

# Hacia un Procesamiento Universal de Señales de Sensores Industriales

Gustavo Monte, Hector Kessel, Damian Marasco, Norberto Scarone, Walter Lagos, Pablo Liscovsky

Universidad Tecnológica Nacional  
Facultad Regional Del Neuquén  
Plaza Huincul, Argentina  
gmonte@frn.utn.edu.ar

**Resumen**—Este trabajo presenta una propuesta de un marco algorítmico para procesar señales de sensores en forma universal. Las señales provenientes de los sensores en un sistema de instrumentación y control es la información básica a través de la cual se infieren todos los estados operativos. Desde que la inteligencia emigró hacia el punto de medición, se han realizado acciones tendientes a normalizar el proceso de conexión de sensores. Como un resultado importante nació la norma IEEE 1451. Esta norma permite automatizar el proceso incorporando parámetros operativos y en forma electrónica en el mismo transductor. El estándar simplifica la incorporación de sensores a un sistema normalizando la forma que los sensores se describen asimismo. De la misma manera que se concibió la norma, el análisis de la señal del sensor debería también estar normalizado. Un tratamiento universal de la señal sensorial posibilita un dialogo entre sensores y que el sensor se de cuenta del tipo de señal obtenida con el objetivo de realizar inferencias de comportamiento. En el trabajo se describe técnicas realizadas en el dominio temporal que forman una plataforma fundacional para el análisis de la señal del sensor. Los algoritmos están basados en segmentación y etiquetado. La señal es vectorizada y analizando la secuencia de las clases de segmentos se infieren comportamientos, tendencias y predicciones sobre ella. El costo computacional es muy bajo lo que posibilita que estos algoritmos se ejecuten en tiempo real y que se incluyan en el sensor. Se presentan numerosas aplicaciones y un ejemplo práctico de un sensor predictivo de temperatura implementado en un microcontrolador.

**Keywords**- IEEE 1451, fusión sensorial, segmentación de señales, tiempo real, interpolación, normalización.

## I. INTRODUCCION

Actualmente todo sistema o dispositivo lleva el prefijo “inteligente”. La principal razón de la inteligencia es que realmente necesitamos dispositivos inteligentes para interactuar con la complejidad de los sistemas que los seres humanos estamos creando. Para abordar esta complejidad, nace la tecnología “Plug and Play” como una consecuencia natural. Los sensores no son la excepción de la regla. Ellos se convirtieron en inteligentes hace aproximadamente treinta años, en el sentido que incluyen inteligencia en el punto de medición junto con la transmisión digital normalizada de los datos. Los sensores son los componentes más importantes en todo sistema de monitoreo y control, ellos miden parámetros físicos y los convierten en señales. Sin información confiable

de los sensores, los sistemas no pueden tomar las decisiones correctas.

Al convertirse los sistemas en una red de sensores inteligentes surgió la necesidad de normalizar la conexión. Como consecuencia, se concibió la norma IEEE 1451.X, que aún hoy se encuentra en estado de evolución [1] [2]. La norma define todo lo necesario para automatizar la conexión de un sensor a un sistema. El sensor incorpora en si mismo información para que el sistema lo reconozca automáticamente, además de autocalibración, transmisión normalizada de la información y hoja de datos electrónica, con lo que el sensor se transforma en un dispositivo “Plug and Play” [3].

La creación de la norma IEEE 1451 fue un gran avance, pero ¿Son los sensores inteligentes lo suficientemente inteligentes? El principal objetivo de la creación de la norma fue automatizar la conexión – desconexión de un sensor, aun de distintos fabricantes. Sin embargo, la principal información que proporciona un sensor es el valor de la señal que esta traduciendo. Más aún, la información que debe proporcionar un sensor al sistema u otros sensores tiene que incluir mucho más que el valor de la señal. Tendría que incluir, tendencia, forma, futuros valores y reconocer patrones típicos y anormales. El sensor debe ser lo suficientemente inteligente para darse cuenta del tipo de señal que esta midiendo para marcar porciones de la señal que el objeto de analizarla más profundamente. El conocimiento de la señal le posibilita detectar comportamientos anormales o asociar una “verosimilitud” al valor o comportamiento de la señal.

Además de requerir de dispositivos “Plug and Play”, los sistemas complejos necesitan ser entendidos. Existe una explosión de interés en minería de datos aplicada a series temporales, incluyendo datos de sensores [4] [5]. También, se han aplicado tecnologías inteligentes para inferir comportamientos a partir de datos sensoriales, tales como redes neuronales artificiales, lógica difusa y algoritmos genéticos. Todos estos procesos tienen un factor en común: la abstracción y la representación de la señal.

El procesamiento de la señal de los sensores es un campo muy activo. Existen infinidad de tratamientos y algoritmos pero sufren de dos particularidades. Primero son procesamientos dedicados a señales específicas o provenientes de algún sensor en particular, por lo tanto no se pueden aplicar a todas las señales de los sensores. Segundo, la mayoría de los algoritmos sobre señales de sensores se encuentran protegidos por patentes y/o son confidenciales.

Resumiendo, hemos detectado que no existe un tratamiento universal de la señal de los sensores que permita una síntesis y análisis de la señal, un dialogo entre ellos para permitir fusión sensorial y que proporcione una plataforma para una minería de datos u otros procesos de orden superior. En este trabajo se brindan herramientas para analizar y conocer las señales de los sensores de una manera eficaz y sintética a través de secuencias de comportamientos elementales que permiten inferir un comportamiento global. En las siguientes secciones se expone el procesamiento de la señal de los sensores en el dominio temporal. Se comienza con una segmentación en tiempo real basada en interpolación lineal adaptiva sobre una señal sobremuestreada. Se clasifican los segmentos en ocho clases y se analizan la secuencia de las clases de segmentos. Se logra una abstracción de señales de sensores desde un enfoque universal logrando una plataforma unificada basada en una estructura en capas. En la capa de orden superior es tal el grado de abstracción de la señal real, que permite en una forma simplificada inferir condiciones y estados para cualquier tipo de señal. La fortaleza del proceso es la universalidad, lo cual garantiza que las aplicaciones sean innumerables. La estructura en capas podría ser el origen de un estándar para señales sensoriales. En la sección III se ejemplifica el uso de los algoritmos propuestos desarrollando un sensor de temperatura basado en microcontrolador que predice el valor futuro del estado estacionario de la medición.

II. DESCRIPCION DE LOS ALGORITMOS

Consideremos a  $x(t)$  una señal analógica proveniente de un sensor arbitrario como se observa en la Fig. 1 En  $x(t)$  se encuentra embebida la información del sistema físico asociado a él. Dado un instante de tiempo  $t_o$ , la información importante es  $x(t_o)$  pero además toda la trayectoria de la señal hasta llegar a ese instante de tiempo y cuanto más cercana a  $t_o$ , más valiosa es la información.

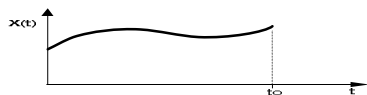


Figura 1. La información que proporciona un sensor en un instante de tiempo  $t_o$  es la señal en ese instante junto con la trayectoria hasta llegar a ese valor.

El sensor debe ser lo suficientemente inteligente como para poder estimar la función de la trayectoria de la señal hasta un determinado valor. El objetivo principal de los algoritmos propuestos es que el sensor estime la señal como lo haría un operador humano al observar la trayectoria de la señal. En la

Fig. 2 se observan señales típicas que son fácilmente interpretadas por un humano, pero son un desafío para un sensor inteligente.

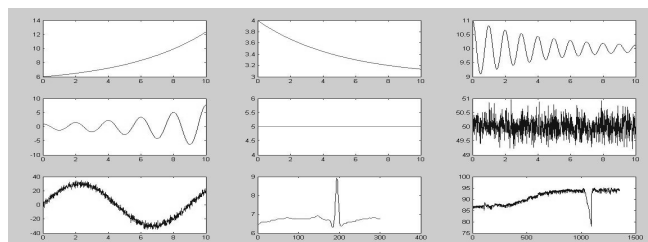


Figura 2. El sensor inteligente debe reconocer las distintas señales de la misma forma que un observador humano lo hace.

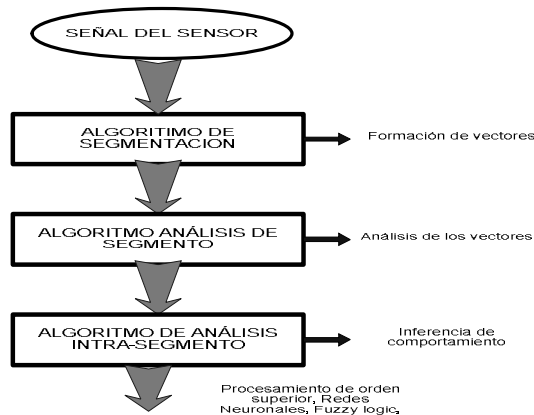


Figura 3. Estructura en capas de los algoritmos sobre la señal sensorial..

A. Algoritmo de segmentación

La segmentación es el proceso inicial a partir de cual se construye la abstracción de la señal. El algoritmo de segmentación de la señal se basa en el grado de redundancia que puede ser observada en una señal de un sensor sobremuestreada. Este tipo de análisis de la segmentación fue tratado previamente en [6]. En un proceso de muestreo uniforme, no todas las muestras poseen el mismo grado de importancia [7]. La segmentación se basa en el grado de alejamiento de la trayectoria de la señal con respecto a una lineal. Considerando, como una primera aproximación, que una muestra no conlleva información importante, es substituida por una muestra obtenida por interpolación lineal. Esta operación genera un error, o sea, la diferencia entre la muestra real y la interpolada. Cuando se supera un umbral de error, las muestras no interpoladas son marcadas como significativas y almacenadas en un vector. También se guarda la posición de la muestra en otro vector. Se considera que estas muestras sintetizan a la señal. El algoritmo comienza interpolando la muestra entre dos muestras, o sea, se interpola la muestra 2 de 1 y 3. Luego se calcula el error como la diferencia entre la muestra interpolada y la real. Si el error es menor que un valor prefijado, la muestra no tiene importancia relativa ya que su valor puede obtenerse de sus muestras vecinas. Luego, la muestra 3 es interpolada desde la muestras 1 y 5, la 4 de 1 y 7 y así siguiendo. Dos clase de errores son calculados, primero el error de reemplazo y segundo el error que resulta de la suma de

los errores de reemplazo dentro del segmento. Si el error supera un umbral, la muestra es marcada como significativa, y el proceso comienza nuevamente desde esta muestra.

La Fig. 4 presenta el pseudocódigo. Sea  $x(n)$  la señal del sensor muestreada uniformemente a una frecuencia de un orden de magnitud superior a la de Nyquist. Un factor importante de sobremuestreo es necesario si se ejecuta el análisis de la forma del segmento como es mostrado en la sección siguiente. El sensor tiene acceso a la señal en su origen, por lo tanto es viable muestrearla arbitrariamente. Si no es posible sobremuestrear, se procede a filtrar la señal mediante un filtro FIR pasabajos de fase lineal para lograr redundancia entre las muestras. El sobremuestreo o el filtrado son necesarios para que el segmento quede determinado por la forma de la señal y no por acumulación de errores producida por la digitalización. Sea  $\mathcal{E}_{\max}$  el error acumulativo permitido y  $\mathcal{E}_i$  el error de interpolación. El segmento culmina cuando se alcanza alguna de las dos cotas de error. Si ninguna de las dos condiciones de errores fueran encontradas para  $N$  muestras consecutivas, la última muestra es marcada como importante y el segmento se considera terminado. Este es el caso cuando la señal experimenta un comportamiento lineal por un período prolongado.

Como resultado del algoritmo se obtienen dos vectores, el vector de muestras importantes  $mark(n)$  y el vector de índices temporales  $timepos(n)$  que marca el instante de ocurrencia de las muestras importantes en módulo  $N$ . El algoritmo es extremadamente simple, aún así permite una segmentación en tiempo real de la señal y una plataforma para el algoritmo de análisis de la forma de la señal del segmento. Solamente una división por dos es necesaria en el cálculo del promedio y en el subíndice del vector de posición.

```

//Inicialización:
k = 2
n = 1
etotal = 0 ; error acumulativo
mark(n) = x(n) ; vector de muestras marcadas
timepos(n) = n ; vector de índices temporales
//Parámetros:
E_max ; max. error acumulativo
E_i ; max. error de interpolación
N ; max. tamaño de segmento
//Procedimiento:
DO
    error=abs ((x(n)+x(n+k))/2 - x(n+k/2))
    etotal=etotal + error
    IF (error < E_i AND etotal < E_max AND k<N)
        k=k+2
    ELSE
        n=n+k/2
        mark(n)=x(n)
        timepos(n)=n
        k=2
        etotal=0
    END
LOOP
    
```

Figura 4. Algoritmo de segmentación LIN1

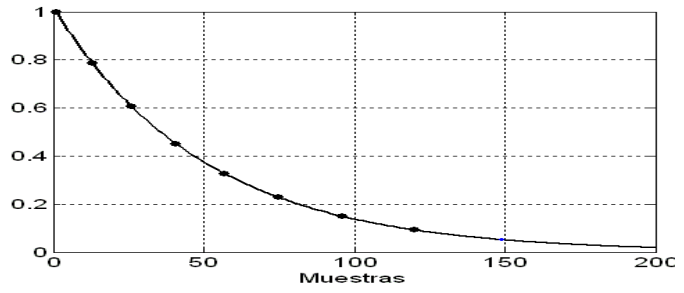


Figura 5. Muestras importantes como resultado del algoritmo LIN1 aplicado a una señales testigo con  $\mathcal{E}_i = 0.0.5$  y  $\mathcal{E}_{\max} = 0.1$ .

Es importante destacar que la señal de la Fig. 5 puede ser representada por dos vectores de nueve elementos en los cuales se encuentra embebida la información de la señal. Estos vectores son:

```

mark[n]={1,0,78,0,6,0,45,0,32,0,23,0,15,0,09,0,05}
timepos[n]={1,13,26,41,57,75,96,120,149}
    
```

Mediante estos vectores es posible además, reconstruir la señal con un aproximación dada por las cotas de error  $\mathcal{E}_i$  y  $\mathcal{E}_{\max}$ .

B. Algoritmo de análisis de segmentos

Con este proceso comienza el primer nivel de abstracción de la señal. Una vez que la señal ha sido procesada por el algoritmo LIN1 un segundo proceso llamado Análisis de Segmento SA puede realizarse. El principal motivo de la realización de LIN1 es que proporciona información como datos de entrada para el algoritmo SA.

El objetivo de este algoritmo es el de analizar el comportamiento de la señal dentro de cada segmento. En este proceso cada segmento es considerado independiente de los demás. En las siguientes secciones se analizan las relaciones entre los segmentos.

La Fig. 6 muestra el diagrama de flujo del algoritmo SA. Es un algoritmo eficiente que incluso se calcula en el algoritmo LIN1 dentro del cálculo de los errores. La operación básica es sobre el signo entre las muestras reales y las interpoladas como se observa en (1).

$$dif(i) = \sum_{k=k1}^{k2} sign[(x^*(n) - x(n))] \quad (1)$$

Donde  $x^*(n)$  es el vector de muestras interpoladas;  $k1, k2$  los índices de las muestras izquierda y derecha del segmento  $i$  respectivamente y  $k2-k1$  es el largo del segmento. Asumiendo que la señal fue muestreada a una frecuencia mucho más alta que la de Nyquist, la señal del segmento puede clasificarse en ocho clases de forma como se muestra en la Fig. 7.

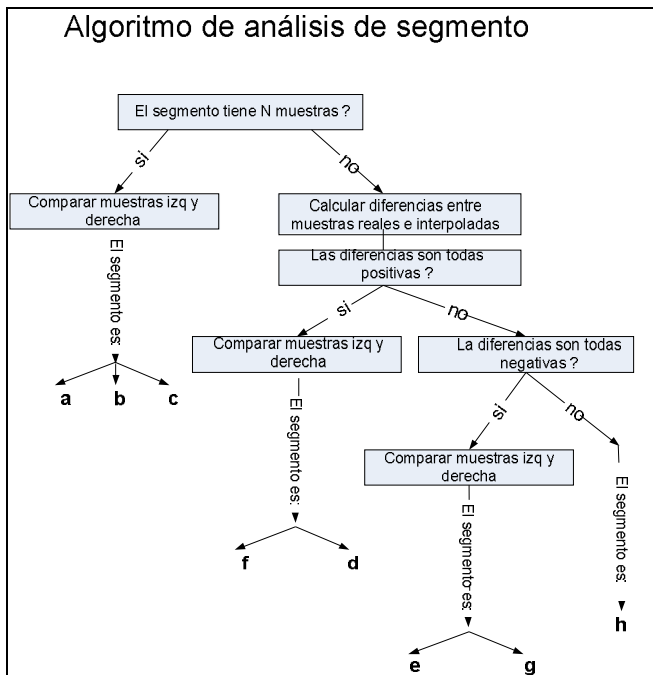


Figura 6. Algoritmo de clasificación de segmentos en ocho clases a.,b,c,d,e,f,g,h.

Las clases a, b y c son el resultado del límite máximo de N muestras por segmento. Es decir, se termina el segmento por haber alcanzado la máxima longitud preestablecida. Por lo tanto, en estos tres casos, solo es necesario calcular la diferencia entre la muestra izquierda y derecha. Para el resto de las clases, la diferencia entre las muestras interpoladas y las muestras reales junto con la diferencia entre las muestras izquierda y derecha son los elementos que determinan su clasificación.

Mediante la segmentación de la señal y la clasificación de los segmentos, la señal del sensor puede ser analizada. Con las técnicas descritas, la señal se representa ahora por los tres vectores: *mark(n)*, *timepos(n)* y el vector de clases; *clases(n)*.

a		constante
b		lineal +
c		lineal (-
d		+exp+
e		(- exp+
f		c+exp(-
g		c-exp(-
h		ruidoso

Figura 7. Las ocho clases de segmentos discriminados.

Para la señal de la Fig. 5, por ejemplo, el vector de clases es: resultante es [f f f f f f f f].

C. Algoritmo de análisis intrasegmento

Los dos algoritmos previos permiten la representación de cualquier señal de sensor con simplicidad, a la vez que capturan la información embebida en ella. Para analizar más profundamente la señal, es necesario analizar la secuencia de segmentos, teniendo en cuenta la duración y la clase de ellos. La tipificación de secuencias de segmentos proporciona información acerca del comportamiento macroscópico de la señal. Por ejemplo, con reglas IF... THEN, el sistema podría inferir conclusiones que sobre la forma de la señal o de la tendencia. Por ejemplo, si se observa una secuencia de segmentos "g" de longitudes crecientes entonces la señal experimenta un comportamiento exponencial y se esta estabilizando en un valor constante. Si por el contrario se observa una secuencia de segmentos "d" de longitudes decrecientes, la señal del sensor esta creciendo descontroladamente. Si se detectan secuencias consecutivas tipo "h", implica la presencia de ruido o la rotura de un sensor debido a presencia de alta frecuencia. A continuación se analizan algunos casos encontrados en el mundo real.

a) Detección de señales sinusoidales

El comportamiento sinusoidal es identificado con la secuencia de segmentos "gefd" en donde repeticiones de mismo tipo de segmento están permitidas. Por ejemplo, "gggeeffdd" también es un patrón válido. Los máximos locales ocurren en la unión de segmentos "g" y "e" y los mínimos locales en la unión de "f" y "d", por lo tanto, una estimación de la frecuencia de la señal puede ser fácilmente calculada. Si el sensor calcula la evolución de estos máximos y mínimos puede determinar si las oscilaciones son crecientes o decrecientes. Dependiendo de la cota de error puede ocurrir un segmento "h" en el cambio de tendencia que se descarta. En la Fig. 8 se observa un ejemplo de este tipo de señal.

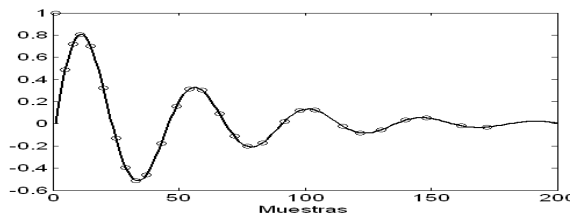


Figura 8. Señal oscilatoria decreciente identificada por la secuencia GGEEFFDDGGEEFFDHGGHFDHGHF y observando la trayectoria de los máximos y mínimos.

b) Detección de formas exponenciales

La ocurrencia de segmentos consecutivos tipo "g", "f", "d" o "e" indican una forma de onda exponencial de la señal del sensor. Por ejemplo, segmentos consecutivos tipo "g" de longitudes crecientes indican un comportamiento exponencial tendiente a estabilizarse a un valor de estado estacionario. Este valor se puede predecir basándose en los cambios de pendiente en los puntos extremos de los segmentos y es importante en sistemas de control [8]. La Fig. 9 muestra este tipo de comportamiento.

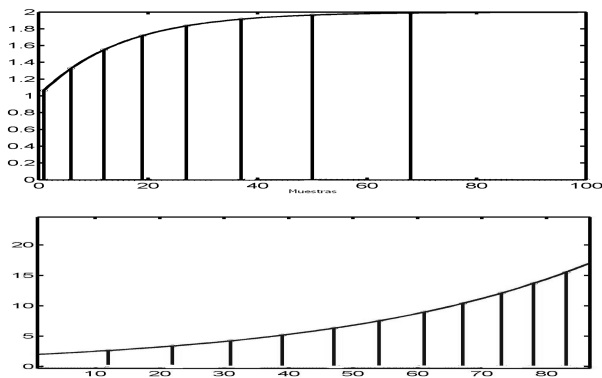


Figura 9. Superior: comportamiento exponencial estable con segmentos tipo “g” de longitudes crecientes. Inferior: comportamiento exponencial inestable de dado por segmentos tipo “d” de longitudes decrecientes.

c) *Detección de ruido impulsivo*

El ruido impulsivo es rápidamente detectado por una abrupta reducción de la longitud de los segmentos como se observa en la Fig. 10.

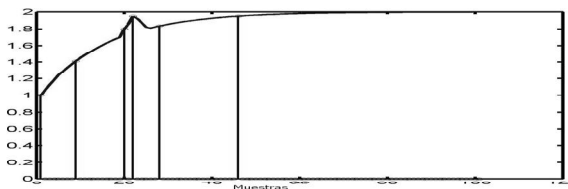


Figura 10. La ocurrencia de ruido impulsivo es detectada por una abrupta reducción en la longitud de los segmentos.

d) *Estimación de la media y la varianza*

La señal se representa por el valor de las muestras en los extremos de los segmentos. Las muestras que fueron eliminadas son una combinación aproximadamente lineal de las de los extremos. Por lo tanto ellas no agregan “aleatoriedad” y la media y la varianza se pueden aproximar eficientemente empleando en el cálculo solamente los valores extremos de los segmentos.

e) *Determinación del muestreo óptimo*

Para evitar el aliasing en la operación del muestreo, se debe emplear un filtro pasabajos en la entrada. En general el ancho de banda es dinámico y responde a estímulos físicos sobre el transductor. Al fijar una frecuencia de muestreo para poder capturar todos los eventos, ocurrirá que ella es excesiva la mayor parte del tiempo. En el esquema propuesto es posible encontrar la frecuencia de muestreo óptima. Para un error dado de interpolación, la frecuencia optima de muestreo es la frecuencia de sobremuestreo dividida por la cantidad de muestras que posee el segmento más corto.

f) *Detección de ruido*

Una determinada cantidad de segmentos tipo “h”, podrían indicar la presencia de ruido sobre la señal o la rotura del sensor.

g) *Determinación de la tendencia a largo plazo*

La información de la tendencia a largo plazo de una señal se encuentra en la región de bajas frecuencias. Se procede a

remover el ruido impulsivo, a filtrar la señal mediante un filtro FIR de fase lineal para no alterar la forma y a continuación se ejecutan los algoritmos. Se espera obtener segmentos de mayor duración para luego analizar la secuencia obtenida.

h) *Detección de patrones complejos*

Otras reglas pueden ser establecidas en función de la aplicación. Por ejemplo se pueden detectar tipos específicos de ruido impulsivo y forma de la señal. Como ejemplo se analizó una señal de ECG como se observa en la Fig. 11. Obsérvese que la secuencia “fdfd” codifica el complejo QRS de la señal y que el valor máximo (R) es encontrado en la secuencia “df” más larga. Es importante destacar que la detección es insensible a ruido de 50 Hz por ejemplo.

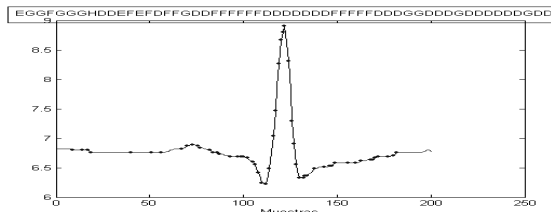


Figura 11. Tipificación de una señal típica de ECG con  $\epsilon_i = 0.001$  y  $\epsilon_{max} = 0.001$ .

III. RESULTADOS EXPERIMENTALES

Los algoritmos LIN1 y SA fueron escritos en lenguaje de programación para MatLab. Luego fueron implementados en “C” para microcontroladores de propósitos generales y DSP.

Para testear los algoritmos se procesaron señales de sensores reales. En primer lugar se procesaron datos de una planta separadora de gas ubicada en Loma La Lata en la provincia de Neuquén, Argentina. En la figura 12 se observa la temperatura de un reactor para un periodo de muestreo de un minuto. Como la señal no fue muestreada a una frecuencia necesaria para el presente proceso, se procedió a filtrarla mediante un filtro FIR de 50 coeficientes y frecuencia normalizada de corte 0.1 como se observa en la Fig. 13.

Se realizó el análisis a la señal filtrada con una cota de error tendiendo a cero para eliminar la ocurrencia de segmentos tipo “h”. Los máximos locales de la señal son detectados por la secuencia de segmentos “DF” o “GE”. Los mínimos locales son caracterizados por las secuencias “EG” o “DF”.

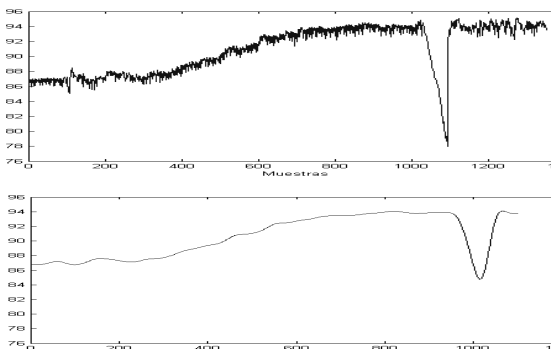


Figura 12. Señal de temperatura de un proceso de separación de gas y filtrada mediante un filtro FIR de orden 50 y frecuencia de corte normalizada 0.1.

La importancia relativa del máximo o mínimo será acotada por la cantidad de segmentos consecutivos que componen el patrón. Por ejemplo, la caída abrupta de la señal es detectada por la ocurrencia de una cantidad excesiva de segmentos consecutivos tipo “f” seguidos de segmentos consecutivos tipo “d”.

Una vez que el sensor se da cuenta de la forma de la señal, puede efectuar predicciones del valor estacionario. Se implementó un sensor de temperatura basado en microcontrolador que predice el valor detectando secuencias consecutivas “g” o “d”.

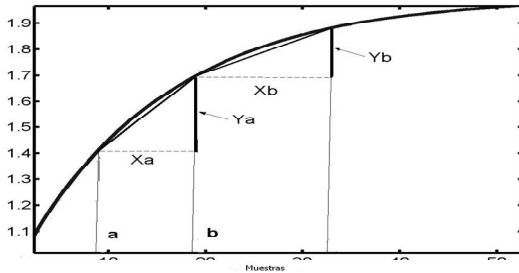


Figura 13. Señal con secuencia de segmentos “g”. Empleando los extremos de los segmentos para estimar las pendientes es posible predecir el valor de estado estacionario.

La expresión general de la exponencial en la Fig 13 es:

$$x(t) = X_{ss} - (X_{ss} - X_i) e^{-t/T} \quad (2)$$

Donde  $X_{ss}$  es el valor en estado estacionario y  $X_i$  el valor inicial. La derivada de  $x(t)$  en  $t=0$  es:

$$\left. \frac{dx}{dt} \right|_{t=0} = \frac{X_{ss} - X_i}{T} \quad (3)$$

La pendiente de los segmentos puede emplearse como la aproximación de la derivada en (3) aplicada en los puntos **a** y **b**. Si consideramos la derivada en el punto **a** y empleando este punto como valor inicial obtenemos:

$$\frac{X_{ss} - X_{ia}}{T} = M_a \quad (4)$$

Donde  $M_a = Y_a/X_a$  es la aproximación de la pendiente en el punto **a**. Repitiendo el análisis ahora en el punto **b**, y empleando este punto como valor inicial obtenemos:

$$\frac{X_{ss} - X_{ib}}{T} = M_b \quad (5)$$

Donde  $M_b = Y_b/X_b$  es ahora la aproximación de la pendiente en el punto **b**. Resolviendo (4) y (5) obtenemos (6).

$$X_{ss} = \frac{M_a X_{ib} - M_b X_{ia}}{M_a - M_b} \quad (6)$$

Por lo tanto, el sensor predice el valor futuro de la medición cuando pasa de una temperatura inicial a una final.

#### IV. CONCLUSIONES

Los análisis propuestos brindan una variedad de parámetros que extraen importante información embebida de la señal del sensor. Estos algoritmos son válidos para cualquier señal sobremuestreada o filtrada para lograr correlación. Un alto grado de simplificación es logrado con la segmentación de la señal y la tipificación en ocho clases. Este es el proceso fundamental que permite desarrollar algoritmos más sofisticados ya que es posible darse cuenta que tipo de señal se encuentra presente. A modo de ejemplo, algunos de estos algoritmos fueron presentados, pero otros podrían ejecutarse en el sensor o en el sistema de control y monitoreo. Además, estos algoritmos proporcionan una excelente plataforma para un proceso de minería de datos ya que brindan una abstracción de la señal como entrada a un software de inteligencia artificial. Los algoritmos detectan regiones de la señal donde tiene y donde no tiene sentido aplicar otros procesos para analizar más profundamente la señal. Por lo tanto es ideal para marcar subconjuntos de la señal para otros procesos. Un concepto fundamental en un sistema de control es que para reducir falsas alarmas es importante reducir la varianza del estado normal [9]. ¿Como reducimos la varianza del estado normal?, conociendo el sistema más en profundidad Este conocimiento se basa en la interacción entre las señales de los sensores involucrados en el proceso.

Los resultados de este trabajo enfatizan el hecho que bajo la misma necesidad que fue creada la norma IEEE 1451 para sensores inteligentes, se debería normalizar procesos de análisis de señales para suplementar la información proporcionada por un sensor inteligente al sistema de control. Esta normalización permitirá un dialogo entre sensores facilitando la fusión sensorial.

#### REFERENCIAS

- [1] R. Wynn. “ The TEDS smart sensor revolution”. *IEEE Computing and Control Engineering*. August/September 2004. Vol 15, Issue 4, pp 25-27.
- [2] B. Betts. “ Smart sensors”. *IEEE Spectrum on line*. April 2006.
- [3] P. Cleaveland. “What is a smart sensor”. *Control Engineering*. January 1, 2006.
- [4] E. Keogh, Chu,S. and Hart,D. “An online algorithm for segmenting time series”. *Proceeding IEEE International Conference on Data Mining ICDM 2001*. pp 289-296.
- [5] JE. Keogh, M Pazzani “An Enhanced Representation of Time Series Which Allows Fast and Accurate Classification, Clustering and Relevance Feedback”. *Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98)* . pp 239-241, 1998.
- [6] Monte, G. “Sensor Signal Preprocessing Techniques for Analysis and Prediction” *Industrial Electronics, 2008. IECON 2008. 34th Annual Conference of IEEE*. pp 1788-1793 ISBN 978-1-4244-1766-7.
- [7] Marten D. van der Laan. “*Signal sampling techniques for data acquisition in process Control*”. Thesis Rijksuniversiteit Groningen. - 1995. ISBN 90-367-0502-9.
- [8] F. Bertling, S. Soter, “Real-time prediction of the steady state temperature of circuit components as a tool for power electronic circuit testing”. *PCIM Europe 2007 Conference, Nuremberg, Germany*.
- [9] Barford Paul, Jha Somesh. “Fusion and Filtering in Distributed Intrusion System”, *Proceedings of the 42nd Annual Allerton Conference on Communication, Control and Computing, 2004*.