

Desarrollo de un Sensor Virtual basado en Modelo NARMAX y Máquina de Vectores de Soporte para Molienda Semiautógena

Gonzalo Acuña^{a,*}, Millaray Curilem^b, Francisco Cubillos^c

^a Departamento de Ingeniería Informática, Universidad de Santiago de Chile, USACH, Av. Ecuador 3659, Santiago, Chile.

^b Departamento de Ingeniería Eléctrica, Universidad de la Frontera, UFRON, Av. Francisco Salazar 01146, Temuco, Chile.

^c Departamento de Ingeniería Química, Universidad de Santiago de Chile, USACH, Av. Ecuador 3659, Santiago, Chile.

Resumen

La estimación de estados, en procesos complejos como el proceso de molienda semiautógena (SAG) en la minería del cobre, es una tarea difícil debido a las dificultades para medir directamente ciertas variables relevantes en línea y tiempo real. En este trabajo se amplía una comparación, iniciada en trabajos anteriores de estos mismos autores, entre modelos dinámicos NARX y NARMAX construidos con el uso de Redes Neuronales Artificiales (RNA) y Máquinas de Vectores de Soporte (SVM), cuando actúan como estimadores de una de las variables de estado más importantes para la operación de molienda SAG. Para lograr esta comparación se propone una metodología simple y original para desarrollar modelos NARMAX confeccionados con SVM. Los resultados muestran la potencia predictiva de los modelos NARMAX, que incorporan los errores de predicción en tiempos anteriores para predecir la evolución futura del proceso y la ventaja de aquellos elaborados mediante SVM por sobre los confeccionados con RNA. NARMAX-SVM presenta un MSE significativamente inferior al de todos los otros modelos. En términos del proceso de molienda, se proporciona una herramienta útil para la estimación en línea y tiempo real de una variable que permite controlar y optimizar el proceso y que no puede ser medida mediante instrumentos fácilmente disponibles. Copyright © 2014 CEA. Publicado por Elsevier España, S.L. Todos los derechos reservados.

Palabras Clave:

Redes Neuronales Artificiales, Máquinas de Vectores de Soporte, NARX, NARMAX, Proceso de Molienda, Sensor Virtual.

1. Introducción

Con la creciente complejidad de los procesos industriales se hace en ocasiones muy difícil la elaboración de modelos dinámicos fenomenológicos o de primeros principios adecuados para realizar predicción, optimización y control, entre otras tareas importantes. Para elaborar un buen modelo dinámico por lo general es necesario un arduo trabajo de diferentes especialistas con un conocimiento acabado del proceso. De hecho muy a menudo los modelos de primeros principios obtenidos son muy complejos y difíciles de identificar debido a problemas estructurales.

Un enfoque alternativo y fructífero para abordar este problema consiste en el diseño de modelos basados en datos. En este sentido, en la última década, las redes neuronales artificiales (RNA) han demostrado ser una poderosa herramienta para el modelamiento de sistemas. Muchas aplicaciones interesantes en el

campo de la identificación de sistemas, control predictivo, diseño de observadores y predictores se pueden encontrar en la literatura (Chai *et al.*, 2005; Woolley *et al.*, 2010).

A pesar de los buenos resultados logrados con las redes neuronales, todavía subsisten algunas dificultades para su diseño, tales como la elección del número de neuronas en las capas ocultas, el problema del sobreajuste o sobreaprendizaje, la existencia de mínimos locales en la función objetivo y la baja capacidad de generalización, entre otros.

Las Máquinas de Vectores Soporte (SVM) han demostrado su utilidad, mejorando el rendimiento de los diferentes métodos de aprendizaje supervisado, ya sea como modelos de clasificación o como modelos de regresión (SVR). Las SVM tienen muchas ventajas, tales como una buena capacidad de generalización, menor número de parámetros libres a ser ajustados y un proceso de optimización que se basa en una función convexa donde no hay mínimos locales (Schölkopf *et al.*, 2000).

A pesar de estas ventajas, existen dificultades, especialmente relativas al uso de SVM en tareas de regresión en sistemas dinámicos, que aún están por resolver. De hecho las SVM han sido concebidas principalmente para abordar problemas estáticos de clasificación y aproximación de funciones. En el caso de sistemas dinámicos, casi todo el trabajo que se ha realizado en

* Gonzalo Acuña.

Correos electrónicos: gonzalo.acuna@usach.cl (Gonzalo Acuña),
francisco.cubillos@usach.cl (Francisco Cubillos)
millaray.curilem@ufro.cl (Millaray Curilem)
URL: www.informatica.usach.cl (Gonzalo Acuña)

regresión se centra en modelos de tipo NARX (no-lineales, autorregresivos, con entrada exógena) (Sapankevych and Sankar, 2009). Sin embargo, una estructura más completa que NARX y por ende con mejores capacidades predictivas -pues agrega información acerca de errores de predicción anteriores para predecir mejor hacia el futuro- la constituyen los modelos tipo NARMAX (no-lineales, autorregresivos, con entrada exógena y media móvil). Lamentablemente no ha habido reportes claros en la literatura acerca del uso de SVM para desarrollar un modelo dinámico NARMAX. En trabajos de Suykens (Suykens et al., 2002) se establecen las ecuaciones necesarias para entrenar Least-Square SVM (LS-SVM) –una herramienta similar a la SVM tradicional pero con algunas características que le añaden mayor simpleza de uso- de manera de elaborar modelos dinámicos de tipo NOE (no-lineales, con error de salida) que tienen características similares a los NARMAX. Sin embargo en esos trabajos también se establece la gran dificultad de manejar dichas ecuaciones a la hora de realizar implementaciones prácticas. Hornstein and Parlitz (2004) proponen un método iterativo para entrenar un modelo NAR con errores anteriores, utilizando la autocorrelación de los residuos como criterio de término del entrenamiento. Otros autores mencionan aplicaciones de modelos NARMAX implementados con SVM pero no aclaran suficientemente cómo es que fue utilizado SVM en el desarrollo de dichos modelos dinámicos (Martínez-Ramón *et al.*, 2006, Guo *et al.*, 2008). La complejidad del uso de SVM para representar modelos NARMAX hace que las RNA aún constituyan la herramienta preferida (Gao and Joo, 2005).

Desde el punto de vista de los procesos complejos a ser modelados, un caso de relevancia en minería lo constituye la molienda semiautógena (SAG) (Wolmarans and Morgan, 2009). En ella las condiciones óptimas de funcionamiento de los molinos son fuertemente dependientes de la correcta determinación de las variables de estado relevantes del proceso. Desafortunadamente, las condiciones que prevalecen en un molino SAG hacen difícil medir estas variables en línea y en tiempo real, algo que es especialmente complejo en el caso de la variable “Nivel de Llenado” del molino, tal vez la más importante para efectos de controlar y optimizar el proceso (Salazar *et al.* 2009).

Por otra parte, a falta de sensores reales, los sensores virtuales (software-sensors) han demostrado ser herramientas poderosas para la determinación de las variables de estado que no se pueden medir directamente (Suárez and Gómez, 2011). Un sensor virtual es un modelo, estático o dinámico, mucho más simple que el modelo fenomenológico del proceso, pues sólo involucra una relación entre variables de salida posibles de ser medidas en línea y en tiempo real y las variables de estado relevantes del proceso. En consecuencia el sensor virtual puede elaborarse considerando solo una parte del modelo fenomenológico o a partir de modelos basados en datos utilizando herramientas de la inteligencia computacional -como las RNA o los SVM- o a partir de combinaciones de todos ellos (Yan *et al.*, 2004, Gonzaga *et al.*, 2009).

En trabajos recientes aplicados al proceso de molienda SAG (Acuña and Curilem, 2009, Curilem *et al.*, 2011), se realizó la comparación de sensores virtuales implementados con modelos dinámicos basados en SVM y RNA para estimar el nivel de llenado del molino SAG. En estos trabajos, se comparó el desempeño de modelos neuronales de tipo NARX y NARMAX y un modelo NARX implementado con LS-SVM. Se mostró que el desempeño del modelo de tipo NARX implementado mediante LS-SVM fue significativamente superior al desempeño del mismo

modelo implementado mediante RNA. Ambos modelos fueron sin embargo superados por el modelo de tipo NARMAX implementado con RNA. Dado el éxito, por una parte, del modelo tipo NARMAX y por otra de las SVM como herramienta de modelamiento, quedó planteada la necesidad de implementar modelos NARMAX con SVM. Este trabajo amplía la comparación anterior incorporando el desarrollo de modelos de tipo NARMAX con SVM. Su objetivo es presentar claramente la metodología empleada para la elaboración de dichos modelos utilizando como aplicación, el desarrollo de un sensor virtual dinámico para la variable de estado “Nivel de Llenado” de un molino SAG.

La organización de este trabajo contempla, después de esta introducción, una sección dedicada al proceso de molienda SAG. La regresión mediante SVM es descrita en la sección siguiente. Los modelos NARX y NARMAX confeccionados mediante RNA y SVM son revisados a continuación, destacándose, en esta sección, la metodología propuesta para confeccionar modelos NARMAX con SVM. Los resultados comparativos son mostrados y discutidos en la sección siguiente para terminar con una conclusión.

2. Descripción del proceso de molienda SAG

El objetivo de los procesos de concentración en la minería es la recuperación de las partículas de especies valiosas (cobre, oro, plata, etc.) que se encuentran en las rocas mineralizadas. El proceso de concentración se divide en tres pasos: trituration, molienda y flotación.

En el proceso de molienda, el tamaño de las partículas provenientes de la molienda sigue siendo reducido de modo de obtener una granulometría máxima de 180 micras (0,18 mm). El proceso de molienda se realiza utilizando grandes equipos giratorios o molinos cilíndricos de dos maneras diferentes: convencional o molienda SAG.

En los molinos SAG la molienda se produce en parte por la acción de caída del mineral desde una altura cercana al diámetro del molino y también por la utilización de medios de molienda metálico, bolas de acero de 5 pulgadas de diámetro. El molino SAG tiene un nivel de llenado volumétrico que varía desde un 12% al 30% de su volumen, como se muestra en la Figura 1. El mineral se recibe directamente desde la trituration primaria en un tamaño cercano a las 8 pulgadas (Magne *et al.*, 1997).

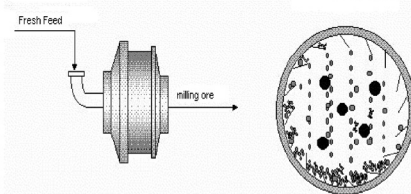


Figura 1: Esquema de Molino SAG- Vista Interna y Externa (Magne *et al.*, 1997)

En la operación de estos molinos, el objetivo es trabajar bajo condiciones que implican el consumo máximo de potencia instalada. Sin embargo, se trabaja en condiciones inestables debido a que un aumento del nivel de llenado del molino más allá

del punto de máximo consumo conduce a una situación de sobrellenado (ver Figura 2).

Adicionalmente, el valor de la potencia máxima que puede ser consumida por un molino SAG no es constante y depende principalmente de la densidad de carga interna, la distribución de tamaños de la alimentación y el estado del revestimiento. El nivel de llenado de la carga interior -que se asocia al consumo máximo de energía- está relacionado con el nivel de carga de medios de molienda y el movimiento de la carga interior. Por esta razón, los operadores del molino SAG deben tratar de conjugar estos factores con el fin de lograr, primero la estabilización de la operación y luego su mejora (Magne *et al.*, 1997). Por eso es importante para ellos contar con información confiable y oportuna sobre el nivel de llenado del molino.

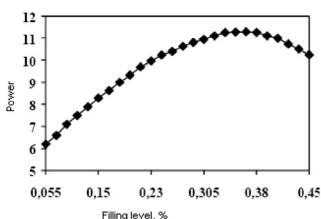


Figura 2: Variación típica de potencia consumida versus nivel de llenado en molinos SAG (Magne *et al.*, 1997).

3. Regresión mediante SVM

Las máquinas de vectores de soporte abordan los problemas de clasificación y regresión no lineal mediante el mapeo de los datos de entrada hacia un espacio de alta dimensión donde la superficie de decisión es lineal.

Los algoritmos SVM aplicados a regresión (SVR, Support Vector Regression) se basan en los resultados de la teoría estadística de aprendizaje dada por Vapnik (Vapnik, 1995) que introduce la regresión como el ajuste de un tubo de radio ν a los datos. El límite de decisión para determinar el radio del tubo está dado por un pequeño subconjunto de ejemplos de entrenamiento llamados vectores de soporte (SV). La SVR de Vapnik estima los valores de \vec{w} para obtener la función de la ecuación 1:

$$f(\vec{x}) = (\vec{w} \cdot \vec{x}) + b \quad \text{con} \quad (1)$$

$$\vec{w}, \vec{x} \in \mathbf{R}^N, b \in \mathbf{R},$$

Introduciendo la función de pérdida ϵ -insensitiva mostrada en la ecuación 2:

$$|y - f(\vec{x})|_{\epsilon} = \max\{0, |y - f(\vec{x})| - \epsilon\} \quad (2)$$

que no penaliza los errores más pequeños que ϵ ($\epsilon > 0$) (donde ϵ corresponde a un valor elegido a priori).

El algoritmo es implementado minimizando el riesgo estructural $\|\vec{w}\|^2$ al que se añade una penalización por dejar los puntos fuera del tubo (identificados por variables de holgura ξ). De esta manera la función de riesgo a minimizar está dada por la

ecuación 3, donde C es una constante que pondera el compromiso entre la complejidad del modelo y los puntos que quedan fuera del tubo, sujeta a las condiciones de borde. La Figura 3 muestra una interpretación geométrica para el caso de una regresión lineal.

$$\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i + \xi'_i) \quad (3)$$

s. a. $y_i - \langle w, x_i \rangle - b \leq \epsilon + \xi_i,$
 $\langle w, x_i \rangle + b - y_i \leq \epsilon + \xi'_i$
 $\xi_i, \xi'_i \geq 0, \quad i = 1, \dots, l$

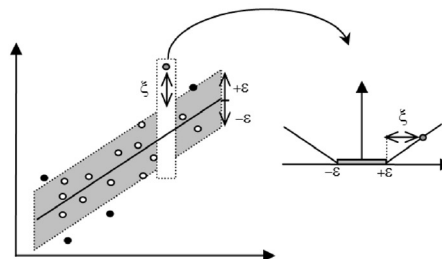


Figura 3: Interpretación geométrica de la SVR en la cual la ecuación de regresión es estimada mediante un tubo de radio ϵ . El compromiso entre la complejidad del modelo y los puntos dejados fuera del tubo de regresión es controlado por las variables "slack" ξ . Los puntos en negro son los Vectores de Soporte

Este problema se resuelve mediante programación cuadrática (QP, Quadratic Programming), planteando el problema dual y utilizando los multiplicadores de Lagrange (Schölkopf *et al.*, 2000).

Para resolver un problema de regresión no lineal es necesario realizar una transformación no lineal $\Phi()$ de las variables de entrada hacia un espacio, llamado espacio de características, que tiene una alta dimensionalidad, pero donde puede existir un hipertubo que contenga los datos, por lo que en ese espacio se puede aplicar una SVR. Algunas funciones, llamadas funciones kernel $k()$ pueden sustituir el producto escalar de dos variables de entrada independientes $\vec{x} \cdot \vec{y}$ por $k(\vec{x}, \vec{y}) = \Phi(\vec{x}) \cdot \Phi(\vec{y})$ que realiza ese producto en el espacio de características. Esta función lleva a cabo el producto escalar en el espacio de dimensión superior de tal manera que se asegura la linealidad de la función de regresión en el nuevo espacio, a través del mapeo no lineal $\Phi()$. Varias funciones se pueden utilizar como funciones kernel, como por ejemplo la función Gaussiana $k(\vec{x}_i, \vec{x}_j) = \exp(-\|\vec{x}_i - \vec{x}_j\|^2 / (2\sigma^2))$ o la función polinomial $k(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \cdot \vec{x}_j)^p$.

4. Estructura e identificación de los modelos NARX

Un modelo no-lineal, regresivo y con entrada exógena (NARX) es la extensión no lineal del modelo ARX y está dado por la ecuación (4) (Ljung, 1987):

$$y(k) = f(y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m), \dots) + e(k) \quad (4)$$

donde $e(k)$, denominado error de predicción en el tiempo k , se

modela como una variable aleatoria proveniente de la realización de un proceso de ruido blanco gaussiano de media cero y varianza σ y que representa la incertidumbre del modelo y el ruido asociado a los datos experimentales.

El predictor asociado a este tipo de modelos está dado por la ecuación (5) y es esquematizado en la Figura 4, donde \hat{y} es la predicción de la variable autoregresiva y , realizada a partir de la información experimental de ella misma y de la variable exógena u en tiempos anteriores $k-1, k-2, \dots$ considerando la función no-lineal ψ (Ljung, 1987).

$$\hat{y}(k) = \psi(y(k-1), \dots, y(k-n), \dots, u(k-1), \dots, u(k-m)) \quad (5)$$

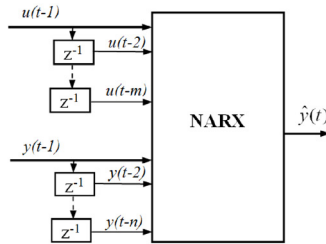


Figura 4: Predictor asociado a modelos de tipo NARX. u es la entrada exógena e y la variable autoregresiva. \hat{y} corresponde a la predicción de la variable autoregresiva.

De la Figura 4 se puede apreciar que la identificación de este tipo de modelos -que se conoce como método de identificación serie-paralela- es muy sencilla. En efecto, basta con suministrar a la herramienta escogida para aproximar la función de predicción ψ -RNA o SVM en el caso de este trabajo- lo siguiente:

- Como entrada:
 - datos experimentales correspondientes a las variables autoregresivas con sus retardos desde $t-1$ a $t-n$ (lo que tiene que ver con el orden del sistema a modelar).
 - datos experimentales de las variables exógenas con sus retardos desde $t-1$ a $t-m$.
- Como salida:
 - datos experimentales de las variables autoregresivas en un instante posterior, t .

En el caso de utilizar RNA para implementar este tipo de modelos, el algoritmo de entrenamiento requerido consiste en el conocido método de retropropagación del error. En el caso de SVM basta con proporcionar adecuadamente la información de entrada y salida al método de optimización cuadrática utilizado para el entrenamiento de SVM para regresión.

Es importante consignar que el entrenamiento de este tipo de modelos es tal que la predicción asociada sólo puede ser rigurosamente hecha a un paso hacia delante (OSA, One Step Ahead). En caso de requerirse de múltiples predicciones hacia delante (MPO, Multiple Step Ahead) es necesario realimentar la variable regresiva hacia la entrada. Sin embargo, como el entrenamiento sólo es hecho a un paso, nada asegura que dicha predicción sea correcta, lo que constituye una posible limitación a la capacidad predictiva de este tipo de modelos a pesar de lo cual

suelen ser utilizados para este tipo de predicción con buenos resultados.

5. Estructura e identificación de los modelos NARMAX

Un modelo no-lineal, regresivo, con entrada exógena y media móvil (NARMAX) es la extensión no lineal del modelo ARMAX y está dado por la ecuación 6 (Ljung, 1987):

$$y(k) = f(y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m), e(k-1), \dots, e(k-p)) + e(k) \quad (6)$$

donde $e(k)$ corresponde a la variable aleatoria descrita anteriormente para la ec. (4) mientras que los términos anteriores son valores conocidos del error de predicción cometido en instantes desde $t-1$ hasta $t-p$ (e.g. $e(k-1) = y(k-1) - \hat{y}(k-1)$).

El predictor asociado a este tipo de modelos está dado por la ecuación 7 (Ljung, 1987) y es esquematizado en la Figura 5:

$$\hat{y}(k) = \psi(y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m), e(k-1), \dots, e(k-p)) \quad (7)$$

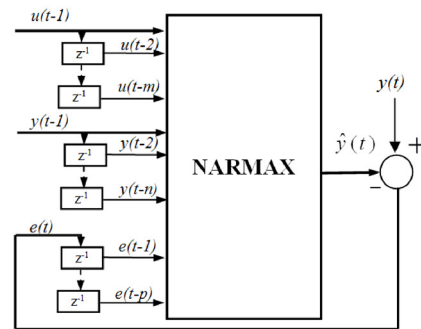


Figura 5: Predictor asociado a modelos de tipo NARMAX. u es la entrada exógena, y la variable autoregresiva y e el error de predicción. \hat{y} corresponde a la predicción de la variable autoregresiva.

Para la identificación de este tipo de modelos es necesario contar con el valor de los errores de predicción cometidos anteriormente. Aparece entonces la dificultad de tener que utilizar el predictor durante el proceso de identificación de modo de fabricar dichos errores en tiempos anteriores. Esto hace que esta identificación, también conocida como método paralelo de identificación de sistemas, sea mucho más compleja de realizar que la identificación serie-paralela anteriormente presentada para los modelos tipo NARX.

La complejidad de la identificación paralela hace que para implementar el modelo NARMAX con RNA se deba utilizar lo que se conoce como retropropagación dinámica o retropropagación a través del tiempo (BPTT, BackPropagation Through Time) inicialmente propuesta por Werbos y ya ampliamente documentada en la literatura (Werbos, 1990)

En cambio, para implementar el modelo NARMAX con SVM, fuera de lo propuesto por Suykens (Suykens *et al.*, 2002) para LS-SVM, no hay metodologías claramente descritas. En consecuencia, en este trabajo se propone la metodología siguiente:

- Dividir el conjunto de datos en tres conjuntos: entrenamiento, validación y prueba.
- Etapa 1:
 - Identificar primero un modelo NARX, como el descrito en la sección 4, utilizando el conjunto de entrenamiento.
 - Obtener el error de predicción MPO con el modelo NARX utilizado con los datos del conjunto de validación.
- Etapa 2:
 - Usar dicho error de predicción como entrada adicional de un segundo modelo NARX que se entrena ahora sobre el conjunto de validación.

Este segundo modelo NARX posee la información del error de predicción en instantes anteriores constituyéndose así en un modelo de tipo NARMAX.

- Probar la capacidad de generalización de este modelo NARMAX sobre el tercer conjunto de datos, el de prueba.

La figura 6 muestra un esquema de este proceso. Nótese que el primer modelo NARX es utilizado únicamente para obtener el error MPO. El segundo modelo, entrenado como si fuera un NARX, pero con los errores anteriores del primer modelo, es el que se constituye como modelo NARMAX.

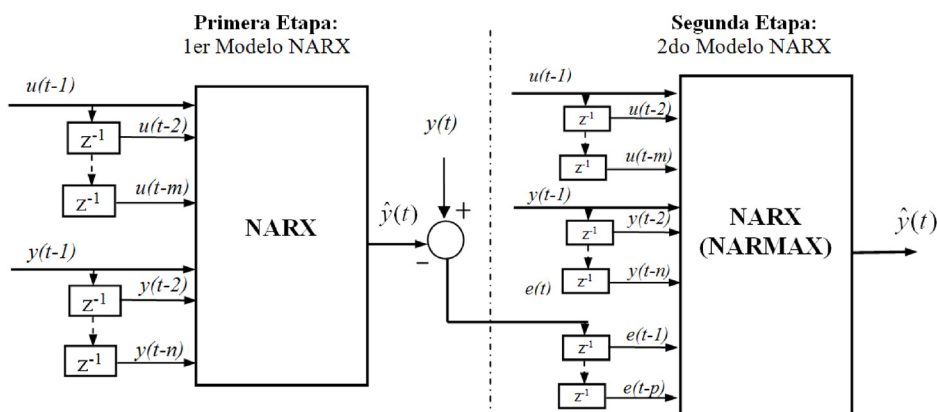


Figura 6: Esquema de las dos etapas de diseño del modelo NARMAX implementado con SVM. En la primera etapa se diseña un modelo NARX usando los datos de un conjunto de entrenamiento. Luego, se obtiene el error de predicción MPO utilizando dicho modelo sobre un conjunto de datos de validación. Posteriormente, en una segunda etapa, se entrena un segundo modelo NARX sobre el conjunto de validación adicionando como entradas hasta p retardos del error de predicción, obtenidos en la primera etapa. Este segundo modelo NARX se constituye en un modelo NARMAX pues fue entrenado con la información de los errores pasados de la variable a predecir y puede ser usado en generalización sobre un tercer conjunto de datos, el de prueba.

6. Resultados y Discusión

6.1. Estructura de los sensores virtuales

La aplicación de los sensores virtuales al proceso de molienda SAG descrito en la Sección 2, tiene por objetivo estimar en línea y en tiempo real los valores de la variable “Nivel de Llenado”. Como se mencionó anteriormente, esta es una variable relevante para el proceso de molienda, cuyos valores son muy difíciles de medir directamente, en línea y tiempo real.

En el modelo NARX propuesto para la implementación de los sensores virtuales, se utiliza como entradas los valores anteriores de la variable a estimar (“Nivel de Llenado”) y una variable exógena, la “Presión en los descansos del eje del molino”. Esta es una variable fácil de medir en línea y en tiempo real y que guarda relación con la variable de interés. Es necesario recordar que aquí se trata solo de implementar un modelo de salida o “sensor virtual”, que suele considerar una relación cercana entre la variable a estimar y una salida fácilmente medible. En caso de tratarse de elaborar un modelo completo del proceso, es necesario previamente hacer un estudio acabado de las variables de entrada,

salida y estado relevantes, lo que puede implicar un estudio para reducir dimensionalidad haciendo uso de técnicas como el análisis de componentes principales o mapas auto-organizados, para sistemas no-lineales.

Para determinar el orden de los modelos se utilizó la función Lipschitz aportada por el toolbox NNSYSID (Norgaard, 2003) lo que arrojó como resultado un orden 2.

En consecuencia, en el modelo NARX propuesto para la implementación de los sensores virtuales, se utiliza como entradas dos valores anteriores de la variable a estimar (“Nivel de Llenado”) y una variable exógena, la “Presión en los descansos del eje del molino”. Esta es una variable fácil de medir en línea y en tiempo real y guarda relación con la variable de interés.

En el modelo NARMAX propuesto para la implementación de los sensores virtuales, se utiliza, en forma adicional a las mismas entradas utilizadas en el modelo NARX, los errores anteriores cometidos por el modelo. En este caso, un método de prueba y error permitió definir dos atrasos en la variable de error.

Los distintos sensores virtuales desarrollados corresponden a las estructuras esquemáticamente representadas en las figuras 7 y 8. Como se puede apreciar en las figuras, cada modelo fue implementado mediante RNA y SVM.

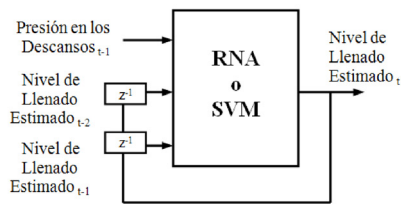


Figura 7: Estimación de la variable Nivel de Llenado del molino SAG a partir de valores anteriores de sí misma y de la variable exógena Presión en los Descansos del molino. Este es un modelo de tipo NARX elaborado con RNA o con SVM. La realimentación de la predicción en t permite utilizar este modelo para realizar múltiples predicciones hacia delante o predicción MPO

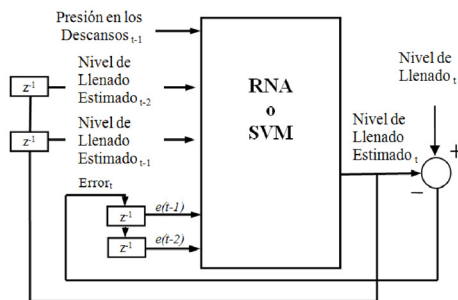


Figura 8: Estimación de la variable Nivel de Llenado del molino SAG a partir de valores anteriores de sí misma, de la variable exógena Presión en los Descansos del molino y de Errores de Predicción en el tiempo anterior $t-1$ y $t-2$. Este es un modelo de tipo NARMAX elaborado mediante uso de RNA o de SVM. Una vez entrenado, cuando este modelo se usa en predicción MPO (múltiples pasos adelante) se introduce un error cero a la entrada del modelo dado el desconocimiento que se tiene de esta variable hacia el futuro.

6.2. Selección de datos, identificación de los modelos y predicción

Se utilizó una base de datos reales provenientes del molino SAG1 de la planta de molienda de El Teniente de CODELCO, Chile, con una capacidad nominal de 1100 thp de mineral. Este proceso está sometido a numerosas perturbaciones no medidas tales como dureza y mineralogía de la alimentación, recirculación de producto rechazado (pebbles) y condiciones de funcionamiento no esperadas tales como obstrucciones en la parrilla de descarga. El molino cuenta con un sistema de control experto que ajusta la velocidad del molino (RPM), alimentación y adición de agua manteniendo la potencia en los niveles adecuados.

Los modelos RNA y SVM fueron entrenados con 500 ejemplos y validados con 1000. Un tercer conjunto de 1000 ejemplos, conjunto de prueba, fue utilizado para obtener los índices finales de desempeño que se muestran en la Tabla 1. Cada ejemplo tiene el Nivel de Llenado en $t-1$ y $t-2$ y la Presión en los descansos en $t-1$ como entradas y el Nivel de Llenado en el tiempo t como la salida (los modelos son de segundo orden).

Una vez identificados los cuatro modelos obtenidos para estimar el nivel de llenado del molino SAG (NARX y NARMAX mediante SVM y RNA, respectivamente), se evaluó su capacidad de predicción. Las predicciones fueron de tipo OSA y MPO en el conjunto de prueba independiente. Es habitual considerar que las

predicciones MPO son más complejas que las predicciones OSA por lo que el modelo que funciona mejor como predictor MPO es el más adecuado (Leontaritis and Billings, 1985). El error de estimación se cuantificó mediante el índice entregado por la función de error cuadrático medio (*MSE*) de Matlab.

6.3. Entrenamiento de RNA

Para elaborar los modelos basados en RNA se utilizó la librería de herramientas de Matlab, NNSYSID, desarrollada por M. Norgaard (Norgaard, 2003) con el algoritmo de optimización de Levenberg Marquardt.

Para el caso NARX se utilizó el procedimiento descrito en (Acuña and Curilem, 2009). En ese trabajo se usó –después de probar diferentes arquitecturas– un Perceptron Multicapa, con tres neuronas en la capa de entrada, una capa oculta con 6 neuronas y una neurona en la capa de salida.

Para el caso NARMAX se utilizó el procedimiento descrito en (Curilem et al., 2011). En ese trabajo varios retardos del error fueron probados empíricamente. Finalmente el orden escogido para el modelo fue 2-1-2-1, o sea, 2 retardos de la variable de salida, “Nivel de Llenado”, un retardo de la entrada exógena, “Presión en los descansos” y dos retardos del error, como lo muestra la Figura 8. Una vez escogido el orden, se diseñó la arquitectura. Esta incluyó 8 neuronas en la única capa oculta. Para probar la capacidad de generalización de este modelo NARMAX en los datos del conjunto de prueba, con predicción de tipo MPO, se optó por asignarles valor 0 a los errores de entrada al modelo, dado que se desconoce el valor de ellos hacia el futuro (Gao et al., 2005).

6.4. Entrenamiento de SVM

Para elaborar los modelos SVM se utilizó Matlab y una biblioteca de herramientas desarrolladas en el INSA de Rouen, Francia (Canu et al., 2005). Como se mencionó anteriormente, los mismos conjuntos de entrenamiento, validación y prueba fueron utilizados tanto para el modelo RNA como para SVM.

Los modelos SVM se elaboraron considerando un Kernel con función de base radial. En el caso NARX, el parámetro C se incrementó en potencias de 2 (Frohlich and Zell., 2005). Los exponentes tomaron valores desde -5 hasta 15, con paso 1 y una búsqueda más exhaustiva se llevó a cabo entre los valores 2 y 5, donde se utilizó un paso de 0.05. Los mejores resultados se obtuvieron para $C = 2^{3.25}$, lo que significa que la solución requiere de una baja complejidad del modelo para lograr una buena generalización cuando se simula una predicción MPO. El parámetro sigma de la función Kernel, también varió según potencias de 2, variando el exponente desde -5 hasta 10 con paso 1. El mejor resultado se obtuvo para $\sigma = 2^7$, por una búsqueda más exhaustiva entre 5 y 8, con un paso de 0.1. El parámetro ϵ varió entre 0 y 1, con paso 0,01. Los mejores resultados se obtuvieron para $\epsilon = 0,2$. La Tabla 1 muestra los resultados del índice MSE para el conjunto de prueba.

En el caso NARMAX –tal como se describe en la metodología explicitada en la Sección 5– se utilizó primeramente un modelo NARX para obtener el error de predicción MPO. Este error fue introducido a la entrada de un segundo modelo NARX de orden 2-1-2-1, la misma configuración que se utilizó con el modelo NARMAX RNA explicado anteriormente (ver figura 8). El modelo SVM fue entrenado buscando los valores de C y sigma en la cercanía de los valores obtenidos en el primer modelo NARX,

con un paso de 0.1. Los mejores resultados se obtuvieron para $C=2^{0.25}$, $\sigma=2^0$ y $\varepsilon = 0.4$.

6.5. Resultados de predicción MPO

La figura 9 muestra los resultados obtenidos por las combinaciones de los dos métodos y los dos tipos de modelos dinámicos, mientras que la Tabla 1 se muestra el resultado de los índices de MSE para cada caso.

A partir de estos resultados se observa que el modelo dinámico NARMAX se comporta mejor que los modelos NARX cuando ambos actúan como predictores MPO en el caso de SVM, lo que no resulta tan significativo en el caso de RNA. La predicción MPO es una prueba bastante difícil para cualquier modelo dinámico. Esto implica la predicción de la variable de salida (Nivel de Llenado), sólo a partir de sus valores iniciales, junto con los valores de entrada experimental de la variable exógena (Presión en los Descansos).

Estos resultados son coherentes con el hecho de que los modelos de tipo NARMAX, aunque requieran de un procedimiento de identificación más complejo, tienen mayor capacidad predictiva al incorporar a su entrada los errores de predicción anteriores. Por otra parte, los modelos implementados mediante SVM presentan un desempeño significativamente superior al desempeño de aquellos confeccionado mediante RNA.

Tabla 1: Error MSE de predicciones MPO para modelos NARX y NARMAX confeccionados con RNA y SVM

	NARX	NARMAX
RNA	1.4027	1.5802
SVM	1.1027	0.4263

7. Conclusiones

Este trabajo presenta una comparación del rendimiento de dos herramientas de modelamiento basadas en datos, las SVM y las RNA para implementar modelos dinámicos de tipo NARX y NARMAX que actúan como estimadores del estado de la variable de Nivel de Llenado de un proceso de molienda semiautógena.

Los modelos resultantes fueron evaluados como predictores de tipo MPO ya que este tipo de prueba requiere de una mayor capacidad de predicción, dado que mide la capacidad de predecir varios pasos hacia adelante, a partir de la entrada actual.

Los resultados comprueban la superioridad de los modelos NARMAX como modelos dinámicos que superan a los NARX respecto de su capacidad de predicción en el caso de SVM. Otro resultado importante de este trabajo es que muestra que en general el rendimiento de las SVM es mejor que el rendimiento de las RNA, que son hasta ahora los modelos basados en datos más utilizados para este tipo de aplicación. Esto es relevante por cuanto las SVM presentan algunas ventajas respecto de las RNA, en especial en lo que se refiere a que poseen un mecanismo que regula el riesgo estructural, por lo que la posibilidad de sobreentrenamiento es menor, y por otra parte, son menos sensibles a los mínimos locales del espacio de soluciones, por lo que su entrenamiento es más robusto.

El trabajo permitió además proponer una metodología novedosa y simple para implementar un modelo NARMAX utilizando SVM, lo que representa una contribución al uso de

estas máquinas para el modelamiento de sistemas dinámicos complejos.

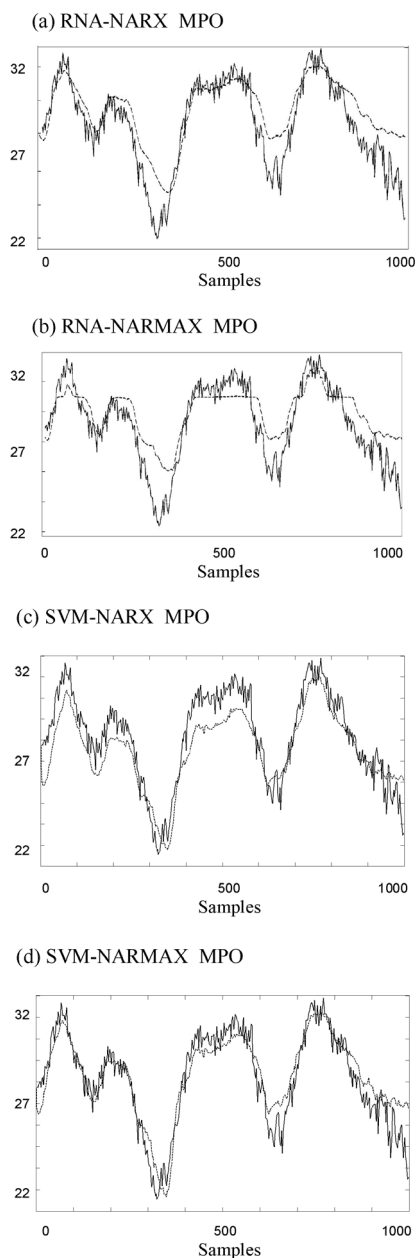


Figura 9: Estimación de la variable Nivel de Llenado (%) mediante modelos NARX y NARMAX utilizando RNA y SVM en modo predictivo MPO (múltiples pasos adelante) para los datos del conjunto de prueba.

Buenos modelos dinámicos de caja negra aplicados a procesos industriales, como el proceso SAG que se muestra en este trabajo, pueden ser de gran relevancia ya que facilitan el diseño de

algoritmos de control predictivo y de estimación que han demostrado ser de gran importancia para mejorar la operación de plantas industriales. El presente trabajo reafirma la incorporación de las SVM como herramientas de modelamiento dinámico que presentan un buen rendimiento para la implementación de sensores virtuales aplicados a procesos industriales complejos.

A futuro, diversos estudios, como el de la robustez ante ruido y perturbaciones de la variable medida, pueden ser incorporados.

English Summary

Development of a Software Sensor based on a NARMAX-Support Vector Machine Model for Semi-Autogenous Grinding

Abstract

State estimation in complex processes such as the semi-autogenous grinding process (SAG) in copper mining is an important and difficult task due to difficulties for real-time and on-line measuring of some relevant process variables. This paper extends a comparison, initiated in previous work of the same authors, between NARX and NARMAX dynamic models built using Artificial Neural Networks (ANN) and Support Vector Machines (SVM), when acting as estimators of one of the most important state variables for SAG milling operation. To accomplish this comparison we propose a simple and original methodology to develop NARMAX models with SVM. The results show that SVM-NARMAX models outperform SVM-NARX models because they incorporate previous prediction errors in order to improve prediction of the future evolution of the process. Advantages of SVM over those RNA models are also highlighted. NARMAX-SVM has a significantly lower MSE than all other models. In terms of the milling process, it provides a useful tool for estimating important state variables that are not easily available on-line and in real time thus aiding control and monitoring of the process.

Keywords:

Artificial Neural Network, Support Vector Machine, NARX, NARMAX, Grinding Process, Software Sensor.

Agradecimientos

Se agradece apoyo de Fondecyt 1090316 y 1090062, DICYT-USACH 06-1219AL y a la Dirección de Investigación de la Universidad de La Frontera.

Referencias

Acuña, G., Curilem, M., 2009, Comparison of neural networks and support vector machine dynamic models for state estimation in semiautogeneous mills, *Lecture Notes in Computer Science*, LNCS, 5845, pp 478-487.
 Canu, S., Grandvalet, Y., Guigue, V., Rakotomamonjy, A., 2005, SVM and Kernel Methods Matlab Toolbox. Perception Systèmes et Information, INSA de Rouen, Rouen, France.

Chai, M-I, S., Song, S., and Li, N-n, 2005, A review of some main improved models for neural network forecasting in time series. In: *Proceedings of the IEEE Intelligent Vehicles Symposium*, Issue 6-8, pp. 866–868
 Curilem, M., Acuña, G., Cubillos, F. and Vhymeister, E., 2011, Neural networks and support vector machine models applied to energy consumption optimization in semiautogenous grinding, *Chemical Engineering Transactions*, 25: 761-766, Dot: 10.3303/CET1125127.
 Fröhlich, H., Zell, A., 2005, Efficient parameter selection for support vector machines in classification and regression via model-based global optimization. In: *Proceedings of the IEEE International Joint Conference on Neural Networks, IJCNN 2005*, pp. 1431-1436
 Gao, Y and Joo Er, M, 2005, NARMAX time series model and prediction: feedforward and recurrent fuzzy neural network approaches, *Fuzzy Sets and Systems*, 150, pp. 331-350.
 Gonzaga, J, Meleiro, L., Kiang, C., Maciel Filho, R., 2009, ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process, *Computers and Chemical Engineering* 33, pp. 43-49.
 Guo, G., Wu, X., Zhuo, S., Xu, P., Cao, B., 2008, Prediction state of charge of Ni-MH battery pack using support vector machines for hybrid electric vehicles, *IEEE Vehicle Power and Propulsion Conference (VPPC)*, September3-5, 2008, Harbin, China
 Hornstein, A. and Parlitz, U., 2004, Bias Reduction for Time Series Models Based on Support Vector Regression. *International Journal of Bifurcation and Chaos*, 14(6), pp. 1947-1956
 Leontaritis, I.J., Billings, S.A., 1985, Input-Output Parametric Models for Non-Linear Systems; Part 1: Deterministic Non-Linear Systems; Part 2: Stochastic Non-Linear Systems, *International Journal of Control*, 45, pp. 303–344.
 Ljung, L., 1987, *System Identification: Theory for the user*. Prentice-Hall, Englewood Cliffs, NJ.
 Magne, L., Valderrama, W., Pontt, J., 1997, Conceptual Vision and State of the Semiautogenous Mill Technology. In: *Revista Minerale, Instituto de Ingenieros de Minas de Chile*. 52(218), 1997 (in Spanish).
 Martínez-Ramón, M., Rojo-Alvarez, J., Camps-Valls, G., Muñoz-Mari, J., Navia-Vásquez, A., Soria-Olivas, E., Figueiras-Vidal, A., 2006, Support vector machines for nonlinear kernel ARMA system identification, *IEEE Transactions on Neural Networks*, 17(6), pp. 1617-1622.
 Norgaard, M., 2003, The NNSYSID Toolbox, for use with Matlab, <http://www.iau.dtu.dk/research/control/nnsysid.html>
 Salazar, J., Magne, L., Acuña, G., Cubillos, F., 2009, Dynamic modelling and simulation of semi-autogenous mills, *Minerals Engineering*, 22(19), pp. 70-77.
 Sapankevych, N and Sankar, R, 2009, Time series prediction using support vector machines: a survey, *IEEE Computational Intelligence Magazine*, pp. 24-38.
 Schölkopf, B., Smola, A., Williamson, R.C., Bartlett, P.L., 2000, New support vector algorithms, *Neural Computation*, 12, pp. 1083--121
 Suárez, A, Gómez, Z., 2011, Softsensor trained using the concept of instrumental variables and applied in the measurement of temperature in a Tientente converter, *Revista Iberoamericana de Automática e Informática Industrial*, 8(1), pp. 54-63.
 Suykens, J.A.K., Van Gestel, T, De Brabanter, J., De Moor, B., Vandewalle, J., 2002, *Least Squares Support Vector Machines*, World Scientific, Singapore.
 Vapnik, V., 1995, *The Nature of Statistical Learning Theory*, Springer Verlag, New York.
 Werbos P.J., 1990, Backpropagation Through Time: What It Does and How to do it. *Proc. IEEE*, Vol 78(10), pp 1150-1560.
 Wolmarans, E and Morgan, P, 2009, Milling circuit selection for the Nkomati 375 ktpm concentrator, *SIAMM-Journal of the South African Institute of Mining and Metallurgy*, 109(11), pp. 653-664.
 Woolley, J., Agarwal, P., Baker, J., 2010, Modeling and prediction of chaotic systems with artificial neural networks. *International Journal for Numerical Methods in Fluids*, 63, pp. 989-1004. DOI: 10.1002/flid.2117
 Yan, W, Shao, H., Wang, X., 2004, Soft sensing modeling based on support vector machine and Bayesian model selection, *Computers and Chemical Engineering* 26, pp. 1489-1498.