TUNING OF INDUSTRIAL CONTROLLERS OVER PUBLIC IP NETWORKS

Renato F. Fernandes Jr, Dennis Brandão
University of São Paulo, Av.Trabalhador Sãocarlense 400, São Carlos, SP, Brazil

Nunzio M. Torrisi
Universidade Federal do ABC, Rua Santa Adélia 166, Santo André, SP, Brazil

Keywords: PID control, Identification, Industrial systems, Remote control, Industraial networks.

Abstract: Tuning of industrial systems is executed in the initial phase of the system and mainly during the maintenance phase, providing characteristics of the performance of the industrial process during production life cycle. Remote tuning supports several practical applications, such as specialized companies outsourcing services or companies distributed in different areas centralizing optimization. This paper proposes a software tool for remote tuning of open or closed PID control loops in an industrial environment that fulfils the requirements described above, in a single platform. The software tool could be used in control loops tuning in industrial systems, as well as in an academic environment simulating control applications and industrial networks.

1 INTRODUCTION

Increasing competitiveness in the industrial sector has required continuous improvement in product quality processes, optimizing the production and reducing operational costs. Nowadays, researches like (Avov, 2004) show the potential of the Internet in the industrial environment. However, considering the use of the Internet for control and supervision of industrial processes, it should be noticed that the nature of production and automation systems demands some requirements to be secured, such as managing multiple accesses, ensuring the communication and the control system, setting maximum periods for process data updates and quality of service maintenance (Abdelzaher, 2002).

Remote access architectures may be implemented at different levels in the control hierarchy: at process level, at supervisory level and at system optimization level (Yang, 2003). In terms of process, the proposal is to include remote control within the process control loop, according to (Overstreet, 1999) (Yang, 2007). For this purpose, the conventional discrete control structure should be altered to conform to the Internet's variable (Luo, 2000), (Yang, 2007).

The concern at the supervisory level is safety and quality of service, discussed in (Kunes, 2001). Yang et al. (Yang, 2003) proposes a remote control at supervision level for services that are independent of the Internet delay, which would be restricted to acyclic services such as tuning parameters for PID block and set points.

In the context of SCADA (Supervisory Control and Data Acquisition) systems, the OPC (OLE for process control) technology (OPC Foundation, 2006) combined with Web technologies, as WebServices, is used to draw complex architecture for manufacturing in order to create communication system directly between shop-floor and decentralized supervision systems (Zheng and Nakagawa, 2001).

Commercial companies offer today solutions for remote monitoring and tuning of industrial systems. However, these solutions have some limitations, because they may be based on non-standard platforms for the industrial environment and may use common WebServices (Calvo, 2006) (Batur, 2000) (Qin, 2007).

Torrisi (Torrisi, 2007) proposes a standard OPC communication mechanism based on the Internet, a platform-independent alternative to WebServices. That standard, called CyberOPC, uses "open"

F. Fernandes Jr R., Brandão D. and M. Torrisi N. (2009).
TUNING OF INDUSTRIAL CONTROLLERS OVER PUBLIC IP NETWORKS.
In *Proceedings of the 6th International Conference on Informatics in Control, Automation and Robotics - Intelligent Control Systems and Optimization*, pages 278-284
DOI: 10.5220/0002218002780284

security technologies for light software components, therefore providing better performance and increasing guarantees for data security, when compared to other technologies based on WebServices.

This paper proposes an architecture to execute remote tuning of industrial control systems using the Internet, fulfilling acceptable security and performance requirements. In order to validate the architecture, a software application using CyberOPC and called Cybertune will be presented. The validation consists of a model-based identification and tuning using first-order-plus-dead-time systems.

This paper is organized as follows: Section 2 shows the requirements for remote tuning and problems related to supervision and remote control of industrial systems. Section 3 presents the generic architecture of the remote tuning system. Section 4 describes tests and results from remote tuning, and finally section 5 presents the conclusions and indicates future researches.

2 REMOTE TUNING REQUIREMENTS

In control loop tuning, the system identification phase demands over half of the effort (Yu, 2006), (Hjalmarsson, 2005). In order to obtain the model in time domain, the most common way to identify low order systems is identifying the transient response to a process alteration. The transient response is found when stimulating the system using a step-format signal, an impulse or a pseudo-random binary sequence (PRBS) as a system input. Identification is executed by estimating parameters through data collection and expression using ARX (Auto-Regressive Exogenous) or ARMAX (Auto-Regressive Moving Exogenous) model (Aguirre, 2004) (Ljung, 1999).

Controllers are tuned in three phases, according to model-based techniques: identifying the plant model, tuning based on the identified model, and validating tuning using simulations based on the identified model with the new controller.

An important step in the model-based identification consists of defining the sampling rate, which must be constant and generally in the order of ten times smaller than the time constant of the system (Aguirre, 2004).

In general, the OPC technology has an update rate defined in seconds for processes with fieldbus technology. This rate reduces our scope to slower systems with time constants and dead time 5 to 10 times of the time of the OPC acquisition, although many industrial systems may fit this scenario, such as temperature control and chemical processes.

2.1 Communication Problems related to Remote Control and Monitoring

The Internet and WebServices available nowadays create some obstacles when used in industrial control systems (Abdelzaher, 2002), (Yang, 2003), (Torrisi, 2007):

- Communication delay of different types, throughout the data source up to the destination nodes.
- Non-determinism of the network due to various routes available on the Internet, where the decision of the best route to be used should be taken for each data packet received.
- Network data should be secured, meeting the following security properties: confidentiality, authentication and integrity of messages.
- Considering the remote control in public networks, OPC solutions based on WebServices are even slower than those based on DCOM(Distributed Component Object Model) (Advosol Inc., 2004).

3 GENERIC ARCHITECTURE FOR REMOTE TUNING

The architecture proposed for the remote tuner is based on the interconnection of modules in three different contexts: the industrial plant, the server and the client. It is based on the client-server cooperation model, which consists of different interconnected modules providing process and configuration variables from the industrial plant to the remote client. The proposed architecture is implemented in a generic "open" design and can be used with any commercial software component. The figure below shows the components of the architecture.

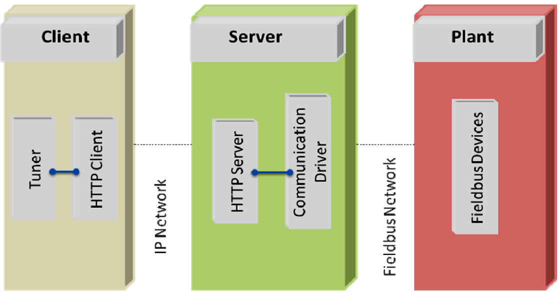

Figure 1: Remote Tuner Architecture.

Inside the "server" module, indicated in the architecture proposed in Figure 1, the communication driver is the communication interface between the server and field devices for data acquisition. Being a widely used standard in the industrial environment, OPC was selected for this project to communicate to the fieldbus network.

Moreover, in the server module, the HTTP (Hypertext Transfer Protocol) Server is responsible for processing remote requests from several clients connected to server module. The communication between the remote HTTP server and the clients can use proprietary protocols, WebService and other protocols over HTTP.

The "client" module represents the remote monitoring and tuning unit. The client must be a standard OPC client allowing the communication to a variety of field device networks. The Cybertune prototype for this study used an OPC client architecture for local communication and a CyberOPC client for remote communication.

In order to avoid the non-determinism of the Internet, the strategy used in this work was attaching a timestamp to each sample. This way, the time measured will always be used, thereby obtaining the actual rate for each performed identification. For the remote communication, using CyberOPC ensures that there will be a constant data acquisition on the control side, with packet sequencing, determining the information obtained by the remote client.

CyberOPC is based on open standard technologies. It minimizes software and interfaces layers for a better and faster network use because it is a technology with a simple philosophy dedicated to industrial applications with "soft real time" requirements in IP networks with assured bandwidth. One of the features from this protocol is the use of an internal cache memory, which provides better performance in processing messages.

Server and client roles in the proposed architecture using CyberOPC communication protocol are detailed below.

3.1 The CyberOPC Communication System

The most commonly used OPC specification is the OPC Data Access (DA) 2.x and 3.x. It does not intend to compensate communication latencies that can occur in wide area networks neither provide communication for clients or servers intermittently connected to the network. Another restriction comes from the fact that OPC DA data cannot be transferred over the Internet, because all TCP/IP ports except Port 80, which allows only textual data, are usually blocked at corporate firewalls. However, OPC is based on a Microsoft standard for component communication – DCOM (Distributed Component Object Model) – that is not textual.

The OPC XML-DA and the incoming OPC-UA specification (OPC Foundation, 2008) define a WebServices based approach for reading from and writing data to plant floor automation systems. In this approach, all OPC data are formatted in XML blocks and transmitted using simple object access protocol (HTTP-SOAP). The choice to adopt WebServices solves the problems related to binary data blocked at firewalls and promotes software integration manufacturing with WebServices technologies, despite the low quality of service level to address typical timing requirements of real time process control networks.

Step 1 in Figure 2 represents the processing of CyberOPC commands without SOAP preprocessor. Introducing the OPC cache strongly reduces calls to the OPC Client. Tests have reported the reduction of 70% of the Message Broker Time when compared to the time consumed by a Gateway WebServices based. Steps 2, 3 and 4 represent the interaction between the OPC library and the CyberOPC Application Server

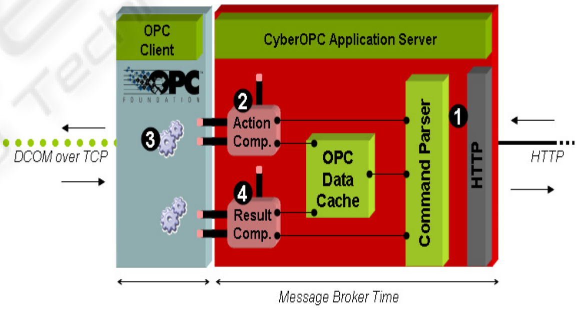

Figure 2: Message Broker Time for CyberOPC Gateway. OPC Packet Data and requests are encoded as text using JSON or XML syntax

For HTTP communications, there are two categories of security mechanisms: transport level security and message level security. The transport level security mechanism uses Server Secure Socket (SSL), using digital X.509 certificates (ITU Recommendation, 1997). The CyberOPC communication system replaces the OPC Polling mechanism over SSL.

3.2 The Cybertune Structure

The Cybertune prototype consists of four main operational modules: data acquisition, system identification, model transformation and tuning modules. A schematic of the structure is shown in figure 3.

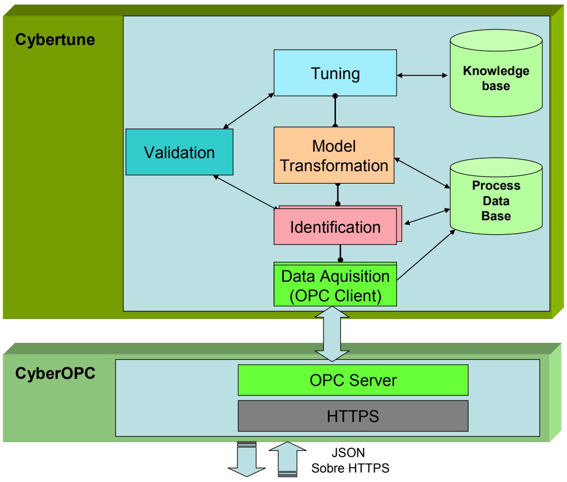

Figure 3: Cybertune structure.

The data acquisition module consists of an OPC or CyberOPC client in accordance with the specifications of the OPC standard (OPC Foundation, 2008) or the specifications in (Torrisi, 2007). Thus, the component interface has the same data access philosophy, consisting of OPC library records, adding groups and items to the database, and acyclic communication per event, where the client is notified when a new data event is issued by the server.

For the identification module, responsible for determining the system transfer function, the ARX model was used, which provides good results for first and second order linear systems, the most common systems in industrial environments (Yu, 2006). As this project aims to validate the architecture for online identification and tuning, the Cybertune needs to receive process data and automatically perform the identification.

Assuming that there are communication delays and transmission failures while executing online remote identification, this work proposes the following methodology. First, every sample collected by CyberOPC has a timestamp with the time when the data was acquired by the gateway. In addition, since the ARX model requires continuous sampling and CyberOPC sends data (per event on data change) in an optimized way, it is necessary to rebuild the process signal at a constant sampling rate.

To solve this case, a "pre-identification” module was included, being responsible for receiving queued data from the acquisition module and sending samples to the identification data queue at a constant sampling rate. Interpolation between two sample points used the first order equation. An example of this architecture is shown in figure 4.

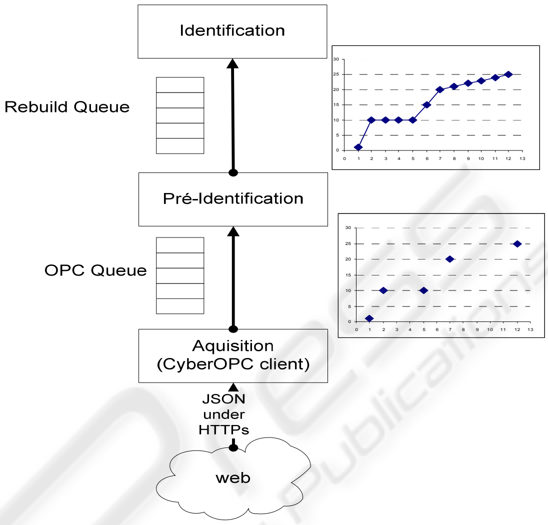

Figure 4: Pre-Identification module.

After the identification and validation of the ARX model is necessary to transform the ARX model in open loop model to be subsequently applied to the model based methods for tuning. The algorithm used for model transformation module for open and close loop was described in (Fernandes and Brandão, 2008).

Finally, the tuning module applies the tuning methods to the model obtained before. The Cybertune uses Ziegler Nichols (ZN) and internal model control (IMC) common model-based tuning methods (Ang, 2005).

4 TESTS AND RESULTS

Identification tests were performed to validate the proposed architecture simulating first-order-plus-dead-time systems using local and remote identification in a corporate network.

Tests were conducted using the FieldBus Plant Simulator (FBSIMU), which simulates industrial plant and fieldbus control logic. .The studies of (Pinotti and Brandão, 2005) showed that FBSIMU has good approach to simulate real system.

Figure 5 presents the tests scenarios of this work. Local tests used communication with Cybertune and FBSIMU in the same station. Remote tests used a 2Mbits internet connection.

Server station consists of a CyberOPC gateway module communicating via OPC to FBSIMU. Remote client station consists of Cybertune communicating via CyberOPC protocol with the FBSIMU.

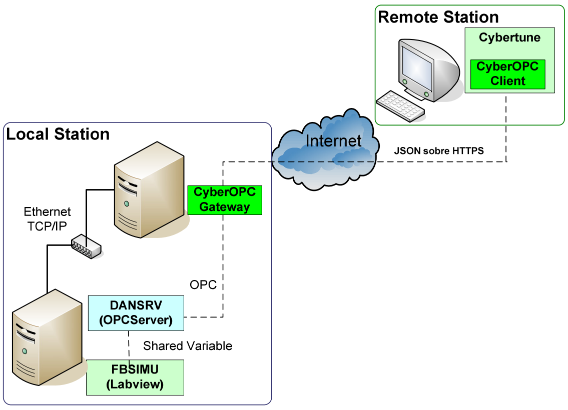

Figure 5: System architecture for local and remote communication between Cybertune and FBSIMU.

Simulation tests were performed using three systems with different characteristics: two tests used slower systems (such as an oven or industrial chemical process), and the other test used a fast system (such as a flow control loop). The transfer function of the systems were showed below, where in this paper is called system 1 the equation (1), system 2 the equation (2) and so on.

$$G_p = \frac{2}{100\,s+1} e^{-50\,s} \tag{1}$$

$$G_p = \frac{25}{15\,s+1} e^{-35s} \tag{2}$$

$$G_p = \frac{3}{150s+1} e^{-20s} \tag{3}$$

To validate the tests, the ITAE performance index and the correlation index (FIT) were used in relation to the actual signal and the identified signal. For a correlation index higher the identification is considered as good (Ljung, 1999).

3.1 Tests Results

For the first test, consider the transfer function (1) of the system 1.

Initially regarding the local identification test, identification is estimated according to approximation using a fourth order ARX model and sampling rate (To = 1.0 sec). The model shown in (4) was obtained with FIT=96.75%:

$$FTMA(z) = \frac{-0.0170z^3 + 0.0060z^2 + 0.0105z + 0.0023}{z^4 - 1.0030z^3 - 0.4280z^2 - 0.0129z + 0.4448} \tag{4}$$

Then, the ARX model is transformed into the open loop model according to the equations proposed in (Fernandes e Brandão, 2008), which results in the following model approximation:

$$G_p = \frac{2.00}{102.3\,s+1} e^{(-46.30\,s)} \tag{5}$$

The graph shown in Figure 6 compares the real system and the system identified locally. The final solution has FIT equals to 98.21%.

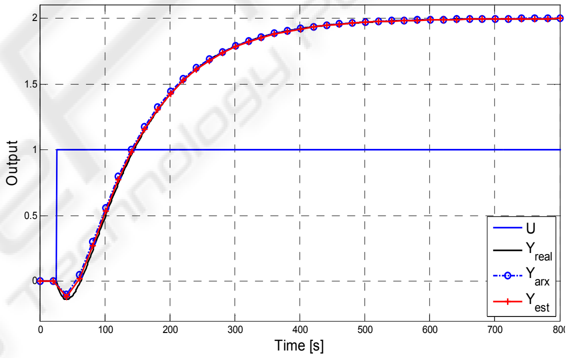

Figure 6: Cybertune identification of system 1 in a local station. It shows the original signal (Yreal), the fourth order ARX signal (Yarx) and the identified open loop system (Yest).

The same system defined in (1) was used in the remote identification test. As described in the previous test, considering an approximation ARX with a fourth order and sampling rate (To = 2.0 sec), the model is estimated according to the following equation (FIT=93.78%):

$$FTMA(z) = \frac{-0.0028z^3 - 0.0120z^2 + 0.0083z + 0.0080}{z^4 - 1.4310z^3 + 0.1628z^2 + 0.07639z + 0.1931} \tag{6}$$

After transforming the ARX model into the open loop model, the model approximation is obtained:

$$G_p = \frac{2.00}{118.7\,s+1} e^{(-38.7\,s)} \tag{7}$$

The graph shown in Figure 7 compares the real system and the system identified remotely. The final solution has FIT equals to 93.65%.

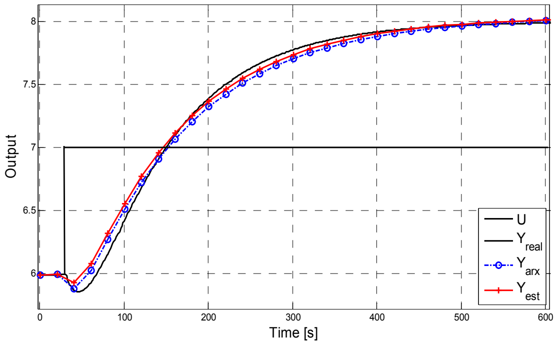

Figure 7: Cybertune identification of a system 1 in a remote station. It shows the original signal (Yreal), the fourth order ARX signal (Yarx) and the identified open loop system (Yest).

Table 1 summarizes the results from the three tested systems. The experimental procedure for systems 2 and 3 are omitted because they are the same as described for system 1.

Table 1: Local and Remote Tests Results.

System	Description of the test with Cybertune	ITAE	FIT[%]
1	Local identification (T_o = 1.0 sec)	4.57E+2	98.21
	Remote identification (T_o = 2.0 sec)	6.34E+3	93.65
2	Local identification (T_o = 1 sec)	5.93E+2	98.46
	Remote identification (T_o = 5 sec)	8.13E+3	91.60
3	Local identification (T_o = 1 sec)	3.54E+2	99.77
	Remote identification (T_o = 5 sec)	1.86E+3	92.80

In the tuning phase is used the open loop model obtained from the identification phase. The figure below shows the tuning with common methods ISTE, ITAE and IMC and ZN.

5 CONCLUSIONS

This paper proposed an architecture to execute remote tuning of industrial control systems using the Internet, fulfilling acceptable security and performance requirements. In order to validate the architecture, a software application using CyberOPC and called Cybertune were presented. The validation consisted of a model-based identification and tuning of three selected first order plus dead time systems once this is a typical class of industrial systems, but the architecture can be extended to other configurations.

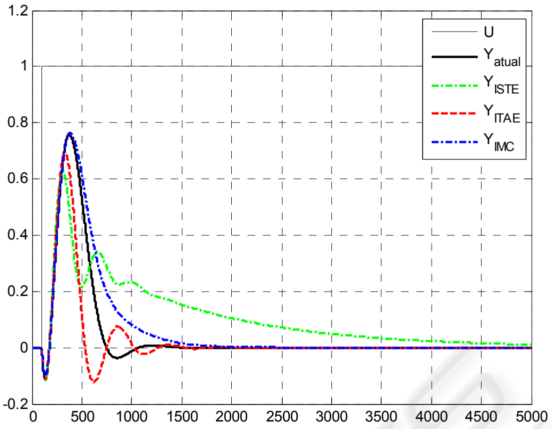

Figure 8: Example of tuning the model obtained in (7) in a remote station. It shows the original tuning with ZN (Y_{atual}) and some common methods based in model.

The tests and results session of this paper focused on a given first order system with dead time for PID controllers, tests for other two systems were conducted and the results were summarized.

The tests demonstrated that the remote model identification is very close to the local identification and the original system, which validates the architecture for identification and subsequent tuning implemented with model-based methods.

For remote identification, it is necessary to pre-filter the signal in order to increase the efficiency of ARX identification.

In future researches, we intend to validate the algorithm in real plant floor systems, through fieldbus system applications.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Brazilian agency FAPESP for financial support received and the academic support and research structure of the Engineering School of São Carlos - University of São Paulo.

REFERENCES

Abdelzaher, T. F..; Shin, K.G.S.; Bhatti, N., 2002. Performance Guarantees for Web Server End-Systems:

A Control-Theoretical Approach, In: *IEEE Transactions on Paralllel and distributed systems*, vol 13, no 1, January 2002.

Aguirre, L. A., 2004. Introdução a Identificação de Sistemas, Técnicas Lineares e Não Lineares aplicadas a sistemas reais. 2ª edição, 2004, Editora UFMG.

Ang, K. H.; Chong, G.; Li, Y. (2005). PID Control System Analysis, Design, and Technology. In: *IEEE Transaction on Control Systems Technology*, Vol 13, No 4, July 2005.

Avoy, T. M.; Jounela, S.L.J.; Patton, R.; Perrier, M.; Weber, H.; Georgakis, C., 2004. Milestone report for area 7 industrial applications. In: *Control Engineering Practice* 12 (2004) 113-119.

Advosol Inc., XMLDA.NET White Paper, 2004 Available:*http://www.advosol.com/driver.aspx*?Topic=WhitePaperXMLDANET.

Batur, C.; Ma, Q.; Larson, K., Kettenbauer, N., 2000. Remote tuning of a PID position controller via internet. In: *American Control Conference*, Chicago, 2000.

Brandão, D., Cunha, M. J., Pinotti, J. M., 2004 . Fieldbus Control System Project Support Tool based on Experimental Analysis and Modeling of Communication Bus. In: *IEEE International Conference on Industrial Technology*, 2004, Hammamet, 2004.

Calvo, I.; Marcos, M.; Orive, D.; Sarachaga, I., 2006. A methodology based on distributed object-oriented technologies for providing remote access to industrial plants. In: *Control Engineering Practice,* 14 (2006), pp. 975-990

Fernandes, R. F., Brandão, D., 2008. Método de identificação online de sistemas industriais com controladores PI em malha fechada, In: *Conferência Internacional de Aplicações Industriais*, Poços de Caldas. Anais do VIII INDUSCON, 2008.

Hjalmarsson, H., 2005. From experiment design to closed loop control. *Automatica,* 41 (2005) pp. 393-438.

ITU Recommendation, 1997. X.509 version 3. Information Technology - Open Systems Interconnection - The Directory Authentication Framework, August 1997.

Ljung, L., 1999. *System Identification – Theory for the User*. Second Edition, 1999, Englewood, Prentice Hall.

Luo, R.C.; Chen, T.M., 2000. Development of a multibehavior-based mobile robot for remote supervisory control through the Internet. In: *IEEE/ASME Transactions on mechatronics*, vol. 5, no. 4, December 2000.

Kunes, M., Sauter, T., 2001. Fieldbus-Internet connectivity: The SNMP approach. In: *IEEE Transactions on Industrial Electronics*, 48(6), 1248–1256.

Kurose, J. F., Ross, K. W., 2006. *Redes de computadores e a Internet*. 3ª Edição, 2006, Editora Addison Wesley.

Qin, W., Wang, Q., 2007. An LPV approximation for admission control of an internet web server: Identification and control. In: *Control Engineering Practice*, Feb 2007.

OPC Foundation, 2006. OPC Standard. In: <*www.opcfoundation.org*>. Access em: 26 jul. 2007.

Overstreet, J. W., Tzes, A., 1999. Internet-based client/server virtual instrument designs for real-time remote-access control engineering laboratory. In *Proceedings of the American control conference*, Vol. 2, pp. 1472–1476.

Pinotti Jr., M., Brandão, D., 2005. A flexible fieldbus simulation platform for distributed control systems laboratory courses. *The International Journal Of Engineering Education*, Dublin, v. 21, n. 6, p. 1050-1058, 2005.

Torrisi, N. M. ;Oliveira, J. F. G. 2007. Remote Control of CNC Machines using the cyberopc communication system over public networks. *International Journal of Advanced Manufacturing Technology*, v. 2007, p. 001, 2007

Torrisi, N. M., 2007. Sistemas de comunicação para redes IP dedicado a dados industriais codificados em JSON. Patente submetida ao Instituto Nacional da Propriedade Industrial (INPI), sn 018070047508, 2007.

Yang, S.H.; Chen, X.; Alty, J.L., 2003. Design issues and implementation of internet-based process control systems. In: *Control Engineering Practice* (11) 2003, 709-720.

Yang, S. H., Dai, C., Knott, R.P., 2007. Remote Maintenance of Control System performance over the Internet, In: *Control Engineering Practice*, Volume 15, Issue 5, May 2007, Pages 533-544

Yu, C. C., 2006. *Autotuning of PID Controllers: A Relay Feedback Approach*. 2nd Edition, Springer, 2006.

Zheng, L., Nakagawa, H., 2002. OPC (OLE for process control) specification and its developments, SICE 2002 – *Proceedings of the 41st SICE Annual Conference*, pp. 917-920 vol. 2, 2002.