Real-Time Modeling in Pervasive Mining

Jose Perez
Electrical Engineering Department
University of Santiago of Chile
Santiago of Chile, Chile
e-mail: jose.perezb@usach.cl

Miguel Alfaro
Industrial Engineering Department
University of Santiago of Chile
Santiago of Chile, Chile
e-mail: miguel.alfaro@usach.cl

Ismael Soto
Electrical Engineering Department
University of Santiago of Chile
Santiago of Chile, Chile
e-mail: ismael.soto@usach.cl

Francisco Cubillos
Chemical Engineering Department
University of Santiago of Chile
Santiago of Chile, Chile
e-mail: francisco.cubillos@usach.cl

Pei Xiao
Centre for Communication Systems Research
University of Surrey, UK
Guilford, United Kingdom
e-mail: p.xiao@surrey.ac.uk

***Abstract*—This paper introduces an integrated design for rea-time plant modeling at a copper mine, using Support Vector Machines (SVM), Neural Networks, and new technologies as Machine-to-Machine (M2M) and Cloud Computing with and an Android client. This solution was designed for a plant inside a copper mine which cannot tolerate interruption and for which their modeling and identification in situ, in real time, is an essential part of the system to control aspects such as instability by adjusting their corresponding parameters without stopping the process.**

Keywords Mining; Modeling; Support vector machines; Pervasive; M2M; Cloud; NARX; NARMAX; neural networks

I. INTRODUCTION

Several dramatic mine accidents have turned the spotlight on mining safety and to the need for stricter supervision and the deployment of new technologies to enhance miner´s safety and processes inside the mine. Several efforts have been made in this direction and the trend is to automate miner´s tasks as much as possible. This can be done by using new generation telecommunication [1] and computing technologies for accessing resources in mining operations. Enhances in automation and remote operations, helped by new applications for mobile devices adapted to the mining sector [2], will allow employees to supervise remotely the production tools and to have access to applications in servers outside the mine, without getting out from it.

In this work we apply the concepts introduced above to solve the problem of monitoring and modeling a plant (dynamic system) which cannot tolerate interruption. Their modeling in-situ, in real-time, is indispensable to control aspects such as instability without stopping the process.

Hence, we present a real-time, integrated monitoring/modeling system, for a plant inside a mine. The obtained model will be used to control the plant and will contribute to enhance mine safety.

The proposed system includes the telecommunications platform, a server in the cloud that runs a modeling method (based on neural network, support vector machines or another) and a mobile client inside the mine.

This is an ongoing project, and at this stage it is still under development. M2M communications must be supported by operators; however this issue has not been solved yet. Currently we are working in the development of software interfaces for the mobile client, and in the server application located in a private cloud.

The rest of the paper is organized as follows: Section II describes briefly the proposed general communications architecture. Section III introduces some terms and concepts related to modeling of dynamic systems used in this work; Section IV presents a practical modeling application related to on-line/real-time estimation of states in a complex Semi-Autogenous Grinding process (SAG) in copper mining; and Section V concludes the paper

II. SYSTEM DESCRIPTION

In this section, we present the description of a system for plant modeling inside a cooper mine, keeping in mind the goal of automate miner´s tasks as much as possible by using new generation telecommunication and pervasive computing technologies. This solution has been applied to a semi-autonomous grinding inside the mine which cannot tolerate interruption and for which their modeling in situ, in real time, is an essential part of the system to control the filling level without stopping the process. The proposed design considers Machine-to-Machine based (M2M-based) communication mode running a client-server application with the server in the cloud and an Android client inside the mine. This brings up pervasive mining, a system with wider coverage, higher communication efficiency, better fault-tolerance, and anytime anywhere availability.

Once the plant has sent the data to the server in the cloud (Fig. 1), they can be used to construct a model using tools like Neural Networks (NN), Support Vector Machines (SVM) or any other suitable tool. In particular in this study the SVM approach was used.

The program in the server analyzes the data collected from the plant in order to identify its behavior and proposes to the on-site Android's operator a pool of candidate plant models from which the operator can choose the most suitable one. At the same time, a series of graphics and results are sent and displayed on a screen in a control room located anywhere inside or outside the mine, where the whole process is controlled. The operator on site through the Android device can pick up the correct model to further adjust the related parameters of the plant.

A. M2M Communications and Architectures

In relation to M2M communications platform, Fig. 2 illustrates the corresponding high level systems architecture.

In this scenario multiple connectivity options are available to serve M2M applications requiring connectivity between end devices. The M2M client device can connect to the M2M server directly through a WAN connection (e.g., cellular 3G/4G), M2M Gateway or an aggregation point [3]. In our design, our server application (Identification and Modeling program) will be installed in the M2M server in the Cloud.

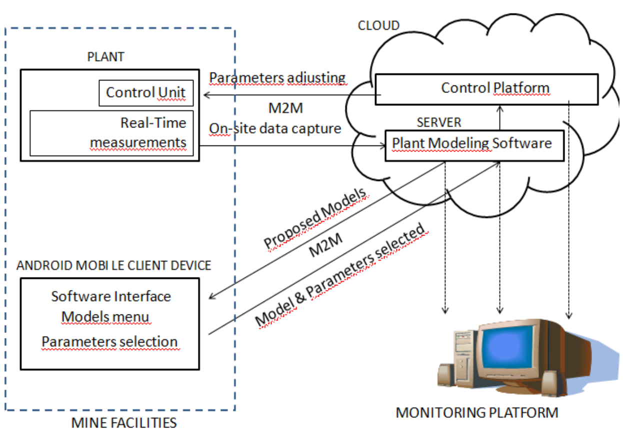

Figure 1. Identification and modeling architecture

III. MODELING IN DYNAMIC SYSTEMS

A. Modeling tools

Modeling theory may fail to generate acceptable reproductions of reality. In many cases, adaptive methods can be successfully applied using function approximation techniques to avoid the necessity of accurate mathematical plant description. Many efforts have been made to address this problem. In [4]the on-line identification of complex processes problem with neural networks, it was proved that feed-forward type neural networks (FFNN) converge satisfactorily in a few iteration cycles, showing better prediction capacity than recursive algorithms. This study considered FFNN networks adapted by three different methods (inverse Hessian matrix approximation, calculation of the inverse Hessian matrix using a Gauss-Newton recursive sequential algorithm, and calculation of the inverse Hessian matrix in a recursive type Gauss-Newton algorithm.).

In [5] , an observer model based on wavelet transform together with neural network modeling has been applied to solve the state observation problem when the dynamic model of a dynamic system contains uncertainties or it is completely unknown resulting in successful results.

In [6], the problem of on-line model identification for multivariable processes with nonlinear and time-varying dynamic characteristics has been solved by using two online multivariable identification approaches with self-organizing neural network model structures. Two adaptive radial basis function (RBF) neural networks have been defined, and the dynamic model is generated autonomously using the sequential input-output data pairs in real-time applications.

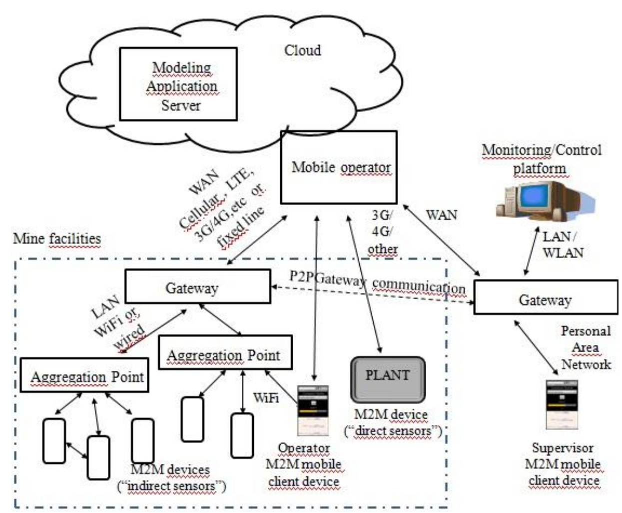

Figure 2. High-level M2M system architecture

B. Model Performance

When we have a set of plant's real data, we divided it into three groups, to train, validate and test the model. However, when this goals are hampered by a lack of reliable experimental data, it is necessary to construct a set of good-quality, hypothetical data. In this sense, an important tool to construct these data is the Lorentz model, which we used at the beginning of our work.

This study will show a brief description of Lorentz model, that reveals the behavior of a dynamic system, and the related Forrester model, which graphically represents a hypothetical plant under study.

Data constructed from Lorentz are very valuable since they may represent the behavior of a plant inside a mine having chaotic outputs, and is useful for reference and comparing purposes. If we have deterministic, chaotic behavior, then reliable forecasting is possible and controlling the process in the mine becomes easier.

C. Influence Diagram

To study dynamic systems behavior, causal diagrams can be used to outline all the elements of a problem without going into the mathematical details in the possible model. An influence or causal diagram (also known as Forrester's diagram) represents influence relations that exist between the elements of a system and therefore provides information about the structure.

Forrester diagram is a symbolic representation of identified level, flow and auxiliary variables of a causal diagram, and is intermediate between the causal diagram and the corresponding first order differential equations. The level variables describe the states of the systems by continuous integration of the results of the action inside the system; flow variables refers to the input and output flows which influence the level; and auxiliary variables can determine a level or rate variable or another auxiliary variable but is not itself a level or rate variable of direct interest to be solved within the model. Forrester model has a direct correspondence with the differential equations that define the relationship among the elements of the dynamic systems. As an example, Fig. 3 depicts the Forrester model corresponding to a plant whose behavior can be represented by the set of differential equations (1) related to Lorentz model of a plant.

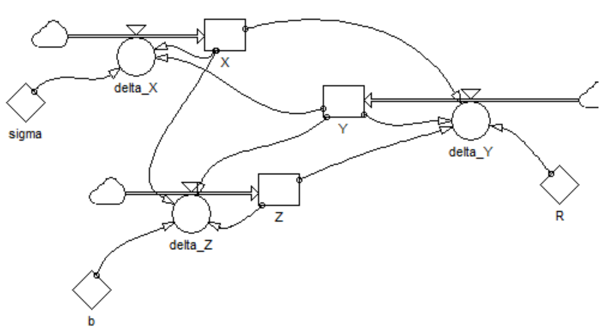

Figure 3. Forrester diagram for Lorentz model

D. Lorentz Model

In 1963 Lorentz abstracted three differential equations that can be used to test ideas in nonlinear dynamics. These equations read:

$$\begin{aligned} \dot{x} &= \sigma(y-x) \\ \dot{y} &= -xz + rz - y \\ \dot{z} &= xy - bz \end{aligned} \tag{1}$$

This set of equations (Lorentz's model) has been applied to the comprehension of complex processes, obtaining generic models to do continuous simulation, in particular related to complex production systems behaviors. As a result, it allows identifying regulation mechanisms following internal and internal disturbances. However, a system having such deterministic behavior may result unpredictable having chaotic solutions because of their sensitivity on initial conditions and setting of the parameters σ, r and b. In particular, the Lorenz attractor is a set of chaotic solutions [7] of the Lorenz system which, when plotted, resemble a butterfly or figure eight, as depicted in Fig. 4.

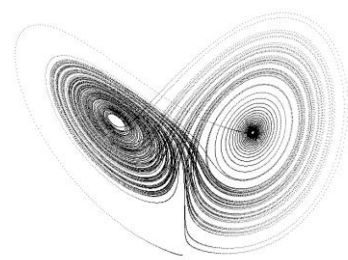

Figure 4. Lorentz attractor

Due to the nonlinear terms, (1) cannot be solved analytically. As an example, if we choose $\sigma = 10$; $r = 28$; $b = 2.67$; and initial conditions $x_0 = 0$; $y_0 = 1$; $z_0 = 0$, and solve using Euler´s numerical solution, for 50000 iterations the three-dimensional plot is the same shown in Fig. 3, and its projection on planes XY, YZ, and XZ, is depicted in Fig. 5.

$$\begin{aligned} \dot{x} &= 10(y-x) \\ \dot{y} &= -xz + 28z - y \\ \dot{z} &= xy - 2.67z \end{aligned} \tag{2}$$

$$x_0 = 0;\ y_0 = 1;\ z_0 = 0$$

is responsible for getting data directly from plant sensors in the mine to the server in the cloud using the M2M infrastructure. Finally, a third interface must communicate any event, related to both mobile client and the plant inside the mine, to the screen at the platform for monitoring and control.

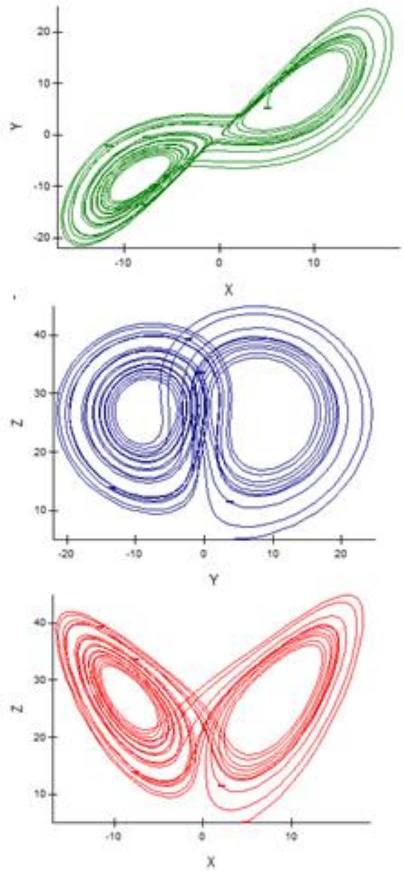

Figure 5. Lorent'z chaotic attractors projections

E. *Client Mobile Device Interface*

Whatever neural network modeling used, the process can be automated in a pervasive context with the help of modern mobile communications platform, cloud computing, and a mobile device acting as a client in-site by the plant, accessing and controlling the modeling process in a server inside the cloud. To achieve this, in the client mobile device, several software interfaces must be implemented to support the proposed solution [8]. An interface must be implemented for data transmission between the application server side in the private cloud and the M2M platform from the service provider [9]. This interface consists of two parts, one is to deploy a web service which provides local services for the remote M2M platform and is responsible for receiving real-time data from the Android client side. The other is to develop the service software which runs in the communication server background, with the functions of commands sending, data parsing, storage and query to the database. Two modules are considered; the first is to request services from the M2M platform, including platform login and sending command and the second is the background processing of data which is called by local web service for parsing data and storing into the database. Another interface is responsible for getting data directly from plant sensors in the mine to the server in the cloud using the M2M infrastructure. Finally, a third interface must communicate any event, related to both mobile client and the plant inside the mine, to the screen at the platform for monitoring and control.

F. *Collecting Data and Request*

From Fig.1 we note that data from plant's measurements are transmitted directly to the server in the cloud. In order to perform modeling and control over the plant, the process will be controlled by the operator accessing the server in the cloud through the client program in the Android mobile device, allowing the operator in real-time, without leaving the site, to control the modeling process. Table 1 represents data from the plant inside the mine received by the server in the cloud. At this point no real data are collected, so they correspond to the solution of equation (2).

TABLE 1. COLLECTING DATA

Time (ms)	X	Y	Z
1	0	1	0
2	0,02	0,998	0
3	0,03956	0,997124	3,99E-05
…	…	…	…
1998	-8,493289	-8,793423	26,608915
1999	-8,499291	-8,799466	26,616371
2000	-8,505295	-8,805386	26,623996

G. *Getting the Model*

To identify and get a model for the plant's behavior, the operator using the application interface in the client mobile device, asks the server to compute the model for the dynamic system under study (Fig. 6), specifying the modeling method.

The server will reply showing the asked model (Fig. 7). Several models can be tested until select a suitable model.

H. *Selecting the Mobile Client*

According to Gartner's statistics up to November 2012 systems, the market share of different operating systems for mobile devices is 72.4% for Android, 13.9% for iOS, 5.3% for Research In Motion, 3.0% for Bada, 2.6% for Symbian, 2.4% for Microsoft and 0.4% for others.

Android is an operating system designed specifically for mobile devices [11]. It runs on the Linux kernel. The Android Software Development Kit (SDK) provides the tools and Application Programming Interfaces (APIs) necessary to develope applications using Java.

Applications written in Java can be compiled to be executed in Dalvik virtual machine, which is a specialized virtual machine implementation designed for mobile device use. Other interesting characteristics of Android are the capability for reusing and replacing components, and the availability of a number of handset layouts, adaptable to larger, VGA, 2D graphics library, 3D graphics library based on OpenGL ES 1.0 specifications and traditional Smartphone layouts.

Hence, to implement our mobile client we used Android because of its open nature, widespread use and the portability of the code.

I. Monitoring Platform Screen

During the entire process, every event is displayed on a monitor screen (Fig. 8) at the platform for control and monitoring, located either inside or outside the mine. The information displayed on the screen includes commands, data sent by plant's real-time measurement system, request and replay messages between mobile client and server, numerical and graphical results from modeling, and real-time plant's response. These enable supervisors or other expert employees to supervise the automated control and tools from a remote operations centre. identify and get a model for the plant's behavior, the operator using the application interface in the client mobile device, asks the server to compute the model for the dynamic system under study (Fig. 6), specifying the modeling method.

The server will reply showing the asked model (Fig. 7). Several models can be tested until select a suitable model.

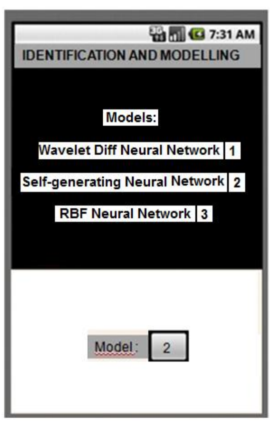

Figure 6. Android client interface

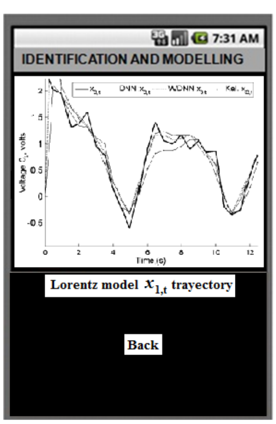

Figure 7. Final Model and parameters adjusting

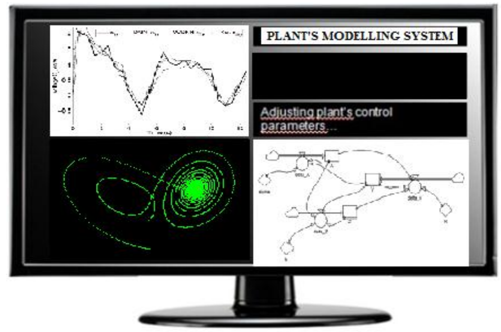

Figure 8. Monitoring screen

IV. MODELING APPLICATION

A. Software Sensor based on a NARMAX - Support Vector Machine Model for Semi-Autogenous Grinding

The estimation of states in complex processes such as the Semi-Autogenous Grinding process (SAG) in copper mining is an important and difficult task due to difficulties in measuring some relevant variables directly online and real time. In [12] the authors present interesting modeling results using Nonlinear Autoregressive Moving Average with Exogenous Input (NARMAX) and Support Vector Machines (SVM), when acting as estimators of a of the most important state variables for SAG milling operation. They propose a simple and original methodology to develop models NARMAX made with SVM. The results show the predictive power of NARMAX models that incorporate the prediction errors in earlier times to predict the future evolution of the process and also the advantage of those elaborated by SVM over those made of neural networks. NARMAX-SVM has a significantly lower MSE than all other models. In terms of the milling process, it provides a useful tool for estimating real-time online and one of the most important variables when it comes to control and optimize the process and cannot be measured using readily available tools.

The effective modeling results from NARMAX-SVM may be available to the operator through its mobile device, and for the in the monitoring platform as well as their mobiles devices anywhere inside the mine.

In the following lines we will explain shortly how NARMAX-SVM works.

B. Virtual sensor structure

The application of virtual sensors to SAG milling process described in [12], is to estimate on-line and in real time the values of the variable "Level". This is a significant variable for the grinding process, which values are very difficult to measure directly, in real time and offline.

In the NARX model proposed to implement the virtual sensors, we use as inputs previous values of the variable to be estimated ("Level"), and an exogenous variable, the "Pressure on the mill shaft breaks." This is a variable easy to measure online and in real time and is related to the variable of interest. are used as inputs. the previous values , the "Pressure on the mill shaft breaks." This is a variable easy to measure online and in real time and is related to the variable of interest.

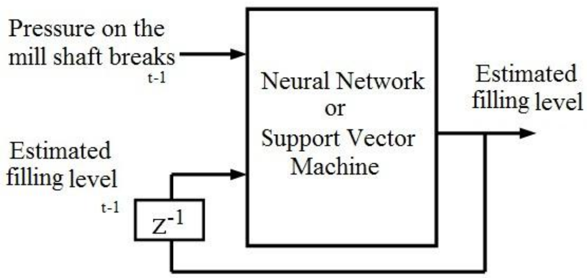

Figure 9. NARX virtual sensor structure

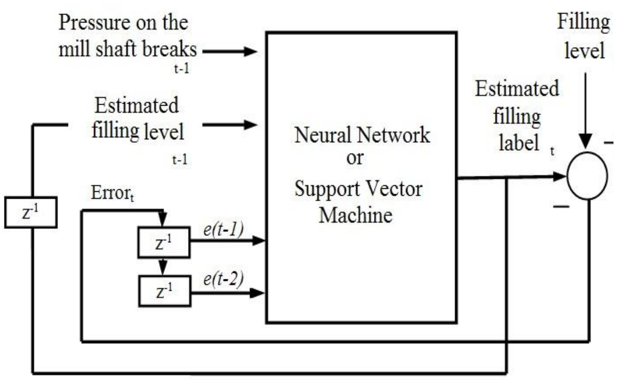

Figure 10. NARMAX virtual sensor structure

C. Data selection, Model Identification and Forecasting

SVM and neural network models were trained with 500 examples and validated with 1000 examples. A third set of examples, the test set was used to obtain the final perpormance indices shown in Table 2. Each example has the fill level and the pressure during breaks at time t-1 as inputs, and the filling level at time t as the output (the models are of first order).

Once identified, the four models obtained for estimating the filling level of the SAG mill (NARX and NARMAX using SVM and neural networks, respectively), their prediction capability for Multiple Step Ahead (MSA) forecast was evaluated. The estimation error was quantified by using the mean square error (MSE) of Matlab. As a result, we can see that SVM implementations perform better than neural networks case.

TABLE 2. MSA FORECASTING MEAN SQUARE ERROR

	NARX	NARMAX
RNA	3.5889	1.0773
SVM	1.0256	0.4424

D. Forecasting Results

Figs.11 and 12 show the estimation of the variable filling level (%) obtained with NARX and NARMAX models, respectively, using SVM in MSA forecasting for the test data set.

From these results we can see that NARMAX model performs better than NARX when both act as predictors MSA. NARMAX type models, though requiring a more complex identification procedure, consider previous prediction errors. Moreover, the models implemented using SVM significantly outperforms the performance of those made by neural networks.

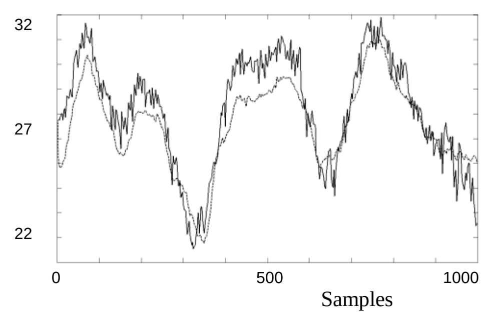

Figure 11. SVM-NARX MSA Filling level estimation

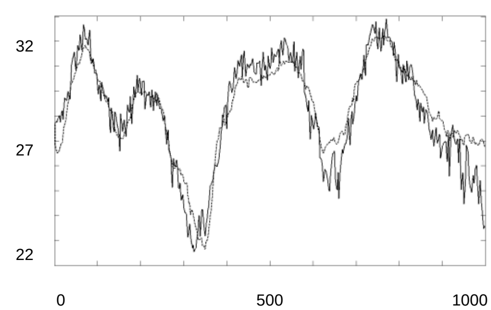

Figure 12. SVM-NARMAX MSA Filling level estimation

V. Conclusions

This solution has advantages in many aspects. The M2M-based communication mode with the server in the cloud and Android client inside the mine brings up pervasive mining, a system with wider coverage, higher communication efficiency, better fault-tolerance, and anytime anywhere availability.

This solution was designed for a plant inside the mine which cannot tolerate interruption and for which their modeling in situ, in real time, is an essential part of the system to control aspects such as instability by adjusting their corresponding parameters without stopping the process.

The proposed model has considered a server in the cloud where it runs the modeling software.

The study has compared the performance of two modeling tools based on data, SVM and neural networks to implement dynamic models. The purpose of these models, namely, NARMAX and NARX, is acting as state estimators for variable filling level of a semi-autogenous grinding process.

The results proved the superiority of SVM-NARMAX models to perform forecasting related to this filling level estimation. Therefore, efforts will be made to implement in the server these modeling tools using a suitable high-level programming language.

The solution proposed responds to both modeling and forecasting of plant's functioning inside a mine. It simplifies the monitoring process, contributes to better control and enhanced safety.

The system proposed may represent a valuable design that helps to do a stricter supervision, set up safer work conditions for the miner, and to deploy new technologies to enhance miner´s safety and processes inside the mine.

Acknowledgment

The authors would like to thanks to the "Center for multidisciplinary research on signal processing" (project Conicyt/ACT1120) and project USACH/Dicyt N°06.917S) for the financial support.

References

[1] N. Radio, Y. Zhang, M. Tatipamula y V. Madisetti, «Next-Generation Applications on Cellular Networks: Trends, Challenges, and Solutions,» *Proceedings of the IEEE , vol.100, no.4,* pp. 841-854, 2012.

[2] E. Mancini, G. Wainer, K. Al-Zoubi y O. Dalle, «Simulation in the Cloud Using Handheld Devices,» de *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*, 2012.

[3] S. Pocuca y D. Giljevic, «Machine to machine (M2M) communication impacts on mobile network capacity and behaviour,» de *MIPRO, 2012 Proceedings of the 35th International Convention* , 2012.

[4] F. Cubillos y G. Acuna, «Simulation Studies of On-Line Identification of Complex Processes with Neural Networks,» *Lecture Notes in Computer Science,Volume 3972*, pp. 808-814, 2006.

[5] I. Chairez, «Wavelet Differential Neural Network Observer,» *Transactions on Neural Networks,* pp. 1439-1449, 2009.

[6] K. Salahshoor y A. S. Kamalabady, «On-line Identification of Nonlinear Multivariable Processes using self-generating RBF Neural Networks,» *Asian Journal of Control, Vol. 12, No. 5,* pp. 626-639, 2010.

[7] M. D. Alfaro, J. M. Sepúlveda y J. A. Ulloa, «Forecasting Chaotic Series in Manufacturing Systems by Vector Support Machine Regression and Neural Networks,» *International Journal of Computers Communications & Control*, pp. 8-17, 2012.

[8] Y. Yang, H. Ye y S. Fei, «Design of communication interface for M2M-based positioning and monitoring system,» *Electronics, Communications and Control (ICECC)*, pp. 2624-2627, 9-11, 2011.

[9] W. Sun y M. Song, «A general M2M device model,» de *Web Society (SWS), 2010 IEEE 2nd Symposium on*, 2010.

[10] C. Wang, W. Duan, J. Ma y C. Wang, «The research of Android System Architecture and application programming,» *Computer Science and Network Technology (ICCSNT), 2011 International Conference on,* vol. 2, pp. 785 - 790, 2011.

[11] W. Hu, D. Han, A. Hindle y K. Wong, «The build dependency perspective of Android's concrete architecture,» de *Mining Software Repositories (MSR), 2012 9th IEEE Working Conference on* , Zurich, Switzerland, 2012.

[12] G. Acuna, M. Curilem y F. Cubillos, «Development of a Software Sensor based on a NARMAX-Support Vector Machine Model for Semi-Autogenous Grinding,» *Revista Iberoamericana de Automatica e Informatica Industrial,* p. unpublished, 2011.