE rule at node i in $\tilde{T}_{k+1}$, e^* cannot be chosen for service at time τ in $\tilde{T}_{k+1}$, a contradiction. □

References

[1] F. Baccelli, P. Boyer, and G. Hebuterne, "Single server queues with impatient customers," *Adv. Appl. Prob.*, vol. 16, pp. 887–905, 1984.

[2] F. Baccelli and K. S. Trivedi, "A single server queue in a hard real time environment," *Oper. Res. Lett.*, vol. 4, no. 4, pp. 161–168, 1985.

[3] P. P. Bhattacharya and A. Ephremides, "Optimal scheduling with strict deadlines," *IEEE Trans. Automat. Contr.*, vol. 34, pp. 721–728, 1989.

[4] P. P. Bhattacharya, L. Tassiulas, and A. Ephremides, "Optimal scheduling with deadline constraints in tree networks," IBM Tech. Res. Rep., RC 16348, 1990.

[5] P. P. Bhattacharya, "Real time scheduling problems with application to communication networks," Ph.D. dissertation, Electrical Engineering Department, University of Maryland, College Park, 1989.

[6] Z. Liu and D. Towsley, "Stochastic scheduling in in-tree networks," *Advances Appl. Probability*, vol. 26, pp. 222–241, 1994.

[7] S. S. Panwar, D. Towsley, and J. K. Wolff, "Optimal scheduling policies for a class of queues with customer deadlines to the beginning of service," *J. Assoc. Comput. Mach.*, vol. 35, no. 4, pp. 832–844, 1988.

[8] S. S. Panwar, "Time-constrained and multi-access communication," Ph.D. dissertation, Univ. Massachusetts, Amherst, 1985.

On Deadlock-Free Modular Supervisory Control of Discrete-Event Systems

Yonghua Li

***Abstract*— In this paper the problem of synthesizing deadlock-free modular supervisors for discrete-event systems is discussed. By introducing the d-invariant relation between a pair of supervisors, it is shown that when the control objective is described in terms of intersection of two languages, a necessary and sufficient condition for the modular supervisor to be deadlock-free is that this pair of subsupervisors satisfies a d-invariant relation. A procedure for synthesizing the deadlock-free modular supervisor is presented. Some issues concerning deadlock-free modular supervisory control are also discussed.**

***Index Terms*—Automata, deadlock, discrete-event systems, language, modular supervisory control.**

I. Introduction

A. The Problem

Modular supervisory control of discrete-event systems (DES's) has been studied by Ramadge and Wonham [8], [11]. It is an effective way to overcome computational complexity in control of DES's. By "modularity" we mean that the desired system behavior, i.e., the specification, is given in terms of a set of independent subspecifications. If one synthesizes the component subsupervisors independently and then merges them in the form of supervisor conjunction or disjunction, one gets the so-called modular supervisor. This kind of synthesis offers us the merits of lower computational and hardware requirements and the convenience of supervisor maintenance and redesign [11].

The main problem related to modular supervisory control is when the component subsupervisors have some desired properties, how to guarantee that the modular supervisor behaves in the same way. To illustrate this, let us take $L = L_1 \cap L_2$ to be the control task. A natural question is that if one computes $L^\uparrow$ and $L_i^\uparrow (i = 1, 2)$ (the supremal closed controllable sublanguage of L and $L_i (i = 1, 2)$, respectively), is it true that $L^\uparrow = L_1^\uparrow \cap L_2^\uparrow$? This problem, together with the problem of nonblocking modular supervision, has been discussed in [11] by defining a concept called nonconflicting language pairs. There are several other related results. In [6] sufficient conditions for checking blockings in decentralized supervisory control have been given. In [1] the computational problems that arise in nonblocking modular supervisory control and the supervisory control problem with blocking (SCPB) have been discussed.

Another problem related to modular synthesis is deadlock. This is the situation when the supervised system cannot evolve further (see next section for the definitions of deadlock-free languages and supervisors). Deadlock may occur when the modular supervisor is formed by supervisor conjunction, and even the component subsupervisors are all deadlock-free, as illustrated by the following example.

Manuscript received December 1, 1993; revised June 5, 1995. This work was supported by the NNSF of China under Grant 69104003.

The author is with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI 48202 USA (e-mail: yli@genius.eng.wayne.edu).

Publisher Item Identifier S 0018-9286(97)08832-6.