Issues on Communication Network Scheduling using Numerical Simulations

Oscar Esquivel-Flores and Héctor Benítez-Pérez
Universidad Nacional Autónoma de México
México

1. Introduction

Nowadays, industry has successfully used Network Control Systems (NCS) therefore several lines of research have arisen. A NCS is a current application of a Real-Time Distributed Systems (RTDS), composed of a number of nodes capable of developing a complete control process. In this systems several nodes exchange information through a communication network to achieve specific control goals, nevertheless network traffic increases. This affects the overall system performance. Several approaches have been developed to satisfy requirements of both control and communication performance. Particularly, some methodologies focus on saving bandwidth, one of such methodologies is network scheduling. The objective of this methodology is the accurately use of the computing resources.
Network scheduling deals with to elect a sampling rate , aiming to reduce the number of data transmitted over the network . The effectiveness of the control system depends on such a sampling rate (Lian et al., 2001; 2002; 2003). A region is acceptable in networked control performance terms if it is contained within two sampling rate boundaries, which can be statistically determined.
The use of a common-bus network architecture and a particular network protocol introduces different forms of time delay uncertainties between, sensors, actuators and controllers. Hence, it is quite important to explore different network protocols and network scheduling strategies, before to implement the RTDS, in order to obtain a desired control performance.
The main objective of this work is to expose the use of numerical simulations to explore several issues on communication network scheduling of a RTDS, besides to implement a particular network scheduling strategy to evaluate its effectiveness. This is presented using a simulated case study, based upon a 2-DOF helicopter simulation benchmark (Quanser, 2006). This simulation provides an approximation to system response, in which, for demonstration purposes, the main results are obtained for a typical fault scenario. Thus, for this simulation, a scheduling strategies is implemented using TrueTime (Cervin et al., 2003) performing dynamic scheduling.

2. Real-Time Simulation Tool TrueTime

This section gives an introduction to the TrueTime simulator and briefly exposes a basic example on how TrueTime can be used to simulate a RTDS. TrueTime (Cervin et al., 2003; 2007; Henriksson, Cervin, Andersson & Årzén, 2006; Henriksson, Redell, El-Khoury, Cervin, Törngren & Årzén, 2006) is a simulator for networked and embedded control system based on

Matlab/Simulink, it has been developed at Lund University since 1999. The simulator software consists of a Simulink block library, the kernel block simulates a Real-Time kernel executing user-defined task and interrupt handlers. To communicate kernel blocks (nodes) several network blocks may be used, thus it makes quite simple to develop networked control simulations. TrueTime kernel block simulates a computer node with a generic real-time kernel, A/D and D/A converters, and network interfaces. The block is configured via an initialization script, in this script the programmer may create task, timers, interrupt handlers, semaphores, etc., it means the software executing in the computer node. The kernel periodically calls the code functions of the tasks and interrupt handlers. The initialization script and the code functions may be written in either Matlab code or in C++. TrueTime Kernel block supports several scheduling algorithms such as fixed-priority scheduling and earliest-deadline-first scheduling. The TrueTime Network block simulate the physical layer and the medium-access layer of several local-area networks. CSMA/CD Ethernet), CSMA/AMP (CAN), Round Robin (Token Bus), FDMA, TDMA (TTP), Switched Ethernet, WLAN (802.11b), and ZigBee (802.15.4) are some types of network supported. The network blocks are mainly configured using blocks dialogues. Some parameters common to all types of networks are bit rate, the minimum frame size, and the network interface delay. Each type of network has a number of parameters that can be specified. The network blocks may be used having one kernel block for each node in the network. The tasks inside the kernels can then send and receive arbitrary Matlab structure arrays over the network using certain kernel primitives. This way is to quite flexible but requires to program some routines to configure the system.

3. A Real-Time Distributed System

This section exposes the RTDS used for implementation purposes, it is a 2-DOF helicopter prototype (Quanser, 2006). The following section briefly introduces and describes this 2-DOF helicopter prototype and its controller design. The 2-DOF Helicopter system is a prototype with two propellers driven by DC motors, and integrating a CanBus network. The front propeller controls the elevation of the helicopter nose on the pitch axis, and the back propeller controls the side-to-side movement on the yaw axis. These pitch and yaw angles describe the state of the helicopter, and are measured using high-resolution encoders. The dynamics of the model are based on kinetic and potential energy, and it is used for the design of a position controller: the helicopter center of mass is described in xyz cartesian coordinates regarding the pitch (θ) and yaw (ψ). Regarding control issues, two controllers may be used: a FF-LQR and a FF+LQR+I. The FF+LQR regulates the pitch axis of the helicopter, using feed-forward (FF) and proportional-velocity (PV) compensators, while the yaw axis only makes use of a (PV) to control. The FF+LQR+I controller uses an integrator in the feedback loop to reduce the steady-state error, by a feed-forward and proportional-integral-velocity (PIV) algorithms to regulate the pitch, and only a PIV to control the yaw angle. This work focuses on the FF+LQR+I controller.

4. Experimental Approach

In order to study the impact of network utilization on closed control loop, the 2-DOF Helicopter control model is built as a RTDS. Several nodes are connected through a common communication network. The experiment focuses on network scheduling, and the main objective is to balance the amount of data sent through the network, in order to avoid latency and under sampling. Two network scheduling proposals are given to explore several aspects in control performance when the use of the network exceeds network bandwidth. To have a performance criteria, and thus, quantify the system's quality performance, the integral of the absolute value of the error IAE is used:

$$IAE = \int_{t_0}^{t_f} |e|\, dt \;\; or \;\; IAE = \sum_{k=k_0}^{k_f} |e_k| \tag{1}$$

where $t_0(k_0)$ and $t_f(k_f)$ are the minimum and maximum time of the evaluation period, whether a continuous or discrete time is used, and e represents the error between the actual and reference trajectories. The actual RTDS for the experiment consists of 8 processors. These real-time kernel processors and the network are simulated using TrueTime (Cervin et al., 2007). The network used is a CSMA/AMP (CAN) with a transmission rate of 80000 bits/second, and not data loss. Four sensor nodes execute periodic tasks to sense control signals, as well as other additional periodic tasks. Each task has a period p_i and time consumption c_i. The sensed control signals are θ, ψ, $\dot{\theta}$, $\dot{\psi}$. This model has a controller node, two actuator nodes and scheduler node. Each node initializes, specifying the number of inputs and outputs of the respective TrueTime kernel block, defining a scheduling policy, and creating periodic tasks for the simulation.

5. Network Scheduling based on Frequency Transmission

This section develops an approach that modifies the frequency transmission using tree parameters: minimum frequency rate (f_m), real frequency rate (f_r) and maximum frequency rate (f_x). The RTDS dynamics, then, is modeled as a linear time-invariant system, whose state variables are transmission frequencies$(f_i = \frac{1}{p_i})$ of the n nodes that compose the RTDS (Esquivel-Flores et al., 2010). Frequency rates of a node are affected by some external input frequency rates, minimal frequencies of all nodes and particular ratios serve as coefficients of the linear system. So, it is possible to control the NCS using the input vector u, such that the output vector y contains the frequency rates of all nodes within a nonlinear region L, bounded by the maximum and minimum transmission frequency rates.

6. Numerical Simulations

In this section numerical simulations using the Network scheduling strategy based on Frequency Transmission are exposed. The network scheduling strategy dynamically adjusts the frequencies, considering the participation of several nodes of the RTDS. Numerical simulations shows that the dynamical changes of this strategy improve the RTDS response under fault scenarios. These scheduling approach show a way to manage the network resources, especially with a limited network bandwidth . These techniques avoid network delays during transmission.

7. Conclusions

This work show a study of network scheduling strategy using numerical simulations. A simulation of a particular RTDS a 2-DOF helicopter is built using TrueTime as real-time simulation tool. A network scheduling strategy based on changes of frequency transmission rates is implemented in order to expose the advantages of using dynamic scheduling in an ad-hoc implementation for the network of a NCS. The use of numerical simulations aid to explore several considerations in design and analysis of a NCS.

8. References

Cervin, A., Henriksson, D., Lincoln, B., Eker, J. & Årzen, K.-E. (2003). How does control timing affect performance? analysis and simulation of timing using jitterbug and truetime, *Control Systems, IEEE* **23**(3): 16 – 30.

Cervin, A., Ohlin, M. & Henriksson, D. (2007). Simulation of networked control systems using truetime, *Proc. 3rd International Workshop on Networked Control Systems: Tolerant to Faults*, Nancy, France.

Esquivel-Flores, O., Benítez-Pérez, H., E., M.-M. & Menéndez, A. (2010). Efficient overloading techniques for primary-backup scheduling in real-time systems, *ICI Express Letters Part B: Applications* **1**(1): 93–98.

Henriksson, D., Cervin, A., Andersson, M. & Årzén, K.-E. (2006). Truetime: Simulation of networked computer control systems, *Proceedings of the 2nd IFAC Conference on Analysis and Design of Hybrid Systems*, Alghero, Italy.

Henriksson, D., Redell, O., El-Khoury, J., Cervin, A., Törngren, M. & Årzén, K.-E. (2006). Tools for real-time control systems co-design, *in* H. Hansson (ed.), *ARTES – A network for Real-Time research and graduate Education in Sweden 1997–2006*, Department of Information Technology, Uppsala University, Sweden.

Lian, F.-L., Moyne, J. & Tilbury, D. (2001). Analysis and modeling of networked control systems: Mimo case with multiple time delays, *American Control Conference, 2001. Proceedings of the 2001*, Vol. 6, pp. 4306 –4312 vol.6.

Lian, F.-L., Moyne, J. & Tilbury, D. (2002). Network design consideration for distributed control systems, *Control Systems Technology, IEEE Transactions on* **10**(2): 297 –307.

Lian, F.-L., Yook, J., Otanez, P., Tilbury, D. & Moyne, J. (2003). Design of sampling and transmission rates for achieving control and communication performance in networked agent systems, *American Control Conference, 2003. Proceedings of the 2003*, Vol. 4, pp. 3329 – 3334 vol.4.

Quanser (2006). *Quanser 2 DOF Helicopter. User and Control Manual*, Quanser. Innovate-Educate.