Neural Network Based Feedback Scheduler for Networked Control System with Flexible Workload

Feng Xia, Shanbin Li, and Youxian Sun

National Laboratory of Industrial Control Technology,
Institute of Modern Control Engineering, Zhejiang University,
Hangzhou 310027, China
{xia, sbli, yxsun}@iipc.zju.edu.cn

Abstract. Most control applications closed over a shared network are suffering from the time-varying characteristics of flexible network workload. This gives rise to non-deterministic availability of communication resources and may significantly impact the control performance. In the context of integrating control and scheduling, a novel feedback scheduler based on neural networks is suggested. With a modular architecture, the proposed feedback scheduler mainly consists of a monitor, a predictor, a regulator and an actuator. An online learning Elman neural network is employed to predict the network conditions, and then the control period is dynamically adjusted in response to estimated available network utilization. A fast algorithm for period regulation is employed. Preliminary simulation results show that the proposed feedback scheduler is effective in managing workload variations and can provide runtime flexibility to networked control applications.

1 Introduction

Today's control systems are representatively closed over a shared network of certain type, e.g. CAN (Controller Area Network), FF (Foundation Fieldbus), and ControlNet. Despite widespread employment in many fields like automotive electronics, process control, and robotics, control networks in real world are always bandwidth limited. The reasons behind may be economical and/or technical. Although the adoption of common-bus instead of point-to-point connections reduces system complexity of installation and maintenance, it also conceives the time-varying property of network workload. For a specific control loop, the availability of communication resources may change unexpectedly [1,2], due to changes in network user demands, or disturbances in the network environments such as the loss of a link. In addition, this may also arise from the alternative use of general-purpose networks like Ethernet for control applications. Consequently, the network QoS (Quality of Service) becomes unexpectedly changeable in such environments that feature workload uncertainty and may not be able to provide QoS requirements to a networked control application as needed. This could considerably impact the networked control systems [3], especially when the available communication resources are scarce.

Although many control techniques have been proposed to attack time delay effects [4], the performance of control applications over a network is still tied with the net-

L. Wang, K. Chen, and Y.S. Ong (Eds.): ICNC 2005, LNCS 3611, pp. 242 – 251, 2005.

work conditions regardless of the control algorithm used. It has been recognized that applying these control techniques on existing systems that are extensively being used in industrial plants could be costly, inconvenient, and time consuming. Moreover, as the real-world control applications become more and more complex, only compensating delay effects via controller design could not always guarantee satisfactory performance. While the system may perform unacceptably under overload conditions, certain communication resources may be wasted inadvertently in other cases. That is to say, traditional networked applications lack of flexibility, since they are usually designed regardless of the availability of communication resources.

In this paper, we introduce a novel methodology to address these problems, applying feedback scheduling methodology [5-8] to enable existing networked control applications in the presence of flexible workload. Emerging as a technique that integrates control and scheduling, feedback scheduling maps the methodology of feedback control to scheduling and provides a promising approach to manage uncertainty and enhance runtime flexibility. In this work, the well-established neural network (NN) technology [9] is employed to construct a feedback scheduler that is able to handle the workload variations intelligently. To achieve high efficiency as well as fast computation, we use a modified Elman network to learn from past and current network conditions and predict the future availability of network resources in order that the feedback scheduler can improve its behavior and respond in a pre-active fashion. If the workload abruptly increases, i.e. the QoS requirements cannot be provided as needed, the feedback scheduler will lower the required network utilization of a control loop and use the available network resources to perform the task as best as it can. In other cases when the workload decreases, the control performance will be upgraded to a maximum extent with the help of the feedback scheduler so that the available network resources are maximumly utilized. In this way, the feedback scheduler acts as an intelligent assistant to automate the management of flexible workload in NCSs.

The rest of this paper is structured as follows. Section 2 observes related works associated with our study. The architecture of the neural network based feedback scheduler is given in Section 3. We present the involved algorithms in Section 4. And its performance is evaluated via preliminary simulations in Section 5. Section 6 concludes this paper.

2 Related Works

An area that closely related to network QoS variations is congestion control. In this context, many mechanisms based on feedback methodology have been presented to manage network QoS. Feedback control technologies such as PID and fuzzy logic have been successfully used online to prevent the network from being congested [10]. When the aggregate demand for a bandwidth resource exceeds the available capacity of the network, they attempt to lower communication requirements of certain applications, thereby maintaining good network performance. Others employ neural networks to predict network traffic. Examples can be found in ATM [11], Internet [12] and other networks. However, almost all works in congestion control are not control related. Applications of neural networks can also be found in Internet based control systems, e.g. [13]. In these cases, neural networks are commonly employed for time delay forecast, which is different from the way we utilize them.

Researchers from the control community have made efforts to handle the impact of limited communication on control performance, for example, [14,15]. Most of these works focus on the design of control algorithms, and attempt to improve the system performance such as robustness with respect to uncertain communication delay. The methods used are static, i.e. they cannot provide run time adaptation to control tasks. In addition, these algorithms are often built upon simplified models of the complex characteristics of network workload variations. Instead of controller design, we focus on run time flexibility of control tasks. An interesting approach to providing networked control adaptation for network QoS variations is [1], where Chow and Tipsuwan propose a gain scheduling approach for networked DC motor control systems to compensate for the changes in QoS requirements.

Recent years witness considerable amount of attention on codesign of control and scheduling, both from the control community and the computing community. Several approaches to real-time QoS adaptation and graceful performance degradation in control applications are presented in the literature, e.g. [16,17]. A system' resource allocation is adjusted online in order to maximize the performance in certain respects. Feedback scheduling has been proposed as a promising methodology to increase flexibility and to master uncertainty with respect to resource allocation. Its applications for control purpose include optimal feedback scheduler [18] and its approximation versions [6], intelligent feedback schedulers [8,19] and those for anytime controllers [20,21]. However, these works are dedicated to co-design of control and CPU scheduling, while the main concern of this paper is control over networks with flexible workload.

In order to achieve dynamic integration of control and network scheduling, several methods have been proposed in the context of NCS. For example, Branicky et al [22] propose a co-design approach to the treatment of both network and controlled systems issues, where a set of control loops are optimally scheduled. Park et al [23] present a scheduling method for NCSs to adjust the sampling period as small as possible, allocate the bandwidth of the network for three types of data, and exchange the transmission orders of data for sensors and actuators. In [24], the allocation of bandwidth to control loops is done locally at run time according to the state of each controlled process, and control laws are designed to account for the variations on the assigned bandwidth. The methods employed in these works are reactive in the sense that they will only adjust the communication resource requirements of control applications once the network is already overloaded. We attempt to develop a pre-active approach to flexible quality of control (QoC) management with respect to network workload variations. A more detailed survey with additional references related to real-time scheduling in networked and embedded control systems can be found in [25].

3 Feedback Scheduling Architecture

The system we attempt to deploy the feedback scheduler into is a control loop sharing a network with other communication entities. The workload within the network may vary over time, and hence the available network utilization for this control application is non-deterministic. Generally, a traditional controller is designed offline regardless of workload variations. From the control perspective, the control performance may be degraded, or even destabilized due to the uncertain delays stemming from scarcity of network resources. From the scheduling perspective, the network resources may be under-utilized. To address these problems, both control and scheduling are synthetically considered. Following the methodology of feedback scheduling [5-8], a NN based feedback scheduler (see Fig.1) is proposed to maximize the control performance via maximum use of available network resources.

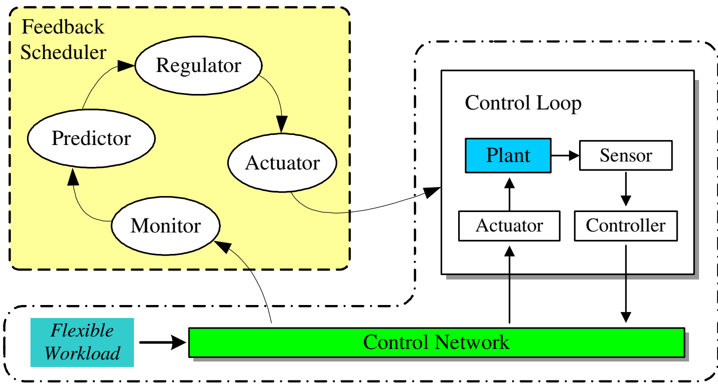

Fig. 1. Architecture of a networked control system equipped with a feedback scheduler

According to real-time scheduling theory, the requested network utilization can be calculated as $U_r = c/h$, where c is the transmission time and h is the transmission period, which is assumed to be equal to the sampling period of the control loop (also called control period). In order to comply with the CPU scheduling theory, we choose c to denote the time needed for transmitting the control loop's data/message in each sampling interval under the condition where total network bandwidth is exclusively dedicated to the considered control application, i.e. no other applications consuming communication bandwidth exist. It is assumed that the size of the data/message to be transmitted remains constant all the time. Therefore, the value of c does not change. For example, for a control network with a data rate of 1 Mb/s, if the data size is 500 bytes, the transmission time $c = 500 \times 8 / 10^3 = 4$ ms. And then, the requested network utilization U_r will only depend on the control period h. It is well-known in digital control theory that smaller sampling period leads to better control performance. In the case of resource constraints, however, this is not always the fact [3], and the sampling period should be determined properly so that the message is schedulable, i.e. the resource constraints cannot be violated. To reflect the uncertain characteristic of the workload, we use U to denote the available network utilization for the control loop, which is time varying and naturally bounded in the range [0, 1]. As the total bandwidth is fixed, U will be determined directly by current workload. Intuitively, larger U represents light workload while smaller U corresponds to heavy workload. In this way, the workload uncertainty can be reflected by unexpected changes in U.

Then, the problem of managing workload uncertainty can be stated as dynamically adjusting the sampling period with respect to the variations of U so that the tradeoff between control performance and available communication resource is achieved. As shown in Fig.1, the feedback scheduler introduced to address this problem mainly consists of four components: a monitor, a predictor, a regulator, and an actuator. The *monitor* interfaces with the control network and measures the variations in the network condition. It works periodically in time-driven fashion to activate the feedback scheduler. The *predictor* is built using neural networks and responsible for making prediction of the next-to-come value of the available network utilization. This enables the feedback scheduler to work in an intelligent way based on the history knowledge about the workload variations and to act in advance. Furthermore, the online learning capability of neural networks ensures the validity and efficiency of the feedback scheduler even when the characteristic behavior of workload variations changes. The *regulator* makes its decision based on the prediction of the neural predictor. It performs the role of determining a new sampling period to maximize the control performance under the predicted network resource constraint. The *actuator* within the feedback scheduler acts as an interface with the control loop to adjust its sampling period. To reduce sampling period jitters, an invocation condition is introduced in the actuator. The control period will be updated only when the absolute difference between the current value and the newly produced one exceeds a pre-specified deadband/threshold.

From the viewpoint of feedback control, the feedback scheduler can be regarded as a NN prediction based controller. The controlled variable is the network utilization, and the manipulated variable is the control loop's period. With flexible workload, the problem of feedback scheduling is similar to some kind of trajectory tracking issues, which are familiar to control engineers.

4 Algorithms

In this section, we present the algorithms utilized in the feedback scheduler. Particular emphasis is on two major components, the predictor and the regulator.

4.1 Neural Predictor

In the predictor, a neural network is employed to model the complex characteristics of the variations in available network utilization and estimate in real-time the next U value. To meet the timing constraints, we use a modified Elman network [9] (given in Fig.2) because of its simple structure, fast computation, and dynamic memory capability. There are two inputs, the current available network utilization $U(k)$ and its previous value $U(k-1)$, and one output, the predicted value of available network utilization at the next sampling instance, i.e. $\hat{U}(k+1)$, where k denotes sampling instances of the *monitor* within the feedback scheduler. Note that the sampling interval of the monitor is different from that of the control loop. The number of the hidden nodes is chosen to be 3. The squared error between the predicted and actual values, i.e. $\hat{U}(k+1)$ and $U(k+1)$ is chosen as the performance index for the online training operation at the $(k+1)^{\text{th}}$ instance.

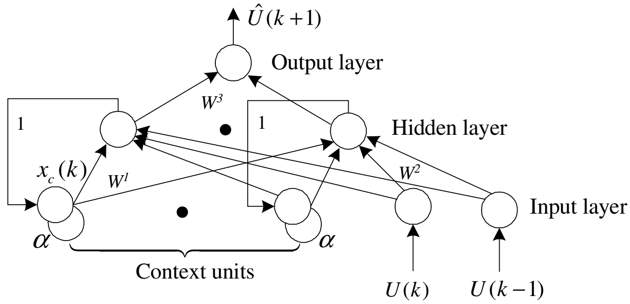

Fig. 2. Neural predictor structure

According to the neural network structure (Fig. 2), it is held that:

$$\hat{U}(k+1)=\sum_{l=1}^{3}W^3{}_l(k)f(\sum_{j=1}^{3}W^1{}_{qj}(k)x_{c,q}(k)+\sum_{i=1}^{2}W^2{}_{ij}(k)I_i(k)-T_{h,j}(k))-T_o(k) \tag{1}$$

where $x_c(k)=\alpha x_c(k-1)+x(k-1)$, $I_1(k)=U(k)$, $I_2(k)=U(k-1)$, $f(x)=1/(1+e^{-x})$ is the activation function, T_h and T_o are biases in the hidden layer and the output layer, respectively, and W^1, W^2, W^3 are connection weights.

In this work, the standard BP algorithm is employed for the network training. Thus the weights and biases of the network can be updated at each sampling instance according to:

$$\begin{aligned}
\Delta w_j^3 &= \eta\delta^o x_j(k) && j=1,2,3\\
\Delta w_{ij}^2 &= \eta\delta_j^h I_i(k-1) && i=1,2\\
\Delta w_{qj}^1 &= \eta(\delta^o W^3{}_j)\frac{\partial x_j(k)}{\partial w^1{}_{qj}} && q=1,2,3
\end{aligned} \tag{2}$$

where $\frac{\partial x_j(k)}{\partial w_{qj}^1}=f'_j(\bullet)x_q(k-1)+\alpha\frac{\partial x_j(k-1)}{\partial w_{qj}^1}$, $\delta^o=(y_d(k)-y(k))g'(\bullet)$, $g(\bullet)$ is set to be the *purelin* function, $\delta_j^h=(\delta^o w^3{}_j)f'_j(\bullet)$, $f'(\bullet)$ and $g'(\bullet)$ are derivatives of $f(\bullet)$ and $g(\bullet)$, respectively.

4.2 Regulator

According to the data/information flow inside the feedback scheduler, the output of the predictor, i.e. the estimated value of the available network utilization $\hat{U}(k+1)$ will be forwarded into the regulator. Based on this input and its internal knowledge about the transmission time of the control loop, the regulator attempts to determine an optimal sampling period. In the newly emerging field of feedback scheduling, it has been revealed [6,18] that a simple rescaling of the sampling period could bring almost optimal solutions in most cases. Therefore, in order to achieve timely responses, we use the following simple and fast calculation to determine the control period:

$$h(k) = \frac{U_0}{\hat{U}(k)} h_0 \tag{3}$$

where h_0 is the nominal period, and U_0 is the nominal requested network utilization, which satisfies $U_0 = c/h_0$. Let's take a look at the control period $h(k)$ deduced from equation (3) from the viewpoints of both control and scheduling. Firstly, on the behalf of scheduling, the available network resources will be maximumly used, since the requested utilization $U(k) = c/h(k) = h_0U_0/h(k) = \hat{U}(k)$. Given that the available network utilization is accurately predicted, there will be no waste of resources, while the constraints are properly met. Secondly, for control purpose, the minimum control period under current resource constraint is obtained because any h value smaller than the one in (3) will result in a requested utilization more than the network can provide. Therefore, the control performance will be maximized using this *regulator*.

Once produced, the new sampling period $h(k)$ will be put forward to the actuator as its input. And then, it will be decided whether to take an action to update the control period according to the invocation condition inside the actuator.

Obviously, there are two important design parameters within the monitor and the actuator, with one for each. The first one is the sampling interval of the monitor. It is used to determine how often the network condition is sampled. Intuitively, smaller intervals allow high sensitivity while being more resource consuming, and vice versa. The second one is the deadband utilized in the actuator. It is originally employed to avoid too frequent refreshing operations on the control period, which inversely degrades the control performance. Similarly, a smaller deadband leads to more accurate response to workload variations while being more sensitive to noises, and vice versa. Without formal design methodology for these parameters, tradeoffs should be done when implementing the feedback scheduler in practical applications.

5 Performance Evaluation

In this section, we evaluate the performance of the proposed feedback scheduler through considering networked control of a plant over a CAN-bus. The controlled plant is given as $G(s) = 1000/(s^2 + s)$. The transmission time c is assumed to be 4 ms. With nominal network utilization U_0 = 0.5 provided, a traditional PID controller is well-designed pre-runtime to control the plant. The nominal control period h_0 = 8 ms. During run time, the available network utilization varies as shown in Fig. 3 (dashed magenta line), reflecting characteristic behaviors of flexible workload. The following two cases are simulated.

In Case I, the controller works with a fixed sampling period of 8 ms all along. Therefore, the requested network utilization remains changeless, see dash-dot red line in Fig. 3. In the time interval t = 0 to 1 s, because the available network utilization vibrates around 0.5, the control performance is slightly impacted, as shown in Fig. 4 (dashed red line). Still, the performance is satisfactory. From t = 1 to 2s, the plant performs well thanks to good availability of network resources. However, a lot of network resource is wasted. From time t = 2 s, the control system turns to be unstable because the available network utilization falls below the requested value of 0.5.

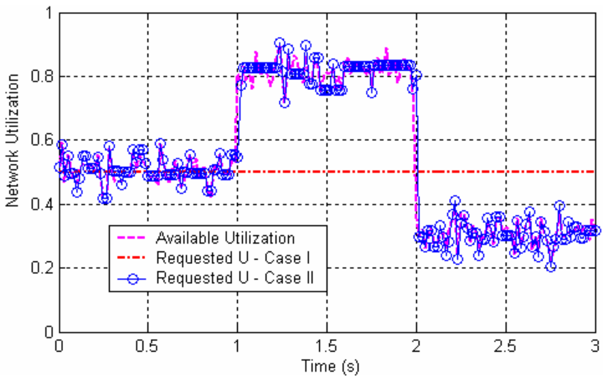

Fig. 3. Network utilization

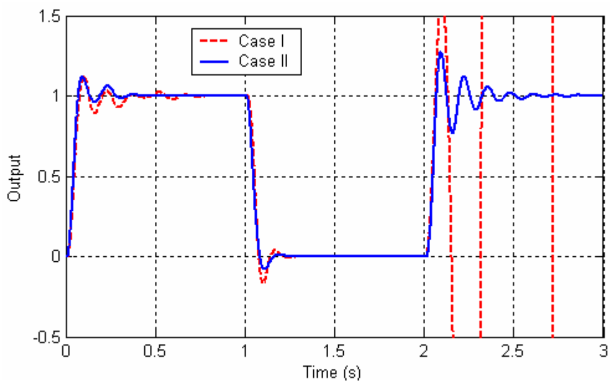

Fig. 4. Transient responses of the plant

In Case II, the NN based feedback scheduler we present is implemented. The sampling interval of the *monitor* is chosen as 15 ms. The deadband in the *actuator* is set to be 0.5 ms. The PID parameters are online updated to compensate jitters of the control period, allowing us to concentrate on the effectiveness of the proposed feedback scheduler. The requested network utilization is illustrated in Fig. 3 (solid blue line with circles). The system response is also plotted in Fig.4 (solid blue line). As we can see, the control performance is improved all along the simulation. This is especially the truth when the available utilization goes below U_0 from t = 2s. With the help of the NN based feedback scheduler, the plant exhibits satisfactory performance even when the network resources become scarce. In response to workload variations, the feedback scheduler attempts to maximize the control performance through maximizing the use of available network resources in an intelligent way all the time, as shown in Fig. 3 and 4. This mainly benefits from the powerful capability of the Elman NN for effectively predicting the network conditions.

6 Conclusions

Control applications built upon a shared control network must meet increasingly demanding requirements to cope with significant degrees of workload uncertainty, especially when the communication resource is limited. These requirements give rise to the integration of feedback control and network scheduling. In this paper, we demonstrate a novel application of neural networks in the newly emerging field of feedback scheduling. In order to handle flexible workload in control networks, we present a feedback scheduler based on a neural predictor. The primary goal is to maximize the quality of control under constraints on the availability of network resources. It is argued that this feedback scheduler allows the control application to be highly flexible with respect to complex workload variations, while improving the control performance to the maximum extent. As a future work in this direction, we will attempt to develop effective control algorithms to compensate the sampling period jitter, which seems to be the main problem in the proposed approach.

References

1. M.-Y. Chow and Y. Tipsuwan: Gain Adaptation of Networked DC Motor Controllers Based on QOS Variations. IEEE Trans. on Industrial Electronics 50:5 (2003) 936-943
2. Feng Xia, Xiaohua Dai, Zhi Wang, and Youxian Sun: Feedback Based Network Scheduling of Networked Control Systems, in Proc. 5th ICCA, Budapest, Hungary (2005)
3. Feng Xia, Zhi Wang, and Youxian Sun: Integrated Computation, Communication and Control: Towards Next Revolution in Information Technology, LNCS, Vol. 3356, Springer-Verlag (2004) 117-125
4. Y. Tipsuwan and M.-Y. Chow: Control methodologies in networked control systems. Control Eng. Practice 11:10 (2003) 1099-1111
5. C. Lu, J. Stankovic, G. Tao, S.H. Son: Feedback control real-time scheduling: framework, modeling, and algorithms, Real-time Systems 23:1/2 (2002) 85-126
6. A. Cervin, J. Eker, B. Bernhardsson, K.-E. Årzén: Feedback-Feedforward Scheduling of Control Tasks, Real-Time Systems 23:1 (2002) 25-53
7. L. Abeni, L. Palopoli, G. Lipari, J. Walpole: Analysis of a Reservation-Based Feedback Scheduler, in Proc. 23rd IEEE RTSS, Austin, Texas (2002) 71-80
8. Feng Xia and Youxian Sun: Neural Network Based Feedback Scheduling of Multitasking Control Systems, to appear in Proc. KES2005, LNCS, Springer-Verlag (2005)
9. Zengyin Sun, Zaixing Zhang, Zhidong Deng: Intelligent Control Theory and Technology, Beijing: Tsinghua University Press (1997) (in Chinese)
10. S. Ryu, C. Rump, C. Qiao: Advances in Internet Congestion Control. IEEE Communications Surveys and Tutorials 5:1 (2003) 28-39
11. Y.-C. Liu, C. Douligeris: Rate Regulation with Feedback Controller in ATM Networks - A Neural Network Approach, IEEE Journal on Selected Area in Communications 15:2 (1997) 200-208
12. A. Bhattacharya, A. Parlos, A. Atiya: Prediction of MPEG-Coded Video Source Traffic Using Recurrent Neural Networks, IEEE Trans. on Signal Processing 51:8 (2003) 2177-2190
13. D. Liu, J. Du, Y. Zhao, N. Song: Study on the Time-Delay of Internet-based Industry Process Control System, in Proc. 5th WCICA, Hangzhou, China, Vol.2 (2004) 1376-1380

14. L. Xiao, M. Johansson, H. Hindi, S. Boyd, A. Goldsmith: Joint Optimization of Communication Rates and Linear Systems, IEEE Trans. on Automatic Control 48:1 (2003) 148-153
15. Shanbin Li, Yong-Yan Cao, Yongqiang Wang, Youxian Sun: Robust H_∞ Control of Uncertain Markovian Jump Systems with Mode-Dependent Time-Delays, in Proc. 16th IFAC World Congress, Prague, Czech Republic (2005)
16. T. Abdelzaher, E. Atkins, and K. Shin: QoS Negotiation in Real-Time Systems and Its Application to Automated Flight Control, IEEE Trans. on Computers 49:11 (2000) 1170-1183
17. G. Beccari, S. Caselli, M. Reggiani, F. Zanichelli: Rate modulation of soft real-time tasks in autonomous robot control systems, in Proc. 11th ECRTS, York, England (1999) 21-28
18. J. Eker, P. Hagander, K.-E. Årzén: A feedback scheduler for real-time controller tasks. Control Engineering Practice 8:12 (2000) 1369-1378
19. Feng Xia, Liping Liu, and Youxian Sun: Flexible Quality-of-Control Management in Embedded Systems Using Fuzzy Feedback Scheduling, to appear in Proc. RSFDGrC, LNCS, Springer-Verlag (2005)
20. Feng Xia and Youxian Sun: Anytime Iterative Optimal Control Using Fuzzy Feedback Scheduler, to appear in Proc. KES2005, LNCS, Springer-Verlag (2005)
21. D. Henriksson, A. Cervin, J. Åkesson, K.-E. Årzén: Feedback Scheduling of Model Predictive Controllers, in Proc. 8th IEEE RTAS, San Jose, CA (2002) 207-216
22. M. Branicky, S. Philips, and W. Zhang: Scheduling and feedback co-design for networked control systems, in Proc. 41st IEEE CDC, Las Vegas (2002) 1211-1217
23. H. Park, Y. Kim, D-S. Kim, W.H. Kwon: A Scheduling Method for Network-based Control Systems, IEEE Trans. on Control System Tech. 10:3 (2002) 318-330
24. M. Velasco, J. Fuertes, C. Lin, P. Marti, S. Brandt: A Control Approach to Bandwidth Management in Networked Control Systems, in Proc. 30th IEEE IECON, Busan, Korea (2004)
25. Feng Xia, Hongxia Yin, Zhi Wang, Youxian Sun: Theory and Practice of Real-time Scheduling in Networked Control Systems, in Proc. 17th Chinese Control and Decision Conference, Harbin, China (2005) (in Chinese)