



PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA

Señales lineales a trozos para filtros de orden analógicos

*Trabajo de grado presentado al Departamento de Ingeniería Electrónica como
requisito para obtener el título de Ingeniero Electrónico*

PRESENTADO POR:
DIEGO FERNANDO DÍAZ PADILLA

DIRECTOR:
ALFREDO RESTREPO PALACIOS (Ph.D.)
CODIRECTOR:
JESÚS ALONSO OCHOA ARANGO (Ph.D.)

COLOMBIA
2 DE JUNIO DE 2022

*A mis padres Lida y Fernando,
a María José y Nicolás,
a la memoria Ulises.*

Agradecimientos

Agradezco profundamente a mis padres por su amor y compañía, por apoyar todos y cada uno de mis sueños. A María José, por ser la mejor hermana que alguien pueda tener. A mi hermano Nicolás, por siempre estar para nosotros.

Al profesor Pedro Vizcaya, por sus amables sugerencias y observaciones. Al profesor Jairo Hurtado, por su orientación y gran labor como profesor. A ellos dos les agradezco haber sembrado en mí un gran interés por el procesamiento de señales.

A mi director Alfredo Restrepo, por transmitirme su pasión, curiosidad, dedicación y creatividad, por sus sabios consejos y guiarme si no entendía algo, a él toda mi gratitud y admiración.

A mi codirector Jesús Ochoa, por siempre mostrarme nuevas formas de ver las cosas, por guiarme y ayudarme a entender muchos aspectos teóricos, también por acompañarme en este universo un tanto lejano de los grupoides y algebroides de Lie.

A mis amigos de toda la vida Sebastián y Daniel García. A Felipe, Laura, Daniel Pineda, Dairon, Daniel Morantes, Nicolás, Hader, Sebastián De La Cruz. A todos mis amigos de Electrónica y Matemáticas por enseñarme el valor de la amistad. Y a Helen, por supuesto, por su cariño, dulzura y soporte, por estar conmigo en los momentos más difíciles.

Resumen

Los filtros de estadísticas de orden son filtros discretos no lineales de ventana móvil que han encontrado aplicaciones en procesamiento de señales e imágenes. Los filtros de orden analógicos, una generalización de los filtros de estadísticas de orden al caso continuo, hasta ahora han tenido un interés principalmente teórico debido a la complejidad de sus implementaciones; estos recaen en una noción de ordenamiento para señales de parámetro continuo. Este trabajo gira alrededor de restringir la entrada de los filtros de ranqueo, que son una clase de filtros de orden analógicos, a señales lineales a trozos para hacer su implementación a nivel algorítmico realizable. Para ello, se describe con detalle el proceso de ordenamiento de estas señales, que consiste en calcular una distribución y luego el ordenamiento o inversa generalizada; se presentan definiciones y se prueban teoremas con varias propiedades de las dos funciones. Se demuestra que esta nueva noción de orden, cuando es aplicada a señales lineales a trozos, produce también señales lineales a trozos completamente caracterizadas, lo que hace posible el diseño de un algoritmo capaz de realizar el ordenamiento.

Se definen formalmente los filtros de orden analógicos, se estudian diferentes funcionales que resultan en el promedio móvil, combinaciones lineales y los filtros de ranqueo. Utilizando el resultado del ordenamiento de señales lineales a trozos se demuestra que cuando la entrada de los filtros de ranqueo es lineal a trozos entonces la señal de salida también lo es; se realiza un análisis para el cálculo de los puntos característicos, llamados *vértices*, de la señal de salida, y con esto se diseñan e implementan algoritmos para el cálculo de dichos puntos: algoritmo de ventana, funcional de ranqueo y cálculo de vértices de salida. Para los métodos implementados se ejecutaron protocolos de prueba y verificación. Por último, se presentan aplicaciones para suavizado de series de tiempo.

Índice general

Introducción	1
I Función distribución y ordenamiento monótono	4
1. Preliminares en teoría de la medida e integración	5
1.1. Medida de Lebesgue	5
1.2. Funciones medibles	7
1.3. La integral de Lebesgue y el espacio \mathcal{L}	7
2. Función distribución y ordenamiento monótono	10
2.1. Ordenamiento discreto: estadísticas de orden	10
2.2. Ordenamiento monótono	11
2.3. Ordenamiento de funciones lineales a trozos	14
II Filtros de orden analógicos	19
3. Filtros de orden analógicos vía funcionales	20
3.1. Filtros de estadísticas de orden	20
3.2. Generalización a los filtros de orden analógicos	21
3.3. Ranqueo, promedio móvil y otros funcionales	22
3.4. Filtrado de ranqueo de señales lineales a trozos	24
4. Algoritmo para filtros de ranqueo y aplicaciones	27
4.1. Algoritmo de ordenamiento	27
4.1.1. Protocolo de pruebas y resultados	28
4.2. Algoritmo de ventana móvil	30
4.3. Algoritmo para filtros de ranqueo	33
4.3.1. Protocolo de pruebas y resultados	33
4.4. Análisis de resultados y consideraciones prácticas	37
4.5. Aplicación de los algoritmos	38
5. Conclusiones	41

Apéndices	43
A. Algoritmos	43
A.1. Algoritmo de ordenamiento	43
A.2. Algoritmos para ventana móvil	44
A.2.1. Algoritmo de ventana	44
A.2.2. Algoritmo de funcional de ranqueo	44
A.2.3. Algoritmo de filtros de ranqueo con ventana móvil	45
A.3. Algoritmo de cálculo de vértices	45
B. Demostraciones	46
B.1. Prueba del teorema 2.2.1	46
B.2. Prueba del teorema 2.2.2	47
B.3. Prueba del teorema 2.2.3	49
Bibliografía	51

Introducción

En aplicaciones de procesamiento de señales e imágenes resulta conveniente poder reducir diferentes tipos de ruido y al mismo tiempo preservar estructura e información importante. Es por este motivo que estudiar la teoría de señales, sistemas y filtros se vuelve crucial. Ya que los filtros son, formalmente, operadores actuando sobre espacios de señales, se les puede dividir en dos grandes clases: lineales y no lineales. Existen dentro de estas dos clases los filtros de ventana móvil, cuya salida está dada por la aplicación de un funcional ψ a una versión truncada (ventaneada) de la señal de entrada x . Los filtros FIR, por ejemplo, son lineales de ventana móvil pues la salida y_n consiste en aplicar $\psi: \mathbb{R}^k \rightarrow \mathbb{R}$, ahora funcional lineal, al k -segmento $[x_n, x_{n-1}, \dots, x_{n-k+1}]$ de la señal de entrada, esto es,

$$y_n = \psi([x_n, x_{n-1}, \dots, x_{n-k+1}]), \quad \forall n \in \mathbb{Z}.$$

Es muy importante el tipo ruido con el que esté contaminada la señal. Si la señal y el ruido tienen espectros que se solapan de manera considerable, un sistema lineal de convolución no será efectivo en distinguir la señal del ruido. Para el caso particular del ruido impulsivo el *filtro mediana discreto*, propuesto originalmente en 1971 por J.W. Tukey en [1] como un método de suavizado, ha demostrado una gran utilidad dado que tiene la propiedad de preservar estructuras localmente monótonas [2]. Esta y otras propiedades del filtro mediana pueden extenderse a otros tipos de filtros invariantes, no lineales y de ventana móvil; en particular, los resultados de N. Gallagher, G. Wise y S.G. Tyan en [3] y [4] sobre las *señales raíz* del filtro mediana discreto (señales que son invariantes bajo la acción del filtro) formalizaron y establecieron una base teórica que permitió posteriores generalizaciones que incluyen los *filtros de orden ranqueado* (*filtros OR*) y los *filtros de estadísticas de orden* (*filtros L*) [5]. En este caso el funcional es la composición $\psi = \lambda \circ \mathbf{o}$, donde $\mathbf{o}: \mathbb{R}^k \rightarrow \mathbb{R}^k$ es un operador de ordenamiento y $\lambda: \mathbb{R}^k \rightarrow \mathbb{R}$ es un funcional lineal.

Se pueden encontrar varias aplicaciones de los filtros L y OR en diversos campos de la ingeniería, e.g., el procesamiento no lineal de señales e imágenes [6], la instrumentación electrónica, el procesamiento de voz, la compresión de datos y telecomunicaciones [7], siendo el filtro mediana discreto uno de los más predominantes actualmente.

En la literatura se han reportado varias implementaciones de circuitos analógicos para el filtro mediana discreto, filtros L y OR (por ejemplo, vea [8–12]). Gran parte de estas implementaciones están dirigidas a un procesamiento y filtrado más rápido, específicamente en procesamiento de imágenes en tiempo real donde son más necesitadas, pues suponen un ahorro de energía y espacio físico.

Basados en la necesidad anteriormente mencionada, J. Fitch et al. en [13] presentan de forma teórica un filtro analógico capaz de realizar la operación mediana sobre una señal

de parámetro (tiempo) continuo, una generalización al dominio analógico del filtro mediana discreto; este filtro se conoce como *filtro mediana analógico*. De igual forma que en el caso discreto, el filtro mediana analógico puede extenderse a casos más generales como los *filtros de orden-ranqueado analógicos* (*filtros de ranqueo*) y *filtros de orden analógicos* [14–16]. Estos filtros no lineales de ventana móvil requieren de una noción de orden para las señales de parámetro continuo semejante a aquella de las señales discretas; este ordenamiento está basado en la medida de Lebesgue y no en el concepto de permutación.

Los filtros de orden analógicos pueden llegar a tener ventajas significativas en comparación con las implementaciones actuales de los filtros discretos mediana, L y OR ya que trabajan con una señal de parámetro continuo directamente. Al ser también suavizadores de señales analógicas, esto los hace muy atractivos para aplicaciones en comunicaciones, sismología, meteorología y biomédica [17–19].

Revisión de la literatura. La teoría de los filtros de orden analógicos ha sido poco estudiada en comparación con el caso discreto de los filtros L; por ejemplo, la generalización natural de los filtros de estadísticas de orden al caso continuo es trabajada por A. Bovik y H.G. Longbotham en [16]. En este artículo se presentan posibles definiciones para versiones analógicas de los filtros de estadísticas de orden, y además se presenta un marco de trabajo para extender resultados del dominio discreto al continuo. Murk J. Bottema en [20] establece dos propiedades deterministas clave de los filtros mediana analógicos: la primera es que las señales raíz se caracterizan por una clase especial de funciones localmente monótonas, y la segunda es que la aplicación repetida de un filtro de mediana produce una señal raíz que conserva características significativas en los datos originales; ambos resultados son extensiones de propiedades conocidas para los filtros mediana discretos.

Paulo J. Ferreira en [14] introduce una interpretación más profunda y precisa del filtro mediana analógico utilizando el concepto de *ordenamiento monótono*, que es utilizado como una herramienta en análisis funcional. El proceso para ordenar una señal de parámetro continuo f es calcular una distribución T_f y con base en esta el ordenamiento f^* . Posteriormente, Ferreira en [15] muestra varias propiedades de la distribución y el ordenamiento monótono. Las posibles generalizaciones que resultan de actuar diferentes operadores en el ordenamiento no son trabajadas, así como también posibles implementaciones ya sea en software o en hardware.

Problemática. La complejidad teórica de los filtros de orden analógicos, la falta de un estado del arte amplio, su no linealidad y el hecho de que trabajan sobre señales de parámetro continuo, hacen que su implementación se dificulte, esta es la problemática general que tratará el presente trabajo. Más concretamente, enfocándose en los filtros de ranqueo, se presentan dos obstáculos claros: implementar el ordenamiento y realizar el filtrado con una ventana que en principio recorre un dominio continuo. Lo que se propone aquí es utilizar *señales lineales a trozos* (*señales LT*) a la entrada de los filtros de ranqueo, estas son señales que toman una forma lineal (polinomios de grado 1, no confundir con aplicación lineal) en intervalos abiertos. Pueden también caracterizarse por un conjunto de vértices

$$\{(x_1, y_1), \dots, (x_n, y_n)\}$$

que corresponden en su mayoría a puntos de cambio de pendiente. Esta restricción de las señales de entrada se propone principalmente por dos razones: primero, las señales de parámetro continuo se pueden aproximar por señales lineales a trozos y segundo, restringir la entrada

de estos filtros a ese tipo particular de señales simplifica su implementación algorítmica. Así pues, este trabajo está dirigido principalmente a investigadores en el área de procesamiento no lineal de señales, y también a las diferentes industrias mencionadas que se están beneficiando de implementaciones de suavizadores analógicos y filtros tipo mediana.

Objetivos del trabajo. La finalidad principal es implementar algoritmos para los filtros de ranqueo con señales lineales a trozos de entrada. De acuerdo a esto se plantea el objetivo general:

- **Implementar de forma algorítmica los filtros de ranqueo analógicos con entradas de parámetro continuo lineales a trozos.**

Para conseguir este resultado se definió una ruta ascendente: primero se debe mostrar que el ordenamiento de señales lineales a trozos es lineal a trozos, así, con base en este resultado, se podrá diseñar un algoritmo para el ordenamiento; similarmente se debe mostrar que dada una señal lineal a trozos de entrada a un filtro de ranqueo, la señal de salida también es lineal a trozos, y con esto diseñar un algoritmo que calcule la señal de salida. Estas pautas serán precisamente los objetivos específicos:

1. Demostrar que el ordenamiento monótono de una señal lineal a trozos es lineal a trozos.
2. Diseñar e implementar un algoritmo para el ordenamiento monótono de señales lineales a trozos.
3. Demostrar que si la entrada de los filtros de ranqueo analógicos es lineal a trozos entonces su salida también lo es.
4. Diseñar e implementar un algoritmo para los filtros de ranqueo analógico con señales lineales a trozos de entrada.
5. Diseñar protocolos de prueba y verificación para los algoritmos.

Contribuciones y esquema del documento. Este trabajo se divide de la siguiente manera: el capítulo 1 recopila el material básico de teoría de la medida e integración que se empleará en capítulos posteriores, el lector familiarizado puede omitirlo. El capítulo 2 define la función distribución y el ordenamiento monótono; las contribuciones significativas de este capítulo son los teoremas 2.2.1, 2.2.2 y 2.2.3 que conciernen a propiedades de la distribución y el ordenamiento, y el teorema 2.3.1 correspondiente al primer objetivo específico. Estos dos capítulos constituyen la parte I. La parte II se divide de la siguiente forma: en el capítulo 3 se introducen con más detalle los filtros de estadísticas de orden para posteriormente definir los filtros de orden analógicos. La contribución primordial en este capítulo es el teorema 3.4.1 que corresponde al tercer objetivo específico. En el capítulo 4 se presenta el diseño, implementación y verificación de los algoritmos para el ordenamiento y filtros de ranqueo, incluyendo diferentes aplicaciones en series de tiempo. La contribución aquí son los algoritmos de la sección 4.1 y 4.3, junto con sus protocolos de pruebas. Finalmente, en el capítulo 5 se enuncian las conclusiones principales del trabajo. Detalles adicionales como pseudocódigos y demostraciones se encuentran en los apéndices.

Parte I

Función distribución y ordenamiento monótono

Capítulo 1

Preliminares en teoría de la medida e integración

En este capítulo se proporciona al lector las herramientas básicas de la teoría de la medida e integración en \mathbb{R} . En la sección 1.1 se define la medida exterior para todos los subconjuntos de los reales y se presentan sus propiedades básicas para luego introducir la medida de Lebesgue; en la sección 1.2 se definen las funciones medibles, y por último la sección 1.3 introduce las integrales de Lebesgue y Lebesgue-Stieltjes, junto con el espacio de funciones integrables y acotadas \mathcal{L} . Una explicación más detallada de estos conceptos se puede encontrar en [21–23].

Notación: los reales extendidos $\overline{\mathbb{R}}$ consisten de \mathbb{R} con los símbolos $-\infty$ y ∞ , la estructura algebraica y topológica usual, y la convención que $0 \cdot \pm\infty$ y $\pm\infty \cdot 0$ se definen como 0.

1.1. Medida de Lebesgue

Intuitivamente el tamaño de un subconjunto $A \subseteq \mathbb{R}$ puede aproximarse tomando la suma de las longitudes de una sucesión de intervalos abiertos cuya unión contenga a A , esto es, una colección $\{(b_n, a_n)\}_{n \in \mathbb{N}}$ tal que $A \subseteq \cup_{n \in \mathbb{N}} (b_n, a_n)$. Si se considera el conjunto de todas estas sumas y se toma su ínfimo, entonces se obtiene una definición razonable del tamaño (o medida) de A . Con base en lo anterior definimos la **medida exterior** de A por:

$$m^*(A) := \inf \left\{ \sum_{n=1}^{\infty} (b_n - a_n) : A \subseteq \bigcup_{n=1}^{\infty} (b_n, a_n) \right\}.$$

Formalmente la medida exterior es una función $m^*: \mathcal{P}(\mathbb{R}) \rightarrow [0, \infty]$, donde $\mathcal{P}(\mathbb{R})$ denota la colección de todos los subconjuntos de \mathbb{R} . Observe también que existen conjuntos con medida exterior infinita y cero, e.g., $m^*((0, \infty)) = \infty$ y $m^*(\{1\}) = 0$.

Teorema 1.1.1 ([21, Prop. 1.2]). *La medida exterior cumple las siguientes propiedades:*

1. $m^*(\emptyset) = 0$.
2. Todo subconjunto contable de \mathbb{R} tiene medida exterior 0.

3. Si A y B son subconjuntos de \mathbb{R} tales que $A \subseteq B$, entonces $m^*(A) \leq m^*(B)$.
4. Si $t \in \mathbb{R}$ y $A \subseteq \mathbb{R}$, entonces $m^*(A + t) = m^*(A)$. Esto nos dice que la medida exterior es invariante bajo traslaciones.
5. Si $\{A_n\}_{n \in \mathbb{N}}$ es una sucesión de subconjuntos de \mathbb{R} , entonces

$$m^*\left(\bigcup_{n=1}^{\infty} A_n\right) \leq \sum_{n=1}^{\infty} m^*(A_n).$$

Dados $A, B \subseteq \mathbb{R}$ disjuntos, en general no se cumple que $m^*(A \cup B) = m^*(A) + m^*(B)$. Con este fin se busca reducir la familia de conjuntos a una que cumpla esta propiedad, lo que motiva la siguiente definición:

Definición 1.1.1. Un conjunto $E \subseteq \mathbb{R}$ se dice **medible** (o también **Lebesgue-medible**) si para cada conjunto $A \subseteq \mathbb{R}$ se tiene que $m^*(A) = m^*(A \cap E) + m^*(A \cap E^c)$.

Si E es un conjunto medible se define la **medida de Lebesgue** de E (denota por $\mu(E)$) como su medida exterior; si \mathfrak{M} denota la colección de todos los subconjuntos medibles de \mathbb{R} , entonces la medida de Lebesgue es una función

$$\begin{aligned} \mu: \mathfrak{M} &\rightarrow [0, \infty] \\ E &\mapsto \mu(E) := m^*(E). \end{aligned}$$

La tupla $(\mathbb{R}, \mathfrak{M}, \mu)$ se conoce como el **espacio de medida de Lebesgue**. Esta medida es una generalización natural de la noción de longitud de un intervalo; cumple las propiedades del teorema 1.1.1 y además:

Teorema 1.1.2 ([21, Teorema 1.4]). Si $\{E_n\}_{n \in \mathbb{N}}$ es una sucesión de conjuntos medibles disjuntos dos a dos, entonces:

$$\mu\left(\bigcup_{n=1}^{\infty} E_n\right) = \sum_{n=1}^{\infty} \mu(E_n).$$

La medida de Lebesgue de un intervalo coincide con su longitud, la medida del conjunto $[0, 1) \cup [10, 12]$ es 3, la medida de $\bigcup_{n=1}^{\infty} [n, n + \frac{1}{2^n}]$ es $\sum_{n=1}^{\infty} \frac{1}{2^n} = 1$. Así como un intervalo puede tener medida infinita, por ejemplo, toda la recta real o el intervalo $[0, \infty)$, también la medida del conjunto $\bigcup_{n=1}^{\infty} [n, n + \frac{1}{n}]$ es $\sum_{n=1}^{\infty} \frac{1}{n} = \infty$.

Teorema 1.1.3 ([21, Prop. 2.6]).

1. Si $\{A_n\}_{n \in \mathbb{N}}$ es una sucesión de conjuntos medibles tales que $A_n \subseteq A_{n+1}$ para todo $n \in \mathbb{N}$ (monótona creciente), entonces

$$\mu\left(\bigcup_{n=1}^{\infty} A_n\right) = \lim_{n \rightarrow \infty} \mu(A_n).$$

2. Si $\{A_n\}_{n \in \mathbb{N}}$ es una sucesión de conjuntos medibles tales que $A_n \supseteq A_{n+1}$ para todo $n \in \mathbb{N}$ (monótona decreciente), entonces

$$\mu \left(\bigcap_{n=1}^{\infty} A_n \right) = \lim_{n \rightarrow \infty} \mu(A_n).$$

1.2. Funciones medibles

Similarmente a lo que ocurre con la medida de Lebesgue, la teoría de integración debe restringirse a una colección razonable de funciones con propiedades deseables, esta es la razón por la cual se introducen las funciones medibles.

Definición 1.2.1. Sea X un conjunto medible. Una función $f: X \rightarrow \overline{\mathbb{R}}$ se dice **medible** (o también **Lebesgue-medible**) si $\{x \in X: f(x) > \alpha\}$ es medible para todo $\alpha \in \mathbb{R}$.

En la práctica la gran mayoría de funciones son medibles; algunos ejemplos importantes son las funciones características y simples: la **función característica** de un conjunto medible $A \subseteq \mathbb{R}$ se denota por χ_A y está definida como

$$\chi_A(x) = \begin{cases} 1, & \text{para } x \in A, \\ 0, & \text{para } x \in \mathbb{R} - A, \end{cases}$$

y las **funciones simples** son combinaciones lineales de funciones característiscas de conjuntos medibles, i.e. son de la forma $f = \sum_{i=1}^n \lambda_i \chi_{A_i}$, donde $\lambda_i \in \mathbb{R}$ y cada A_i es medible.

Teorema 1.2.1 ([21, Prop. 3.5]). Sean f, g, f_n funciones medibles, $\lambda \in \mathbb{R}$ y φ una función continua en un conjunto abierto $G \subseteq \mathbb{R}$. Entonces las siguientes funciones son medibles (donde están definidas y sus dominios sean medibles):

1. $\lambda f, f + g, \max(f, g), \min(f, g), |f|, fg, f/g$;
2. $\sup f_n, \inf f_n, \limsup f_n, \liminf f_n$ y $\lim f_n$;
3. $\varphi \circ f$.

Para la teoría de integración de Lebesgue los conjuntos de medida cero son despreciados, por ende se utiliza el concepto de *en casi toda parte* (c.t.p.); por ejemplo, dos funciones $f, g: X \rightarrow \overline{\mathbb{R}}$ son iguales en μ -casi todas partes (μ -c.t.p.) si existe un conjunto N de medida cero tal que $f(x) = g(x)$ para todo $x \in X - N$; se interpreta de forma análoga en otros contextos.

1.3. La integral de Lebesgue y el espacio \mathcal{L}

A continuación, se define brevemente la integral de Lebesgue: si $s: X \rightarrow \overline{\mathbb{R}}$ es una función simple no negativa, podemos expresar

$$s = \sum_{j=1}^n \beta_j \chi_{B_j},$$

donde B_1, \dots, B_n son medibles disjuntos dos a dos y $\beta_1, \dots, \beta_n \in \mathbb{R}$. Defina su integral como

$$\int_X s \, d\mu := \sum_{j=1}^n \beta_j \mu(B_j).$$

Ahora, si $f \geq 0$ es una función medible sobre X , defina

$$\int_X f \, d\mu := \sup \left\{ \int_X s \, d\mu : 0 \leq s \leq f, s \text{ simple} \right\}.$$

Por último, para una función medible arbitraria $f: X \rightarrow \overline{\mathbb{R}}$, definimos su integral como la resta de dos funciones positivas, $f^+ = \max(f, 0)$ y $f^- = \min(-f, 0)$, es decir:

$$\int_X f \, d\mu := \int_X f^+ \, d\mu - \int_X f^- \, d\mu,$$

si al menos una de las integrales es finita.

La integral de Lebesgue permite introducir los **espacios de Lebesgue** L^p que son espacios de funciones medibles esenciales en la teoría de señales y análisis funcional; en este trabajo se tomará el caso $p = 1$. Se denotará por $\mathbf{L}^*(X)$ a la familia de funciones medibles en μ -c.t.p. de X para las que su integral de Lebesgue está definida; adicionalmente defina

$$\mathbf{L}^1(X) := \left\{ f \in \mathbf{L}^*(X) : \int_X f \, d\mu \in \mathbb{R} \right\},$$

y la norma L^1 de f

$$\|f\|_1 := \int_X |f| \, d\mu.$$

Toda la teoría clásica de integración de Riemann, Newton y Leibniz hace parte de la teoría de Lebesgue, mas esta última permite integrar una clase de funciones mucho más grande y sobre dominios más generales. Algunas propiedades elementales incluyen la linealidad y que preserva la relación de igualdad en casi toda parte, es decir para $f, g \in \mathbf{L}^1(X)$ se tiene que

$$\int_X (af + bg) \, d\mu = a \int_X f \, d\mu + b \int_X g \, d\mu, \quad \forall a, b \in \mathbb{R},$$

y por otro lado, si $g \in \mathbf{L}^*(X)$, f es medible y $f = g$ en μ -c.t.p., entonces $f \in \mathbf{L}^*(X)$ y también

$$\int_X f \, d\mu = \int_X g \, d\mu.$$

La teoría de los espacios normados se puede aplicar identificando funciones iguales en casi toda parte, es decir, a cada función $f \in \mathbf{L}^1(X)$ se asigna la clase de funciones

$$[f] := \left\{ g \in \mathbf{L}^1(X) : g = f \text{ } \mu\text{-c.t.p.} \right\},$$

y se define

$$L^1(X) := \left\{ [f] : f \in \mathbf{L}^1(X) \right\}.$$

Bajo las operaciones de suma y producto por escalar

$$[f] + [g] := [f + g], \quad \alpha[f] := [\alpha f], \quad \forall \alpha \in \mathbb{R},$$

y la norma $\|[f]\|_1 := \|f\|_1$, se tiene que $(L^1(X), \|\cdot\|_1)$ es un espacio normado, más aún es completo y por lo tanto un **espacio de Banach**. Es común no hacer una distinción entre L^1 y \mathbf{L}^1 . Por último, se centrará la atención además a las funciones acotadas en $\mathbf{L}^1(X)$ que denotaremos por $\mathcal{L}(X)$.

El siguiente teorema, que es un caso especial del teorema de Fubini, permite intercambiar el orden de integración:

Teorema 1.3.1 ([24, Teorema 10.40]). *Sean X, Y dos intervalos de \mathbb{R} , y sea k una función definida, continua y acotada en $X \times Y$; dados $f \in L^1(X)$ y $g \in L^1(Y)$ se tiene:*

$$\int_X f(x) \left[\int_Y g(y) k(x, y) dy \right] dx = \int_Y g(y) \left[\int_X f(x) k(x, y) dx \right] dy.$$

Sea g una función no decreciente en \mathbb{R} y $[a, b] \subseteq \mathbb{R}$ un intervalo compacto. Para una función $f \in \mathcal{L}([a, b])$ definimos su **integral de Lebesgue-Stieltjes** como

$$\int_a^b f dg := \inf \left\{ \sum_{i=1}^n c_i \Delta g_i : a = x_0 < x_1 < \cdots < x_n = b, c_i \geq f \text{ en } [x_i, x_{i+1}] \text{ } \mu\text{-c.t.p.} \right\},$$

donde $\Delta g_i := g(x_{i+1}^-) - g(x_i^+)$, a la función g se le denomina *semilla*. Esta integral, que se empleará más adelante en los filtros de orden analógicos, induce un funcional φ sobre $\mathcal{L}([a, b])$ dado por

$$f \mapsto \int_a^b f dg,$$

el cual llamaremos **funcional de Stieltjes**. Este funcional depende claramente de la elección de g ; por ejemplo, si f y g están ambas definidas y acotadas en $[a, b]$, y además g tiene derivada continua g' en $[a, b]$, entonces

$$\int_a^b f(x) dg(x) = \int_a^b f(x) g'(x) dx,$$

y si g es una función constante a trozos con discontinuidades en x_1, \dots, x_m y saltos $\alpha_1, \dots, \alpha_m$, y las discontinuidades de f y g no coinciden, entonces

$$\int_a^b f(x) dg(x) = \sum_{i=1}^m \alpha_i f(x_i).$$

Estas y otras propiedades pueden encontrarse en [24, Cap. 7].

Capítulo 2

Función distribución y ordenamiento monótono

En este capítulo se explora el ordenamiento monótono que es fundamental para los filtros de orden analógicos. En la sección 2.1 se presenta un método para ordenar vectores de números reales; la sección 2.2 extiende este procedimiento para funciones de parámetro continuo, es aquí donde se define la función distribución y el ordenamiento monótono, donde a su vez se enuncian varias propiedades. Por último, la sección 2.3 introduce las funciones lineales a trozos (LT) y se demuestra que el ordenamiento de estas funciones también es lineal a trozos (teorema 2.3.1). Las referencias guía son [15, 25].

2.1. Ordenamiento discreto: estadísticas de orden

Sean $k \in \mathbb{N}$ y $\mathbf{x} = [x_1, x_2, \dots, x_k] \in \mathbb{R}^k$ un vector arbitrario. Se quiere construir una versión ordenada del vector \mathbf{x} ; primero, para cada $z \in \{x_1, x_2, \dots, x_k\}$ se define $T_{\mathbf{x}}(z)$ por

$$T_{\mathbf{x}}(z) := \# \{n \in [k] : x_n \leq z\},$$

donde $\#$ denota la cardinalidad de un conjunto y $[k] := \{1, 2, \dots, k\}$. Ahora, para cada $n \in [k]$ considere

$$x_{(n)} := \min \{z \in \{x_1, x_2, \dots, x_k\} : T_{\mathbf{x}}(z) \geq n\}.$$

Si $x_i \leq x_j$ entonces $\{n \in [k] : x_n \leq x_i\} \subseteq \{n \in [k] : x_n \leq x_j\}$, luego tomando el cardinal se tiene que $T_{\mathbf{x}}(x_i) \leq T_{\mathbf{x}}(x_j)$, es decir $T_{\mathbf{x}}$ es monótona creciente. Más aún, dados $n, m \in [k]$ tales que $n \leq m$, se tiene que

$$\{y \in \{x_1, x_2, \dots, x_k\} : T_{\mathbf{x}}(y) \geq n\} \supseteq \{y \in \{x_1, x_2, \dots, x_k\} : T_{\mathbf{x}}(y) \geq m\},$$

tomando el mínimo se obtiene $x_{(n)} \leq x_{(m)}$, lo cual implica que el vector $[x_{(1)}, x_{(2)}, \dots, x_{(k)}]$ es la versión ordenada de \mathbf{x} ; este se conoce como el **vector de estadísticas de orden de \mathbf{x}** . Lo anterior es equivalente a encontrar una permutación $\sigma: [k] \rightarrow [k]$ tal que el vector permutado $[x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(k)}]$ es ordenado, i.e., $x_{\sigma(1)} \leq x_{\sigma(2)} \leq \dots \leq x_{\sigma(k)}$.

Ejemplo 2.1.1. Para el vector $\mathbf{x} = [6, 1, 3, 1, 7, 4] \in \mathbb{R}^6$ se obtiene: $T_{\mathbf{x}}(1) = 2$, $T_{\mathbf{x}}(3) = 3$, $T_{\mathbf{x}}(4) = 4$, $T_{\mathbf{x}}(6) = 5$ y $T_{\mathbf{x}}(7) = 6$. Se puede verificar que el vector ordenado es $[1, 1, 3, 4, 6, 7]$ (ver figura 2.1).

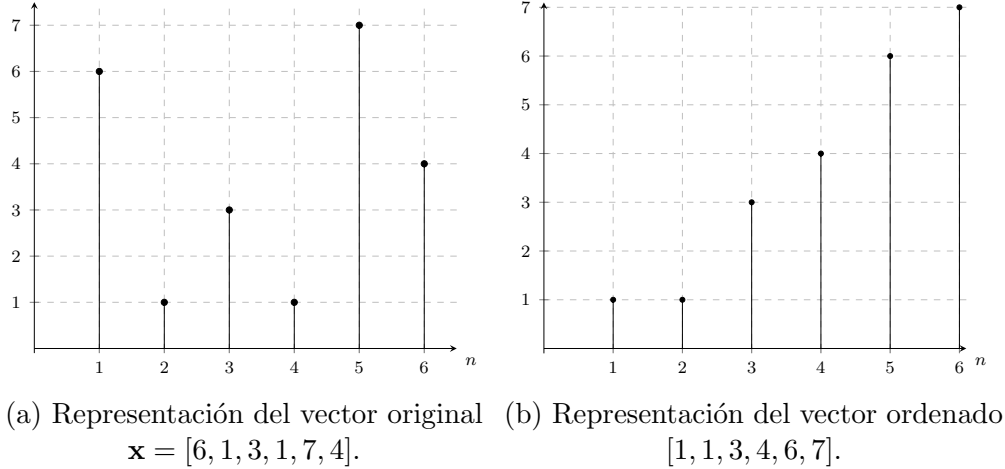


Figura 2.1: Ordenamiento de una 6-tupla.

2.2. Ordenamiento monótono

El concepto de ordenamiento de funciones de parámetro continuo fue tratado inicialmente por G.H. Hardy y J.E. Littlewood en [26]. En general en la literatura matemática el ordenamiento es decreciente y positivo, sin embargo, para los fines de este trabajo se debe recurrir a un ordenamiento que preserve los valores extremos de la función, además, que sea creciente ya que este es el caso más utilizado en filtros discretos.

En lo que sigue, sea I un intervalo real acotado, $f \in \mathcal{L}(I)$ con $B := \sup \{f(x) : x \in I\}$ y $A := \inf \{f(x) : x \in I\}$. Recuerde que μ denota la medida de Lebesgue.

Definición 2.2.1. La **función distribución** $T_f : [A, B] \rightarrow [0, \mu(I)]$ de f se define por:

$$T_f(y) := \mu(\{x \in I : f(x) \leq y\}) = \mu(f^{-1}((-\infty, y])).$$

Observación. El valor $T_f(y)$ es la cantidad de tiempo que f pasa por debajo de un nivel y ; esta definición es análoga a $T_{\mathbf{x}}$ en el caso discreto. La distribución brinda información global del tamaño de la función original ignorando el comportamiento puntual.

Sean $y_1, y_2 \in [A, B]$ tales que $y_2 > y_1$, entonces $T_f(y_2) - T_f(y_1)$ es el tiempo que f pasa entre los niveles y_1 y y_2 ; de hecho, esto es la medida del conjunto $f^{-1}((y_1, y_2])$:

$$\begin{aligned} T_f(y_2) &= \mu(\{x \in I : f(x) \leq y_2\}) \\ &= \mu(\{x \in I : f(x) \leq y_1\} \cup \{x \in I : y_1 < f(x) \leq y_2\}) \\ &= T_f(y_1) + \mu(f^{-1}((y_1, y_2])). \end{aligned}$$

En el siguiente teorema se reúnen algunas propiedades de la distribución, su demostración se encuentra en el apéndice B.1.

Teorema 2.2.1 (Propiedades de la distribución).

1. T_f es monótona creciente y continua por derecha.
2. $T_f(y_0^-) = T_f(y_0) - \mu(f^{-1}(\{y_0\}))$.
3. Si f es discontinua en $x_0 \in I$, con $y_2 = f(x_0^+)$ y $y_1 = f(x_0^-)$, y además $f^{-1}((y_1, y_2)) = \emptyset$, entonces T_f es constante en el intervalo (y_1, y_2) con valor $T_f(y_1)$.

Dado que f puede ser localmente constante y/o discontinua, la distribución T_f no es necesariamente inyectiva ni sobreyectiva, y por lo tanto no tiene una inversa en general; no obstante, la siguiente expresión, que se denomina *inversa generalizada*, proporciona el ordenamiento de f .

Definición 2.2.2. El **ordenamiento no decreciente** (ordenamiento monótono) de f es la función $f^*: [0, \mu(I)] \rightarrow \mathbb{R}$ definida por:

$$f^*(\tau) := \inf \{y \in [A, B] : T_f(y) \geq \tau\}.$$

Observe que si T_f es estrictamente creciente, entonces

$$f^*(T_f(u)) = \inf \{y \in [A, B] : T_f(y) \geq T_f(u)\} = u,$$

lo que implica que f^* es su inversa a izquierda. Sin embargo, en general se tiene la desigualdad $f^*(T_f(u)) \leq u$. Por otro lado, $T_f(f^*(\tau)) \geq \tau$, esto se tiene ya que existe una sucesión decreciente $\{\lambda_n\}_{n \in \mathbb{N}}$ que converge a $f^*(\tau)$ tal que $T_f(\lambda_n) \geq \tau$ y así

$$T_f(f^*(\tau)) = \lim_{n \rightarrow \infty} T_f(\lambda_n) \geq \tau,$$

pues T_f es continua por derecha.

Varias propiedades del ordenamiento se presentan en el siguiente teorema, para su demostración consultar el apéndice B.2.

Teorema 2.2.2 (Propiedades del ordenamiento).

1. f^* es monótona creciente.
2. $s < f^*(\tau)$ si y solo si $T_f(s) < \tau$.
3. f^* es continua por izquierda.
4. f y f^* son equimedibles, es decir $T_f = T_{f^*}$.
5. Si T_f es discontinua en y_0 , entonces f^* es constante en el intervalo $(T_f(y_0^-), T_f(y_0))$ con valor y_0 .
6. $T_{T_f}(\tau) = f^*(\tau) - A + \mu(T_f^{-1}(\{\tau\}))$.
7. $f^*(\tau_0^+) - f^*(\tau_0) = \mu(T_f^{-1}(\{\tau_0\}))$.

A continuación, se enseñan algunos ejemplos de la distribución y el ordenamiento.

Ejemplo 2.2.3. Considere la función definida por $f(x) := 4(x - 1/2)^2$, $x \in [0, 1]$; entonces $T_f(y) = \sqrt{y}$ en $y \in [0, 1]$ y $f^*(\tau) = \tau^2$ con $\tau \in [0, 1]$ (figura 2.2).

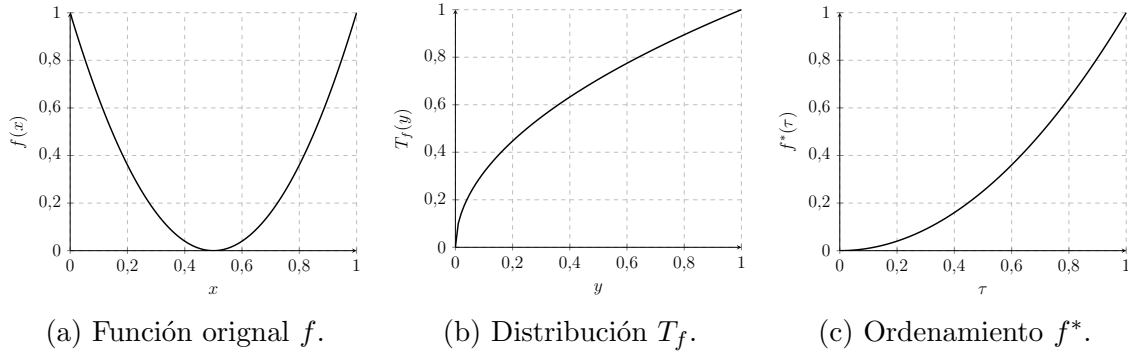


Figura 2.2: Distribución y ordenamiento de $f(x) := 4(x - 1/2)^2$, $x \in [0, 1]$.

Ejemplo 2.2.4. Si $f(x) := \sin x$, $x \in [0, \pi]$, entonces $T_f(y) = 2 \arcsin y$ en $y \in [0, 1]$ y $f^*(\tau) = \sin \tau/2$ para $\tau \in [0, \pi]$ (figura 2.3).

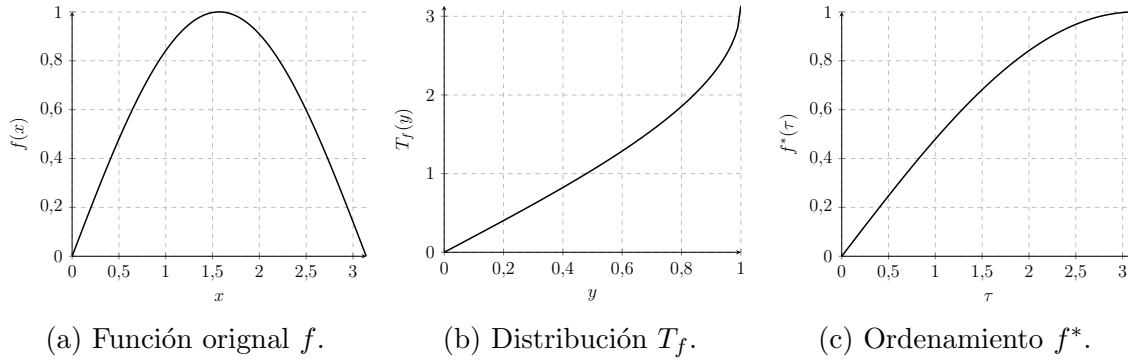


Figura 2.3: Distribución y ordenamiento de $f(x) := \sin x$, $x \in [0, \pi]$.

Ejemplo 2.2.5. Dado $f(x) := \sin x$, $x \in [0, 2\pi]$, entonces $T_f(y) = \pi + 2 \arcsin y$ para $y \in [-1, 1]$; además $f^*(\tau) = -\cos \tau/2$ con $\tau \in [0, 2\pi]$ (figura 2.4).

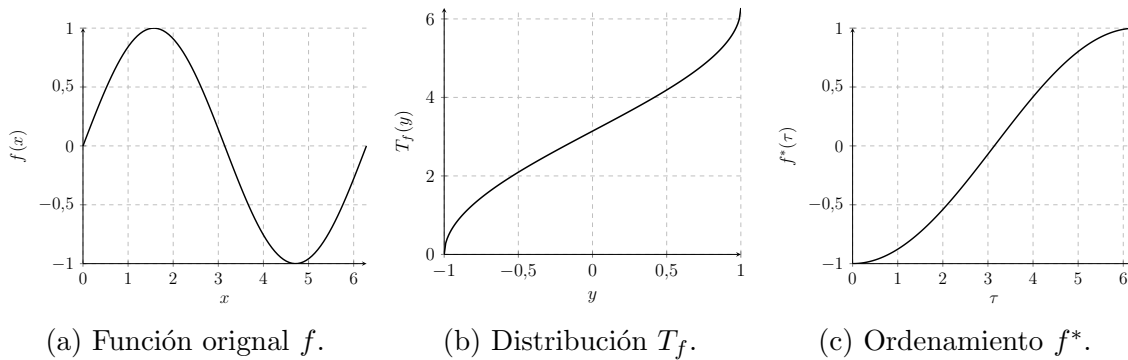


Figura 2.4: Distribución y ordenamiento de $f(x) := \sin x$, $x \in [0, 2\pi]$.

La siguiente es una conexión profunda entre f y f^* : debido a que ambas tienen la misma función distribución conservan la misma integral. La demostración de este teorema se encuentra en el apéndice B.3.

Teorema 2.2.3. *El ordenamiento tiene la misma integral que la función original, esto es:*

$$\int_I f(x) \, dx = \int_0^{\mu(I)} f^*(\tau) \, d\tau$$

Por ejemplo, para la función $f(x) := |\sin x|$, $x \in [0, 2\pi]$, con distribución $T_f(y) = 4 \arcsin y$ para $y \in [0, 1]$ y ordenamiento $f^*(\tau) = \sin \tau/4$ para $\tau \in [0, 2\pi]$ (figura 2.5) se obtiene que $\int_0^{2\pi} |\sin x| \, dx = 4 = \int_0^{2\pi} \sin \tau/4 \, d\tau$.

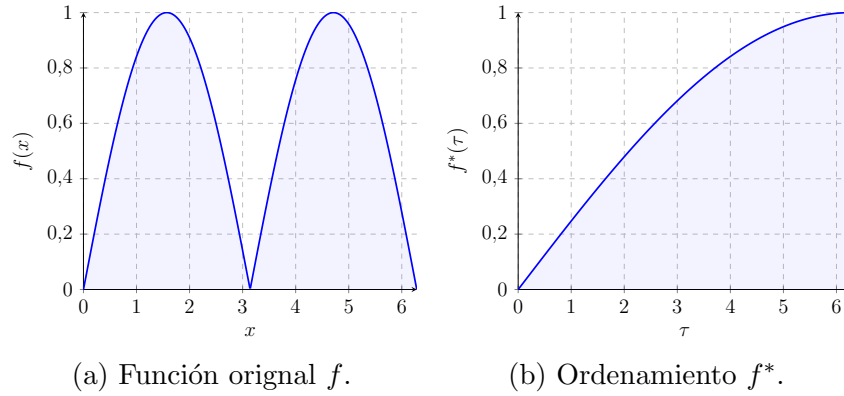


Figura 2.5: La función $f(x) := |\sin x|$, $x \in [0, 2\pi]$ y su ordenamiento.

2.3. Ordenamiento de funciones lineales a trozos

Una función $f: \mathbb{R} \rightarrow \mathbb{R}$ es **lineal a trozos** (LT) si existe un conjunto enumerable de reales, finito o infinito, $\{\dots, \alpha_{-1}, \alpha_0, \alpha_1, \alpha_2, \dots\}$, donde $\alpha_i < \alpha_{i+1}$, sin puntos de acumulación, con intervalos correspondientes $[\alpha_{i-1}, \alpha_i]$, tal que existen $m_i, b_i \in \mathbb{R}$ de forma que

$$f(x) = m_i x + b_i, \quad \forall x \in (\alpha_{i-1}, \alpha_i).$$

Análogamente, si $I := [a, b]$ es un intervalo compacto de \mathbb{R} , $f: I \rightarrow \mathbb{R}$ es lineal a trozos si existe una partición $\mathcal{P} = \{a = \alpha_1 < \dots < \alpha_{n-1} < \alpha_n = b\}$ de I , tal que existen $m_i, b_i \in \mathbb{R}$ de modo que $f(x) = m_i x + b_i$, $\forall x \in (\alpha_{i-1}, \alpha_i)$. Como en principio la partición \mathcal{P} no es única, se dirá que f es **lineal a trozos sobre la partición \mathcal{P}** ; la partición más gruesa posible consiste de los puntos donde f no es diferenciable.

En este trabajo solo será necesario considerar funciones lineales a trozos continuas por derecha o por izquierda en caso de existir discontinuidades, excluyendo así los puntos aislados (ver figura 2.6).

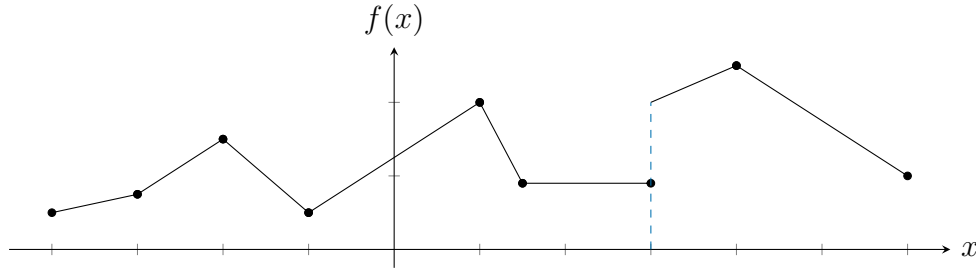


Figura 2.6: Ejemplo de función lineal a trozos.

Una función lineal a trozos de duración finita $f: I \rightarrow \mathbb{R}$ sobre una partición \mathcal{P} puede ser caracterizada por un **conjunto finito de vértices**:

$$\mathcal{V}(f, \mathcal{P}) := \left\{ \left(\alpha, \lim_{x \rightarrow \alpha^\pm} f(x) \right) : \alpha \in \mathcal{P} \right\}.$$

A los elementos del conjunto formado por las segundas componentes de los vértices se les llama niveles verticales o alturas.

Una propiedad importante de las funciones lineales a trozos es que su distribución y ordenamiento también son lineales a trozos: si $[y_{j-1}, y_j]$ es un intervalo dado por dos niveles verticales consecutivos de f , entonces la pendiente de T_f en este intervalo es

$$\sum_i \frac{1}{|m_i|}$$

donde cada m_i es la pendiente de un segmento de la gráfica de la función LT que se encuentra en la franja $\{(x, y) \in I \times \mathbb{R} : y \in [y_{j-1}, y_j]\} \subseteq \mathbb{R}^2$. Aún más, si T_f crece en un intervalo (τ_{j-1}, τ_j) dado por dos niveles verticales consecutivos de T_f , entonces f^* crece linealmente con pendiente

$$\left(\sum_i \frac{1}{|m_i|} \right)^{-1}$$

que es una expresión semejante al paralelo de resistencias. El siguiente teorema, que corresponde al primer objetivo del trabajo, muestra que T_f y f^* son LT tomando como particiones los niveles verticales.

Teorema 2.3.1 (Ordenamiento de funciones LT). *Sea $f: I \rightarrow \mathbb{R}$ una función lineal a trozos sobre una partición $\mathcal{P} = \{\alpha_1, \dots, \alpha_n\}$; entonces T_f y f^* son lineales a trozos.*

Demostración. Denote a los intervalos $I_k^\circ := (\alpha_{k-1}, \alpha_k)$ y las segundas componentes (niveles o alturas) de los vértices de f por:

$$y_i^\pm := \lim_{x \rightarrow \alpha_i^\pm} f(x), \quad \forall \alpha_i \in \mathcal{P}.$$

El conjunto de todos los niveles, $\mathcal{P}' := \{y_i^\pm\}$, es una partición de $[A, B]$. Se utilizará el subíndice j en los elementos de \mathcal{P}' para indicar que no hay repetición de niveles, i.e. $y_{j-1} < y_j$. Se mostrará que T_f es lineal en $[y_{j-1}, y_j]$.

Sea $y \in [y_{j-1}, y_j)$ y defina $H_k(y) := \{x \in I_k^\circ : f(x) \leq y\}$. Observe que algunos $H_k(y)$ pueden ser vacíos para el intervalo escogido $[y_{j-1}, y_j)$, con esto en mente defina

$$\mathcal{M}_j = \{k \in \{2, \dots, n\} : \{x \in I_k^\circ : f(x) < y_j\} \neq \emptyset\}.$$

Con la notación anterior, note que

$$\begin{aligned} T_f(y) &= \mu(\{x \in I : f(x) \leq y\}) = \mu\left(\bigcup_{k=1}^n H_k(y) \cup \{\alpha \in \mathcal{P} : f(\alpha) \leq y\}\right) \\ &= \sum_{k \in \mathcal{M}_j} \mu(H_k(y)). \end{aligned}$$

Observe que si $\sup f(I_k^\circ) \leq y_{j-1}$ (aquí $\sup f(I_k^\circ)$ es el supremo de la imagen del intervalo I_k°), entonces $f(x) \leq y_{j-1} \leq y$ para todo $x \in I_k^\circ$; con lo cual $\mu(H_k(y)) = \alpha_k - \alpha_{k-1}$; por lo tanto definiendo $\mathcal{N}_j := \{k \in \{2, \dots, n\} : \sup f(I_k^\circ) \leq y_{j-1}\}$, se sigue que

$$\sum_{k \in \mathcal{N}_j} \mu(H_k(y)) = \sum_{k \in \mathcal{N}_j} \alpha_k - \alpha_{k-1}.$$

Por otro lado, dado que $f(x) = c_k x + b_k$ para todo $x \in I_k^\circ$ y tomando $\mathcal{D}_j := \mathcal{M}_j - \mathcal{N}_j$, entonces

$$\begin{aligned} \sum_{k \in \mathcal{D}_j} \mu(H_k(y)) &= \sum_{\substack{k \in \mathcal{D}_j \\ c_k > 0}} \mu\left(\left\{x \in I_k^\circ : x \leq \frac{y - b_k}{c_k}\right\}\right) + \sum_{\substack{k \in \mathcal{D}_j \\ c_k < 0}} \mu\left(\left\{x \in I_k^\circ : x \geq \frac{y - b_k}{c_k}\right\}\right) \\ &= \sum_{\substack{k \in \mathcal{D}_j \\ c_k > 0}} \frac{y - b_k}{c_k} - \alpha_{k-1} + \sum_{\substack{k \in \mathcal{D}_j \\ c_k < 0}} \alpha_k + \frac{y - b_k}{|c_k|} \\ &= y \sum_{k \in \mathcal{D}_j} \frac{1}{|c_k|} - \sum_{\substack{k \in \mathcal{D}_j \\ c_k > 0}} \frac{b_k}{c_k} + \alpha_{k-1} + \sum_{\substack{k \in \mathcal{D}_j \\ c_k < 0}} \alpha_k - \frac{b_k}{|c_k|}. \end{aligned}$$

Por lo tanto, la forma explícita para T_f en el intervalo $[y_{j-1}, y_j)$ es:

$$\begin{aligned} T_f(y) &= \sum_{k \in \mathcal{M}_j} \mu(H_k(y)) = \sum_{k \in \mathcal{D}_j} \mu(H_k(y)) + \sum_{k \in \mathcal{N}_j} \mu(H_k(y)) \\ &= \underbrace{y \sum_{k \in \mathcal{D}_j} \frac{1}{|c_k|}}_{m_j} - \underbrace{\sum_{\substack{k \in \mathcal{D}_j \\ c_k > 0}} \frac{b_k}{c_k} + \alpha_{k-1} + \sum_{\substack{k \in \mathcal{D}_j \\ c_k < 0}} \alpha_k - \frac{b_k}{|c_k|} + \sum_{k \in \mathcal{N}_j} \alpha_k - \alpha_{k-1}}_{b_j}, \end{aligned}$$

y así T_f es lineal a trozos sobre la partición \mathcal{P}' . Para el ordenamiento considere el conjunto de vértices de T_f que tiene la forma

$$\mathcal{V}(T_f, \mathcal{P}') = \left\{ \left(y_j, \lim_{y \rightarrow y_j^\pm} T_f(y) \right) : y_j \in \mathcal{P}' \right\},$$

y de nuevo los niveles verticales

$$\beta_j^\pm := \lim_{y \rightarrow y_j^\pm} T_f(y).$$

Con base en los vértices de T_f considere el conjunto de puntos $\mathcal{P}^* = \{\beta_j^\pm\}$, que es una partición de $[0, \mu(I)]$. De nuevo, se utilizará el subíndice l para indicar que no hay repeticiones, i.e. $\beta_{l-1} < \beta_l$. Se mostrará que f^* es lineal en (β_{l-1}, β_l) .

Si la imagen inversa $T_f^{-1}((\beta_{l-1}, \beta_l)) \neq \emptyset$ quiere decir que (β_{l-1}, β_l) corresponde a una recta creciente de T_f , entonces para cada $\tau \in (\beta_{l-1}, \beta_l)$ existe un único $y_0 \in [A, B]$ tal que $T_f(y_0) = \tau$, i.e., $\tau = y_0 m_{l-1} + b_{l-1}$ y así:

$$f^*(\tau) = f^*(T_f(y_0)) = y_0 = \frac{\tau - b_{l-1}}{m_{l-1}}.$$

Si la imagen inversa $T_f^{-1}((\beta_{l-1}, \beta_l)) = \emptyset$, por el ítem (5) del teorema 2.2.2 f^* es constante en (β_{l-1}, β_l) . En cualquier caso f^* es un polinomio de grado 1 y por lo tanto es lineal a trozos sobre la partición \mathcal{P}^* con el siguiente conjunto de vértices:

$$\mathcal{V}(f^*, \mathcal{P}^*) = \left\{ \left(\beta_l, \lim_{\tau \rightarrow \beta_l^\pm} f^*(\tau) \right) : \beta_l \in \mathcal{P}^* \right\}.$$

□

Tanto la distribución como el ordenamiento de funciones lineales a trozos están descritos por sus niveles verticales y pendientes. Por ejemplo, para la señal con tres vértices y dos niveles verticales de la figura 2.7, cuyas rectas tienen pendientes $50/7$ y $-50/3$, el ordenamiento corresponde a una sola recta cuya pendiente está dada por $(7/50 + 3/50)^{-1} = 5$.

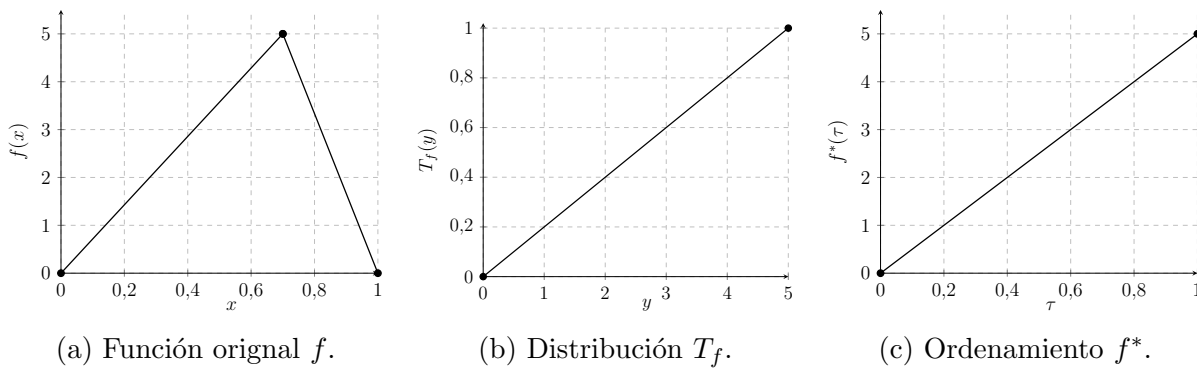


Figura 2.7: Distribución y ordenamiento de una función lineal a trozos.

La figura 2.8 muestra un ejemplo con segmentos constantes y discontinuidades. Note que f es constante con valor $f(x) = 3$ para $x \in (8, 10)$, y por consecuencia T_f es discontinua en $y = 3$ con un salto de magnitud 2; también $f^{-1}((2, 3)) = \emptyset$, con lo cual $T_f(y) = 2$ para todo $y \in [2, 3)$ y f^* tiene un salto en $\tau = 2$ de 2 a 3.

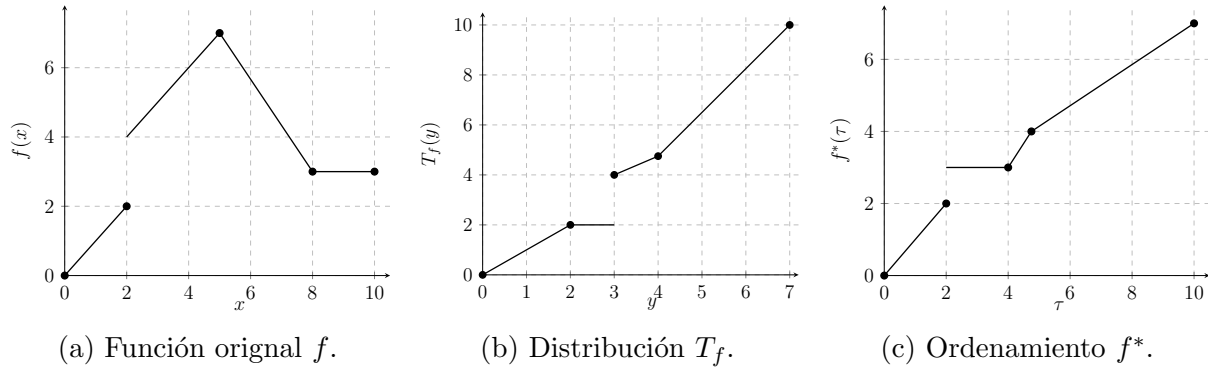


Figura 2.8: Distribución y ordenamiento de función lineal a trozos con discontinuidad y segmento constante.

El teorema 2.3.1 proporciona una expresión determinada para los vértices del ordenamiento de una función lineal a trozos; junto con las propiedades de T_f y f^* , hacen posible un algoritmo para calcular dichos vértices como se describirá en el capítulo 4.

Parte II

Filtros de orden analógicos

Capítulo 3

Filtros de orden analógicos vía funcionales

Este capítulo ofrece una descripción general de los filtros de orden analógicos. En la sección 3.1 se presentan los filtros de estadísticas de orden con los operadores (discretos) de ventana y orden; la sección 3.2 extiende estas nociones discretas al dominio analógico, definiendo un operador de ventana para parámetro continuo, un operador de orden basado en el ordenamiento monótono visto en el capítulo 2 y por último un funcional de Stieltjes. La sección 3.3 explora diferentes funcionales que resultan en los filtros de ranqueo, el promedio móvil, entre otros.

El comportamiento de los filtros de ranqueo a señales LT es de gran importancia ya que facilita su implementación algorítmica, tema que se tratará con detalle en el capítulo 4; sin embargo, en la sección 3.4 se demuestra que si la entrada de estos filtros es LT entonces su salida también es LT (teorema 3.4.1).

3.1. Filtros de estadísticas de orden

Los filtros de estadísticas de orden (filtros L) fueron introducidos por A. Bovik et al. en [5] como una generalización del filtro mediana discreto. Son filtros no lineales de ventana móvil utilizados principalmente para suavizar (o alisar) señales discretas y también para detectar bordes (contornos) en imágenes digitales [6].

La idea principal consiste en tomar segmentos de la señal de entrada por medio de ventanas, ordenar las muestras en los segmentos y por último tomar una combinación lineal de las muestras ordenadas (figura 3.1). Su no linealidad se debe al proceso de ordenamiento que se realiza a los segmentos ventaneados.



Figura 3.1: Los tres bloques principales de un filtro L: toma de segmentos de la señal de entrada, ordenamiento de las muestras y combinación lineal aplicada al ordenamiento.

Recuerde que dado un vector $\mathbf{x} = [x_1, x_2, \dots, x_k] \in \mathbb{R}^k$, denotamos por $[x_{(1)}, x_{(2)}, \dots, x_{(k)}]$ al vector de estadísticas de orden (versión ordenada) de \mathbf{x} . Podemos definir un operador de orden $\mathbf{o}: \mathbb{R}^k \rightarrow \mathbb{R}^k$ por

$$\mathbf{o}([x_1, x_2, \dots, x_k]) := [x_{(1)}, x_{(2)}, \dots, x_{(k)}].$$

Sean $x: \mathbb{Z} \rightarrow \mathbb{R}$ una señal discreta con valores $x_n := x(n)$ y $\mathbf{a} = [a_1, a_2, \dots, a_k] \in \mathbb{R}^k$ los pesos del filtro (factores de ponderación). También, conocido el valor de k , defina el operador de ventana \mathbf{w}_n que actúa sobre señales discretas (ver figura 3.2) y está dado por

$$\mathbf{w}_n(x) := [x_n, x_{n-1}, \dots, x_{n-k+1}] \in \mathbb{R}^k.$$

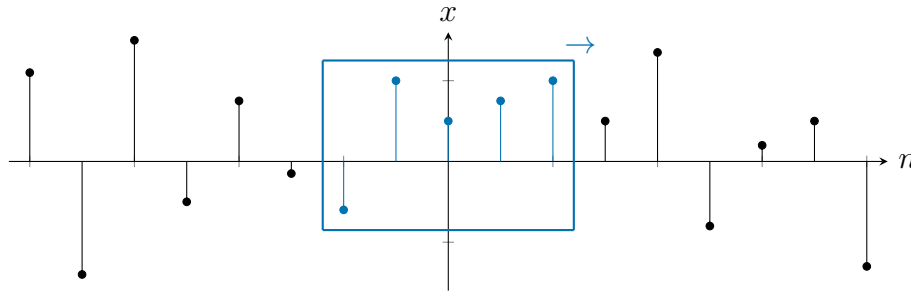


Figura 3.2: Ejemplo de operador \mathbf{w}_n : ventana móvil de 5 muestras en una señal discreta x .

Para cada $n \in \mathbb{Z}$, la n -ésima salida y_n del filtro corresponde al producto punto del vector ordenado $\mathbf{o}(\mathbf{w}_n(x))$ con el vector de pesos del filtro:

$$y_n := \mathbf{a}^T \mathbf{o}(\mathbf{w}_n(x)) = \sum_{i=1}^k a_i x_{(i)}^n,$$

donde $x_{(i)}^n$ denota la i -ésima estadística de orden del vector ventaneado $\mathbf{w}_n(x)$. Cuando el vector de pesos del filtro tiene la forma $\mathbf{a} = [0, \dots, 0, 1, 0, \dots, 0]$ con un 1 en la i -ésima posición, obtenemos los filtros de orden ranqueado (OR); en particular si k es impar y toma la $k+1/2$ posición, se tiene el filtro mediana discreto.

3.2. Generalización a los filtros de orden analógicos

Ahora se definen los filtros de orden analógicos; para ello, se van a generalizar los operadores discretos de ventana y orden para señales de parámetro (tiempo) continuo. Se hará uso del ordenamiento monótono y el funcional de Stieltjes φ con semilla g .

En lo que sigue, sean $I \subseteq \mathbb{R}$ un intervalo y $L \in \mathbb{R}$ el ancho de la ventana; dado $t \in I$ se denotará $I_t := [t - L/2, t + L/2]$. Note que en general $I_t \not\subseteq I$. Todos los operadores actuarán en \mathcal{L} , el espacio vectorial de las señales integrables y acotadas.

El **operador de ventana** $W_{t,L}: \mathcal{L}(I) \rightarrow \mathcal{L}(I_t)$ centrado en t y de ancho L , toma una señal $f: I \rightarrow \mathbb{R}$ y asigna una señal ventaneada $f_t: I_t \rightarrow \mathbb{R}$ que corresponde a restringir f como sigue:

$$f_t(s) = \begin{cases} f(s), & \text{si } s \in I; \\ 0 & \text{en otro caso.} \end{cases}$$

Observe que el operador asigna cero para puntos s que no se encuentren en el intervalo I . Por otro lado, una forma alternativa (causal) para la ventana es utilizar $I_t := [t - L, t]$. La figura 3.3 ilustra la acción del operador $W_{t,L}$ sobre una señal f .

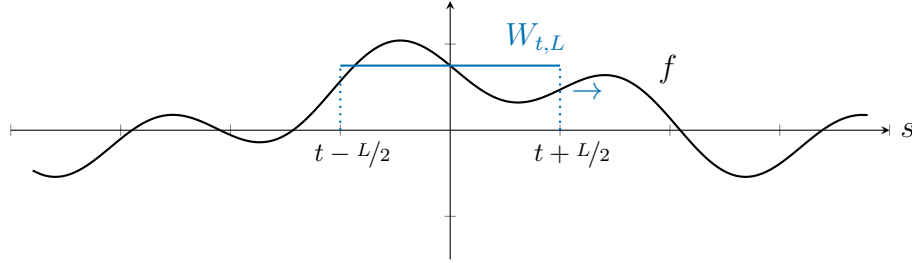


Figura 3.3: Ejemplo de operador $W_{t,L}$: ventana móvil sobre una señal f .

El **operador de orden** asigna a cada señal ventaneada su ordenamiento monótono, es decir $\mathbf{o}: \mathcal{L}(I_t) \rightarrow \mathcal{L}([0, L])$ donde $\mathbf{o}(f_t) := f_t^*: [0, L] \rightarrow \mathbb{R}$. Ahora, sea f una señal de entrada y φ el funcional de Stieltjes con semilla g . **La salida de un filtro de orden analógico** de ancho de ventana L es la señal $\tilde{f}: I \rightarrow \mathbb{R}$ dada por la composición:

$$\tilde{f}(t) = \varphi(\mathbf{o}(W_{t,L}(f))) = \int_0^L f_t^* dg, \quad \forall t \in I.$$

En resumen, el operador de ventana segmenta la señal de entrada para cada tiempo, luego ordena este segmento y aplica el funcional (ver figura 3.4).

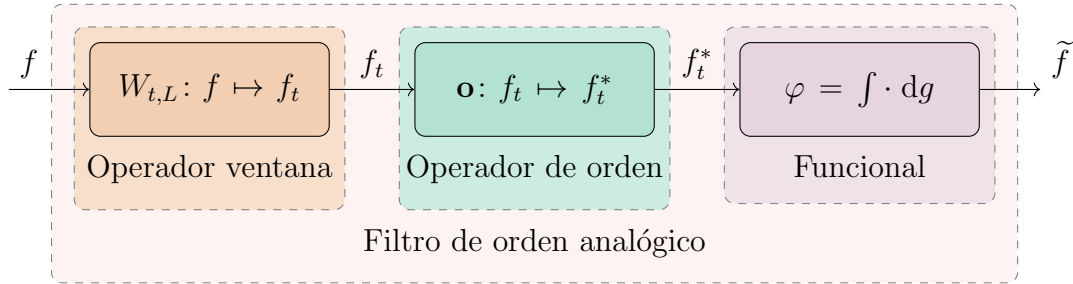


Figura 3.4: Representación en bloques de los filtros de orden analógicos.

3.3. Ranqueo, promedio móvil y otros funcionales

Lo que determina el tipo de filtro de orden es el funcional de Stieltjes φ . En esta sección se exponen algunos ejemplos de funcionales, tenga presente la sección 1.3.

Si para la semilla g del funcional se toma la función de Heaviside o escalón unitario con salto en algún $\tau_0 \in [0, L]$, i.e. $u(\tau - \tau_0)$, entonces se tiene:

$$\tilde{f}(t) = \int_0^L f_t^*(\tau) du(\tau - \tau_0) = f_t^*(\tau_0), \quad \forall t \in I.$$

Se puede ver entonces que este funcional toma la muestra del ordenamiento en τ_0 (figura 3.5); estos se conocen también como *filtros de orden-ranqueado analógicos* (o simplemente filtros de ranqueo) por su similitud con los OR discretos. Por ejemplo, si $\tau_0 = L/2$ se obtiene el filtro mediana analógico, ya que se está capturando la muestra del ordenamiento en la mitad del dominio.

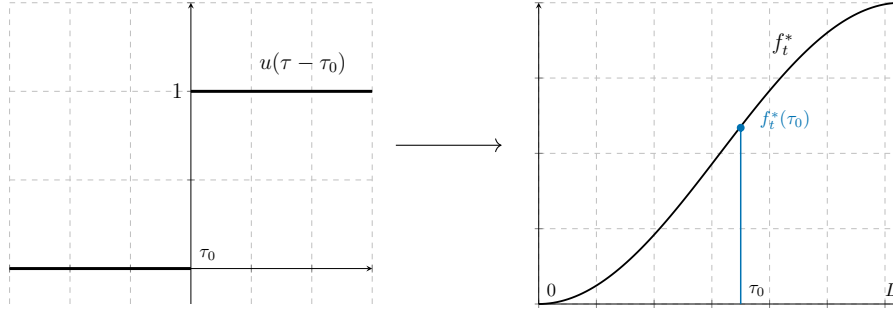


Figura 3.5: Semilla $g(\tau) = u(\tau - \tau_0)$ y efecto en el ordenamiento f_t^* .

Si se quiere una combinación lineal de ciertas muestras del ordenamiento, semejante a los filtros L, se utiliza una semilla g constante a trozos (figura 3.6) con discontinuidades en τ_1, \dots, τ_m y saltos $\alpha_1, \dots, \alpha_m$, en efecto:

$$\tilde{f}(t) = \int_0^L f_t^*(\tau) dg(\tau) = \sum_{i=1}^m \alpha_i f_t^*(\tau_i), \quad \forall t \in I.$$

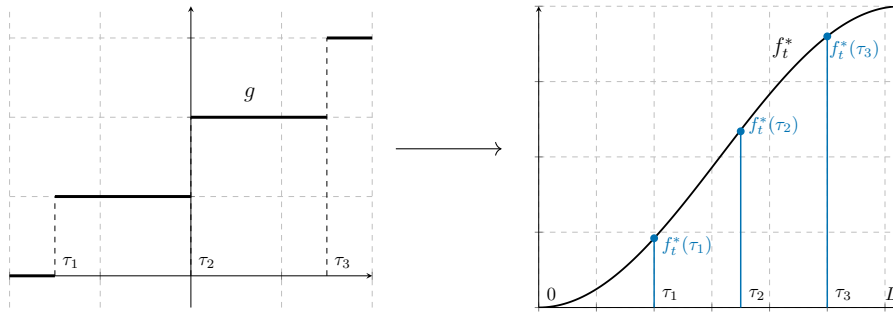


Figura 3.6: Ejemplo de semilla g constante a trozos con saltos $\alpha_1 = \alpha_2 = \alpha_3 = 1$ y efecto en el ordenamiento f_t^* .

Ahora, si g es una recta con pendiente $1/L$ y utilizando el hecho de que el ordenamiento tiene la misma integral que la señal original, se sigue que:

$$\begin{aligned} \tilde{f}(t) &= \int_0^L f_t^*(\tau) dg(\tau) = \int_0^L f_t^*(\tau) \cdot g'(\tau) d\tau \\ &= \frac{1}{L} \int_0^L f_t^*(\tau) d\tau = \frac{1}{L} \int_{t-L/2}^{t+L/2} f_t(\tau) d\tau, \quad \forall t \in I, \end{aligned}$$

que es precisamente el filtro promedio móvil. Por otra parte tomar a g diferenciable con continuidad da lugar a toda una clase de filtros de orden analógicos.

3.4. Filtrado de ranqueo de señales lineales a trozos

Idealmente en los filtros de ranqueo, y en general en los de orden analógicos, el operador de ventana $W_{t,L}$ recorre todo el intervalo I , por lo que para cada $t \in I$ corresponde una señal ventaneada f_t . En la práctica no se puede realizar el corrimiento continuo de la ventana en cada punto del intervalo para obtener la salida del filtro, ahora bien, si se restringe la entrada de los filtros a señales lineales a trozos y se muestra que la señal de salida también es lineal a trozos, **solamente deberán ser calculados finitos vértices de salida**. Con base en lo anterior, en esta sección la señal de entrada $f: I \rightarrow \mathbb{R}$ se asume lineal a trozos.

Notación: si f_t es una señal ventaneada en $t \in I$, entonces $\{(x_1^t, y_1^t), \dots, (x_d^t, y_d^t)\}$ denota su conjunto de vértices; en particular (x_l^t, y_l^t) y m_l denotarán el vértice en el extremo izquierdo de la ventana y la pendiente de ese extremo, análogamente (x_r^t, y_r^t) y m_r para el extremo derecho. Defina $A^t := \inf f_t$ y $B^t := \sup f_t$. Se asumirá que cada y_i es un punto de cambio de pendiente, i.e. la partición más gruesa.

Por el teorema 2.3.1 el ordenamiento f_t^* es lineal a trozos. Las alturas de los vértices del ordenamiento vienen dadas por las alturas de los vértices de la señal f_t ; estas se mantienen constantes con t mientras se encuentren en la ventana y las alturas de los extremos izquierdo y derecho, i.e. y_l^t, y_r^t , varían siempre que no estén en segmentos de pendiente nula.

Para determinar los vértices de la señal de salida \tilde{f} se analizará lo que ocurre entre señales ventaneadas llamadas *hitos*, estas serán suficientes en el sentido de que se puede predecir lo que sucederá entre cada una.

Definición 3.4.1. La señal ventaneada f_t es un **hito** si alguna de las siguientes se satisface:

- $y_l^t = y_r^t$ y $m_l \neq m_r$;
- para algún i , $y_l^t = y_i^t$ donde y_i^t es la altura de un vértice interno;
- para algún i , $y_r^t = y_i^t$ donde y_i^t es la altura de un vértice interno;
- si (x_l^t, y_l^t) o (x_r^t, y_r^t) son iguales a un vértice de la señal de entrada f .

Adicionalmente, si $y_l^t = y_r^t$ y $m_r = m_l$ entonces el tiempo inicial y final en los cuales se cumplen las condiciones definen dos hitos.

Diferentes hitos no necesariamente implican nuevos vértices en la señal de salida \tilde{f} , pero sí **implican una nueva pendiente en el ordenamiento**, es por esto que se estudiará lo que ocurre entre hitos consecutivos, ya que podemos garantizar que el número de pendientes de los ordenamientos (y el valor de las mismas) no cambiará; esto implica que entre hitos consecutivos podemos encontrar un intervalo en el cual el valor $f_t^*(\tau_0)$ varía de forma lineal de acuerdo a la posición de las alturas y_l^t y y_r^t .

Teorema 3.4.1. Sean $f_{t'}$ y $f_{t''}$ dos hitos consecutivos y $\tau_0 \in [0, L]$ fijo; entonces existe un intervalo en el cual $f_t^*(\tau_0)$ varía de forma lineal y en consecuencia \tilde{f} es lineal a trozos.

Demostración. Sea $t \in (t', t'')$ y suponga en primer lugar que $y_r^t < y_l^t$, en todo el intervalo (t', t'') se puede asegurar que esta desigualdad se cumple. Ahora, dado que f_t^* es lineal a trozos, podemos tomar m^* como la pendiente de la recta en la que se encuentra el valor

$f_t^*(\tau_0)$, supongamos $m^* \neq 0$, en caso contrario se mantendrá constante hasta que cambie de pendiente. Bajo la suposición anterior existe $y_0 \in [A^t, B^t]$ tal que $T_{f_t}(y_0) = \tau_0$ y y_0 es la menor altura que cumple esto. Con base en la posición de y_0 respecto a los extremos y_l^t y y_r^t se tendrán los siguientes casos:

- Si $y_r^t < y_0 < y_l^t$ tome el máximo Δt tal que

$$y_r^\vartheta < y_0 < y_l^\vartheta, \quad \forall \vartheta: t \leq \vartheta \leq t + \Delta t,$$

tal que y_0 no cambia de recta y tal que $t + \Delta t$ sea menor al siguiente hito, concretamente, Δt es el movimiento de la ventana de manera que y_0 se mantiene entre y_l^t y y_r^t y además no cambia de pendiente. Bajo estos supuestos se tiene que

$$T_{f_{t+\Delta t}}(y_0) = T_{f_t}(y_0) + \Delta t$$

por el movimiento de la ventana y la desigualdad de los extremos, es decir, como $y_r^t < y_l^t$, si se mueve la ventana Δt y la desigualdad $y_r^t < y_0 < y_l^t$ se mantiene, entonces el tiempo por debajo de y_0 aumenta precisamente Δt . Si se considera $y_\gamma := y_0 - m^* \Delta t$, i.e. si bajamos de altura siguiendo la recta, entonces

$$\begin{aligned} T_{f_{t+\Delta t}}(y_\gamma) &= T_{f_{t+\Delta t}}(y_0) - \mu(f_{t+\Delta t}^{-1}((y_\gamma, y_0])) \\ &= T_{f_t}(y_0) + \Delta t - \mu(f_{t+\Delta t}^{-1}((y_0 - m^* \Delta t, y_0])) \\ &= T_{f_t}(y_0) + \Delta t - \mu((f_{t+\Delta t}^*)^{-1}((y_0 - m^* \Delta t, y_0])) \\ &= T_{f_t}(y_0) + \Delta t - \Delta t \\ &= T_{f_t}(y_0). \end{aligned}$$

En conclusión tenemos que $T_{f_t}(y_0) = T_{f_{t+\Delta t}}(y_\gamma) = \tau_0$, aplicando f_t^* y $f_{t+\Delta t}^*$ se obtiene:

$$\begin{aligned} f_{t+\Delta t}^*(\tau_0) &= f_{t+\Delta t}^*(T_{f_{t+\Delta t}}(y_\gamma)) = y_\gamma \\ &= y_0 - m^* \Delta t \\ &= f_t^*(T_{f_t}(y_0)) - m^* \Delta t = f_t^*(\tau_0) - m^* \Delta t, \end{aligned}$$

es decir $f_{t+\Delta t}^*(\tau_0) = f_t^*(\tau_0) - m^* \Delta t$.

- Si y_0 se encuentra por fuera del intervalo (y_r^t, y_l^t) , entonces tome el máximo Δt tal que

$$y_0 \notin (y_r^\xi, y_l^\xi), \quad \forall \xi: t \leq \xi \leq t + \Delta t,$$

tal que y_0 de nuevo no cambie de recta y que $t + \Delta t$ sea menor al siguiente hito, es decir se mueve la ventana de forma que y_0 siga estando por fuera del intervalo y en la misma recta. Esto implica que $T_{f_{t+\Delta t}}(y_0) = T_{f_t}(y_0) = \tau_0$, es decir el tiempo por debajo de y_0 no cambió, y por lo tanto $f_{t+\Delta t}^*(\tau_0) = f_t^*(\tau_0)$.

Análogamente si $y_l^t < y_r^t$ y se cumple que $y_l^t < y_0 < y_r^t$, en este caso con condiciones similares el valor aumenta $f_{t+\Delta t}^*(\tau_0) = f_t^*(\tau_0) + m^* \Delta t$. Si $y_0 \notin (y_r^{t'}, y_l^{t'})$ igualmente se mantiene constante $f_{t+\Delta t}^*(\tau_0) = f_t^*(\tau_0)$. Si en cualquier caso durante el recorrido de la ventana se produce un cambio de pendiente, eso implica un nuevo vértice a la salida del filtro, se entra a un nuevo hito y utilizando los mismos argumentos se puede extender a todo el dominio.

En conclusión, el movimiento de la ventana produce un cambio de tipo lineal para las alturas $f_t^*(\tau_0)$ del ordenamiento, con lo cual \tilde{f} será lineal a trozos. \square

Poder determinar si $f_t^*(\tau_0)$ sube o baja con el movimiento de la ventana conlleva a que se puedan calcular los vértices de \tilde{f} dentro de hitos consecutivos; por ejemplo, en el caso $y_r^t < y_0 < y_l^t$ se sabe que $f_t^*(\tau_0)$ disminuye, suponga que la pendiente de la frontera derecha de la ventana es positiva, i.e. $m_r > 0$ y además no hay más niveles verticales entre y_0 y y_r^t . Si la recta de la frontera puede llegar a la altura $f_t^*(\tau_0)$ antes del siguiente hito, entonces el tiempo en el que llegue a esta altura, junto con $f_t^*(\tau_0)$, es un potencial vértice. Podría ocurrir también que la recta de la frontera izquierda sea negativa e inmediatamente superior a y_0 , y el potencial vértice se calcula de forma similar. De todos los posibles vértices de salida que se puedan obtener con los vértices internos y en la frontera de la ventana, el de la salida será el de menor tiempo.

El teorema anterior implica también que los filtros de combinación lineal son lineales a trozos, pues tomando el intervalo más grande en el cual cada ranqueo es lineal se obtiene:

$$\tilde{f}(t) = \sum_{i=1}^n \alpha_i f_t^*(\tau_i) = \sum_{i=1}^n \alpha_i [m_i^* t + b_i^*] = t \sum_{i=1}^n \alpha_i m_i^* + \sum_{i=1}^n \alpha_i b_i^*.$$

Capítulo 4

Algoritmo para filtros de ranqueo y aplicaciones

La simplificación del tipo de señal analógica de entrada a señales lineales a trozos supone que el cálculo de sus versiones filtradas, por filtros de ranqueo, es factible por medios digitales; gracias a los teoremas 2.3.1 y 3.4.1, el hecho de que tanto el ordenamiento como la salida de los filtros sea lineal a trozos implica que solo deben ser calculados finitos vértices de estas señales.

En este capítulo se presentan algoritmos que permiten implementar los filtros de ranqueo para señales lineales a trozos; en la sección 4.1 se describe el algoritmo de ordenamiento, que a un conjunto de vértices de entrada que determinan una señal LT f , asigna un conjunto de vértices ordenados que corresponden a f^* ; para la verificación del algoritmo se realiza un protocolo de pruebas y se muestran resultados de la implementación.

En la sección 4.2 se presenta un algoritmo para los filtros de ranqueo por medio de una ventana móvil de corrimiento $\epsilon > 0$; para ello se diseñan los métodos del operador de ventana $W_{t,L}: f \mapsto f_t$ y el funcional de ranqueo $\varphi: f_t^* \rightarrow f_t^*(\tau_0)$. La sección 4.3 detalla el algoritmo para los filtros de ranqueo por el cálculo de finitos vértices, proporcionando un protocolo de pruebas para la verificación y ejemplos de la implementación. Por último, la sección 4.4 analiza los resultados obtenidos y la sección 4.5 ofrece diferentes aplicaciones.

Todas las implementaciones se llevaron a cabo en una máquina con sistema operativo WINDOWS, un procesador Intel Core i5-7200U, 4GB de memoria RAM, sobre Python y con la herramienta GOOGLE COLAB; se pueden encontrar en el repositorio [27].

4.1. Algoritmo de ordenamiento

Dada una señal lineal a trozos $f: I \rightarrow \mathbb{R}$, con vértices $\{(x_1, y_1), \dots, (x_n, y_n)\}$, el teorema de ordenamiento 2.3.1 concede una expresión para los vértices de f^* :

$$\mathcal{V}(f^*, \mathcal{P}^*) = \left\{ \left(\tau_l, \lim_{\tau \rightarrow \tau_l^\pm} f^*(\tau) \right) : \tau_l \in \mathcal{P}^* \right\},$$

donde cada τ_l es de la forma $T_f(y_j^+) = T_f(y_j)$ o $T_f(y_j^-)$, para algún nivel vertical y_j de f , i.e. elementos del conjunto $\{y_1, \dots, y_n\}$, lo que sugiere realizar un análisis por casos.

Note que si T_f es continua en y_j , i.e. $\mu(f^{-1}(\{y_j\})) = 0$, entonces los límites coinciden y se tiene un solo tiempo $\tau_j = T_f(y_j)$ y así $f^*(T_f(y_j)) = y_j$ por ser nivel vertical. Si por el contrario la señal f es constante en el nivel y_j , i.e. $\mu(f^{-1}(\{y_j\})) \neq 0$, entonces T_f es discontinua en y_j y se obtienen dos tiempos: $T_f(y_j)$ y $T_f(y_j^-) = T_f(y_j) - \mu(f^{-1}(\{y_j\}))$. La discontinuidad de T_f implica que f^* será constante en $(T_f(y_j^-), T_f(y_j)]$ con valor y_j . Por último, si f es discontinua en $x_0 \in I$ con $f^{-1}((y_j, y_{j+1})) = \emptyset$, entonces $T_f(y) = T_f(y_j)$ para todo $y \in (y_j, y_{j+1})$ y así $\mu(T_f^{-1}(\{T_f(y_j)\})) = y_{j+1} - y_j$ y se tiene:

- $f^*(T_f(y_j)) = y_j$ por ser nivel vertical;
- $f^*(T_f(y_j)^+) = f^*(T_f(y_j)) + \mu(T_f^{-1}(\{T_f(y_j)\})) = y_{j+1}$.

Del análisis anterior se obtienen las siguientes conclusiones: los niveles verticales de los vértices de f^* son los mismos de f y por lo tanto no hay que calcularlos; si hay segmentos constantes se agrega dos veces el mismo nivel; los tiempos de los vértices están dados por $T_f(y_j)$ y $T_f(y_j) - \mu(f^{-1}(\{y_j\}))$ en caso de segmentos constantes.

Para calcular los tiempos $T_f(y_j)$ se pueden utilizar los niveles por debajo de y_j :

$$T_f(y_j) = \mu(f^{-1}(\{y_1\})) + \sum_{k=2}^j \mu(f^{-1}((y_{k-1}, y_k])).$$

Es posible implementar una rutina que recorra recta por recta de f y calcule el tiempo que estas pasan en la franja dada por $(y_{k-1}, y_k]$, y así obtener el valor de $\mu(f^{-1}((y_{k-1}, y_k]))$. El algoritmo de ordenamiento calcula de esta manera los tiempos de los vértices y asigna el correspondiente nivel vertical, para el pseudocódigo ir a [A.1](#)

4.1.1. Protocolo de pruebas y resultados

El algoritmo de ordenamiento a partir de los vértices de entrada de una señal f calcula los vértices de f^* , $\{(x_1, y_1), \dots, (x_n, y_n)\} \mapsto \{(x_1^*, y_1^*), \dots, (x_m^*, y_m^*)\}$, con $m \leq n$ en general. Para verificar que estos vértices representan al ordenamiento se utilizará el hecho de que f y f^* tienen los mismos niveles verticales y que son equimedibles, i.e. $T_f = T_{f^*}$, que implica

$$\mu(f^{-1}((y_{k-1}, y_k])) = \mu((f^*)^{-1}((y_{k-1}, y_k]))$$

para todo par de niveles consecutivos y_{k-1}, y_k .

Para realizar la verificación se debe calcular $\mu(f^{-1}((y_{k-1}, y_k]))$ y $\mu((f^*)^{-1}((y_{k-1}, y_k]))$ para cada par de niveles consecutivos y **tomar la resta de estos dos valores**; luego de obtener todas la diferencias se toma el promedio, máximo y desviación estándar, que se denotarán por $\text{avg } \mu(f, f^*)$, $\text{máx } \mu(f, f^*)$ y $\text{std } \mu(f, f^*)$ respectivamente.

Las pruebas se realizan sobre señales aleatorias con vértices cuyas componentes están distribuidas uniformemente como $[x_1, \dots, x_n] \sim \text{Unif}(-n, n)$ y $[y_1, \dots, y_n] \sim \text{Unif}(-2n, 2n)$.

En la figura [4.1](#) se encuentra un ejemplo de una señal lineal a trozos con 39 vértices para la cual se obtuvo un tiempo de ejecución de 0.00288 segundos, $\text{máx } \mu(f, f^*) = 1,24344 \cdot 10^{-14}$, $\text{avg } \mu(f, f^*) = 2,12841 \cdot 10^{-15}$ y $\text{std } \mu(f, f^*) = 2,56144 \cdot 10^{-15}$; para la figura [4.2](#) de una señal con 56 vértices, un tiempo de ejecución de 0.00720 segundos, $\text{máx } \mu(f, f^*) = 2,13162 \cdot 10^{-14}$, $\text{avg } \mu(f, f^*) = 2,63980 \cdot 10^{-15}$ y $\text{std } \mu(f, f^*) = 3,38336 \cdot 10^{-15}$.

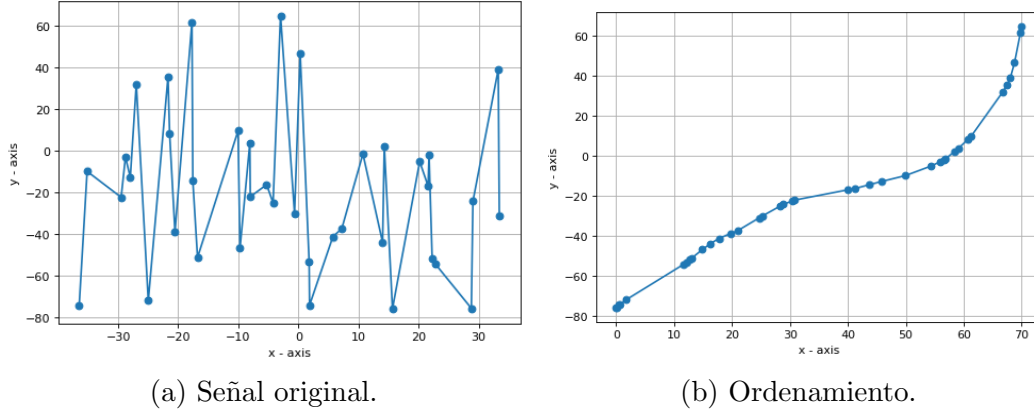


Figura 4.1: Resultado de ordenamiento de una señal LT con 39 vértices.

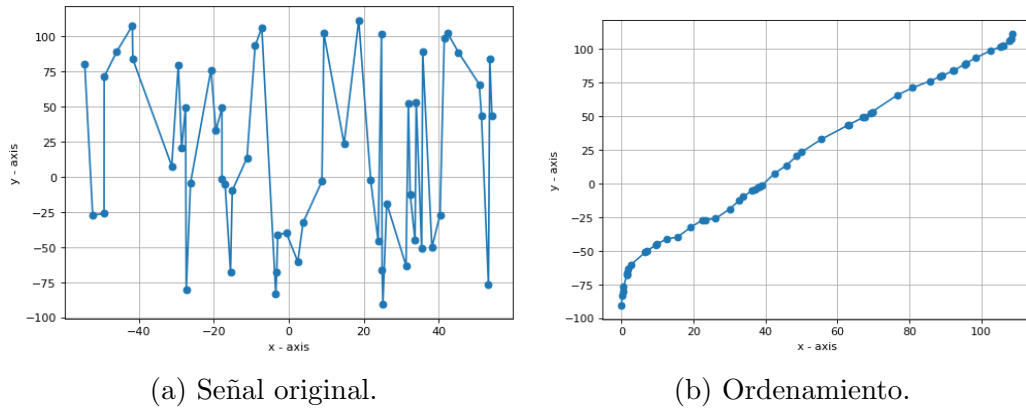


Figura 4.2: Resultado de ordenamiento de una señal LT con 56 vértices.

Una primera evaluación se encuentra en la cuadro 4.1, donde para cada número de vértice fijo (n) se genera una señal aleatoria y se evalúan las métricas definidas, adicionalmente con los tiempos de ejecución (t).

n	$\ \text{máx } \mu(f, f^*) \cdot 10^{-14} \text{ s} \ $	$\ \text{avg } \mu(f, f^*) \cdot 10^{-14} \text{ s} \ $	$\ \text{std } \mu(f, f^*) \cdot 10^{-14} \text{ s} \ $	$t \text{ (s)}$
10	0.35527	0.08010	0.10464	0.00070
50	1.77636	0.30113	0.33623	0.00729
100	3.90799	0.59254	0.79920	0.01688
250	11.4381	1.17023	1.68454	0.09930
500	16.9864	2.37585	3.08299	0.42048
750	33.7508	3.80731	5.35513	0.50584

Cuadro 4.1: Prueba 1, señal aleatoria por número de vértices.

En el cuadro 4.2 por número de vértice fijo se tomaron 15 señales aleatorias y sus máximas diferencias $\text{máx } \mu(f, f^*)$; del conjunto de todas estas se tomó de nuevo el promedio, máximo y desviación estándar, que se denotan por el subíndice 15.

15 señales por número de vértices

n	$\max_{15} \mu(f, f^*) \cdot 10^{-14} \text{ s}$	$\text{avg}_{15} \mu(f, f^*) \cdot 10^{-14} \text{ s}$	$\text{std}_{15} \mu(f, f^*) \cdot 10^{-14} \text{ s}$
10	0.12546	0.19984	0.06026
50	1.60834	2.13163	0.41396
100	3.83915	5.06262	0.65804
250	9.98016	12.9674	1.96141
500	20.6620	28.4217	4.36295
750	36.3339	56.4881	7.51593

Cuadro 4.2: Prueba 2, 15 señales aleatorias por número de vértices.

Además de las pruebas anteriores se debe verificar también el comportamiento del algoritmo para señales con segmentos constantes y/o discontinuidades, este proceso se realiza con señales determinadas previamente. Un ejemplo se encuentra en la figura 4.3, donde la discontinuidad en $x = 2$ se preserva ya que por la franja vertical $(1, 2)$ no cruza ninguna recta; en el ordenamiento se suman los segmentos constantes del nivel vertical $y = 1$. El número de vértices originales es mayor al número de vértices ordenados, esto puede observarse en el nivel $y = 4$ que cuenta con 4 vértices mientras que en la salida solo se tienen 2 para ese mismo nivel. En este ejemplo se obtuvo 0.00063 segundos de tiempo de ejecución, $\max \mu(f, f^*) = 8,88178 \cdot 10^{-16}$, $\text{avg} \mu(f, f^*) = 2,96059 \cdot 10^{-16}$ y $\text{std} \mu(f, f^*) = 4,18691 \cdot 10^{-16}$.

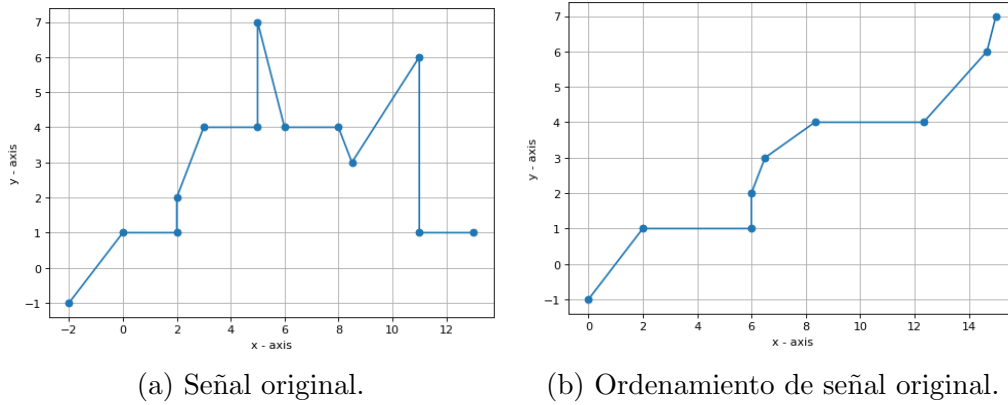


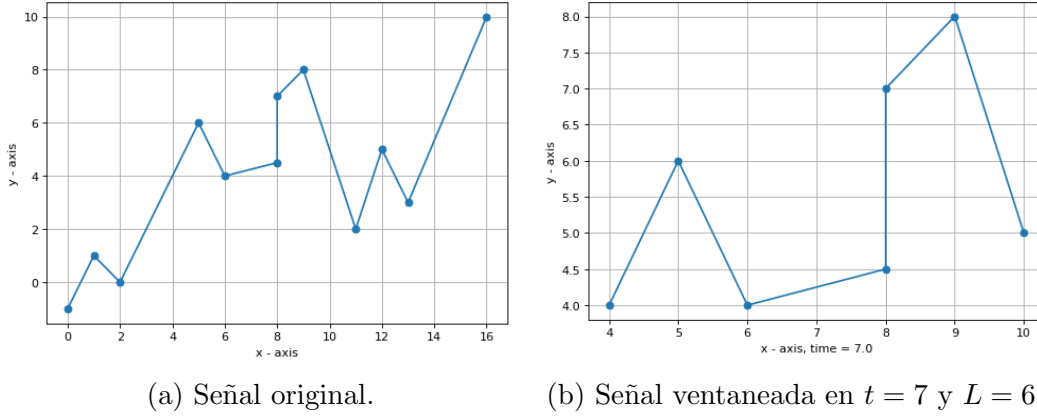
Figura 4.3: Resultado de ordenamiento de una señal LT con 13 vértices.

4.2. Algoritmo de ventana móvil

Un algoritmo preliminar para los filtros de ranqueo consiste en realizar el corrimiento de la ventana de acuerdo a un $\epsilon > 0$. Se deben implementar el operador de ventana $W_{t,L}$ y el funcional de ranqueo φ para así, sobre cada segmento de señal ventaneada, ordenar y aplicar el funcional.

El **algoritmo para el operador de ventana** recibe un tiempo $t \in I$, un ancho de ventana $L \in \mathbb{R}$ y los vértices $\{(x_1, y_1), \dots, (x_n, y_n)\}$ asociados a una señal f , y calcula los vértices $\{(x_1^t, y_1^t), \dots, (x_d^t, y_d^t)\}$ asociados a f_t . Si el ancho de la ventana es menor que la

duración completa de la señal, entonces $d \leq n$. Durante el recorrido de la ventana los vértices de los extremos (izquierdo y derecho) deben ser calculados a partir de la ecuación de la recta en la que se encuentren; como a priori se conocen los vértices originales, se puede diseñar una rutina que busque los vértices de la señal original que se encuentren dentro de la ventana y por último calcular el valor vertical de los extremos. El pseudocódigo del algoritmo se encuentra en A.2.1. En la figura 4.4 se muestra un ejemplo de la implementación en Python de este algoritmo, tenga en cuenta que las figuras están escaladas.

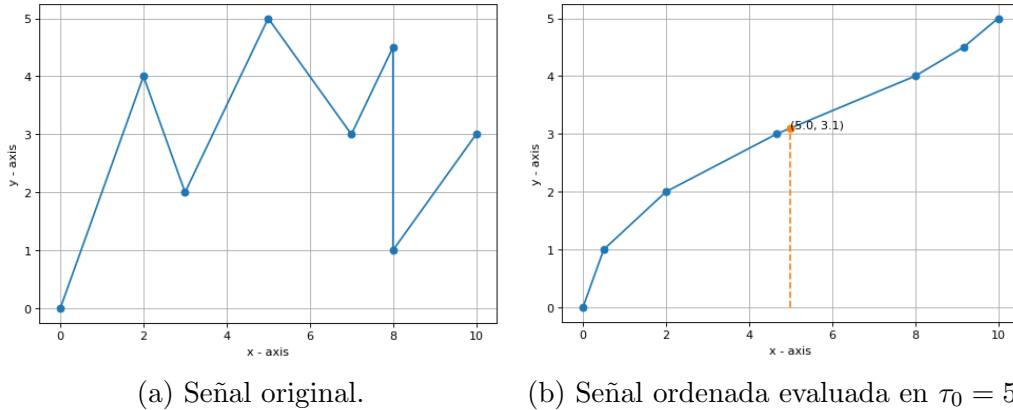


(a) Señal original.

(b) Señal ventaneada en $t = 7$ y $L = 6$.

Figura 4.4: Ejemplo de señal ventaneada en Python.

Para el **algoritmo del funcional de evaluación** se debe fijar un punto $\tau_0 \in [0, L]$. Si se conocen los vértices del ordenamiento, entonces $f^*(\tau_0)$ o es el nivel vertical de un vértice o existen dos vértices consecutivos (x_i^*, y_i^*) y (x_{i+1}^*, y_{i+1}^*) tales que $f^*(\tau_0)$ es un punto en la recta que une a los dos vértices; con lo anterior, un algoritmo que calcule $f^*(\tau_0)$ consiste en calcular la ecuación de la recta en la que se encuentra, o encontrar un vértice con tiempo τ_0 . Si precisamente en τ_0 hay una discontinuidad, se toma el valor correspondiente al límite por izquierda. Para el pseudocódigo ver A.2.2. Un ejemplo de la implementación en Python se encuentra en la figura 4.5.



(a) Señal original.

(b) Señal ordenada evaluada en $\tau_0 = 5$.

Figura 4.5: Ejemplo en Python de funcional de ranqueo en señal ordenada.

El **algoritmo para los filtros de ranqueo con ventana móvil** recibe los vértices $\{(x_1, y_1), \dots, (x_n, y_n)\}$ de una señal f , el ancho de ventana L y el punto τ_0 . Con los métodos

previamente diseñados e implementados, el algoritmo del ventana móvil calcula $f_t(\tau_0)$, es decir la salida del filtro en un tiempo t , realizando corrimientos muy pequeños $t + \epsilon$ y guardando el valor en un vector que corresponderá a la salida. En el apéndice A.2.3 se encuentra el pseudocódigo del algoritmo. En los ejemplos que se presentan posteriormente se fija $\epsilon = 0,01$ s.

En la figura 4.6 se muestra un ejemplo de la implementación en Python. Los recortes de los picos en la señal filtrada es una característica que los filtros de ranqueo analógicos comparten con los filtros de estadísticas de orden, al igual que preservar ciertos segmentos monótonos dependiendo del ancho de la ventana. Por otro lado, en la figura 4.7 se ubica una señal con ruido de alta frecuencia y su versión filtrada; note en este caso la propiedad de suavizado del filtro.

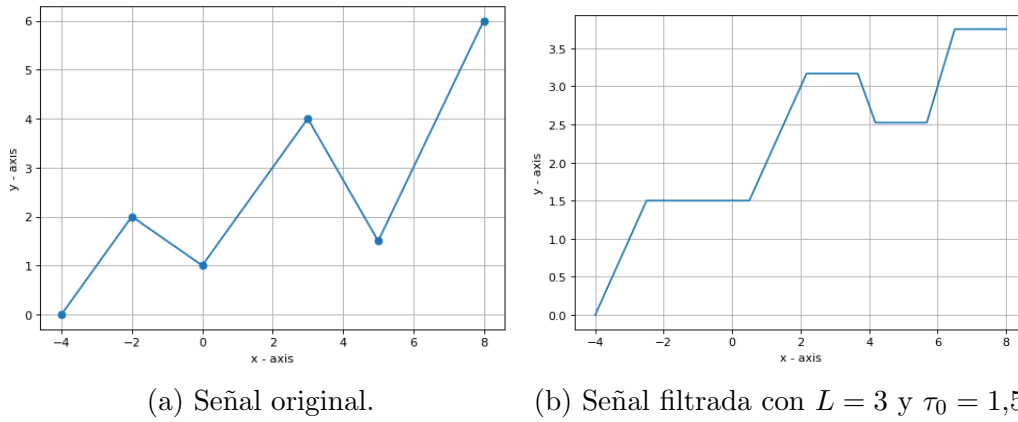


Figura 4.6: Ejemplo en Python de recorte de picos para algoritmo de corrimiento.

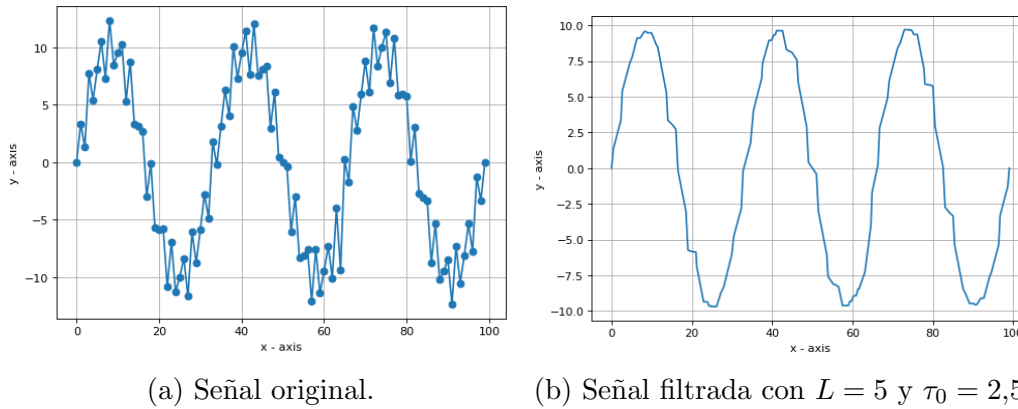


Figura 4.7: Ejemplo en Python de suavizado para algoritmo de corrimiento.

Es claro que entre menor sea el incremento ϵ mejor aproximada estará la salida, mas el algoritmo puede tardar considerablemente más tiempo en especial si la señal es de larga duración. Determinar un corrimiento adecuado dependerá de la duración de la señal de entrada. Este método será útil para realizar una comparación con el algoritmo que calcula finitos vértices de la señal de salida, de modo que proporciona un marco de referencia.

4.3. Algoritmo para filtros de ranqueo

Dado un ancho de ventana $L \in \mathbb{R}$ y un punto de evaluación $\tau_0 \in [0, L]$, el algoritmo de los filtros de ranqueo asocia a los vértices $\{(x_1, y_1), \dots, (x_n, y_n)\}$ de una señal $f: I \rightarrow \mathbb{R}$, finitos vértices $\{(\dot{x}_1, \dot{y}_1), \dots, (\dot{x}_\kappa, \dot{y}_\kappa)\}$ que describen la señal filtrada $\tilde{f}: I \rightarrow \mathbb{R}$. El teorema 3.4.1 proporciona un criterio para determinar los vértices de salida.

En general el algoritmo consiste en un ciclo que realiza tres preguntas para un $t \in I$, de forma que al finalizar el proceso de cálculo del siguiente vértice, t se actualiza y vuelve a comenzar el ciclo. Los tres condicionales principales son los siguientes:

- Si $y_l^t > y_r^t$, es decir si el extremo izquierdo de la ventana se encuentra por encima del extremo derecho.
- Si $y_r^t > y_l^t$, es decir si el extremo derecho de la ventana se encuentra por encima del extremo izquierdo.
- Si $y_r^t = y_l^t$ y $m_r = m_l$, es decir los extremos de la ventana tienen la misma altura y bajarán (o subirán) con la misma pendiente.

Si se tiene $y_l^t > y_r^t$ o $y_r^t > y_l^t$ se pregunta la ubicación de $f_t^*(\tau_0)$; para el caso de $y_l^t > y_r^t$ se definen los condicionales: $y_r^t < f_t^*(\tau_0) < y_l^t$ (1), $y_r^t < y_l^t < f_t^*(\tau_0)$ (2), $f_t^*(\tau_0) < y_r^t < y_l^t$ (3) y $f_t^*(\tau_0)$ se encuentra en un segmento constante por los extremos de la ventana (4). Para el primer caso (1) se sabe que el valor $f_t^*(\tau_0)$ baja con cierta pendiente, luego se calculan todos los posibles tiempos en los cuales la pendiente cambia y se toma el mínimo de ellos, verificando que el tiempo del potencial vértice sea menor que el tiempo al siguiente hito. De igual forma, para el ítem (2) y (3) donde $f_t^*(\tau_0)$ se mantiene constante, se determina si habrá un vértice hasta el siguiente hito, si la respuesta es afirmativa, se calcula el tiempo y se vuelve a evaluar, si la respuesta es negativa, se pasa al siguiente hito. Por último, para el ítem (4), se calcula la distancia al siguiente vértice y se verifica que sea menor al siguiente hito, pues el valor $f_t^*(\tau_0)$ se mantiene constante. El segundo caso $y_r^t > y_l^t$ es completamente análogo al caso $y_l^t > y_r^t$, y para el último simplemente se pasa al siguiente hito.

Una observación importante es que siempre que se calcula un vértice se verifica inmediatamente una posible discontinuidad, para así también agregar ese vértice. Además, también se verifica que si hay cambios abruptos se agregue un vértice, es decir, si en el tiempo t la altura $f_t^*(\tau_0)$ tenía asociada una pendiente, se pasó al siguiente hito y esta cambio de valor, entonces se agrega un vértice. El pseudocódigo se encuentra en A.3.

4.3.1. Protocolo de pruebas y resultados

El protocolo de pruebas del algoritmo para filtros de ranqueo se divide en un muestreo aleatorio del dominio y en la verificación de invarianzas y características principales de los filtros.

Debido a que se tienen dos algoritmos para los filtros de ranqueo, uno por cálculo de vértices y otro por ventana móvil, que serán **algoritmo 1** y **algoritmo 2** respectivamente, se puede plantear un protocolo de pruebas basado en la comparación de las señales obtenidas por los dos filtros. Para esto se necesita una medida de error para comparar las dos señales,

en este caso se implementará una aproximación de la **métrica inducida por la norma uniforme**: si $f, g: I \rightarrow \mathbb{R}$ son dos señales, se define

$$\|f - g\|_\infty := \sup_{t \in I} f(t) - g(t),$$

que representa la *distancia* entre f y g bajo la norma uniforme, realmente la diferencia máxima entre las dos señales. En la práctica no se puede evaluar todo el intervalo I , por lo que se realiza un muestreo aleatorio $D \subseteq I$. Se denotará por f las señales de salida del algoritmo 1 y por g las señales de salida del algoritmo 2. Para el muestreo aleatorio D se toma el máximo valor $f(t) - g(t)$ en la muestra D , que denotaremos por $\max_D(f, g)$, el promedio de las diferencias $\text{avg}_D(f, g)$ y la desviación estándar $\text{std}_D(f, g)$. Al existir infinitos puntos de evaluación en $[0, L]$, se restringirá τ_0 al 25 %, 50 % y 75 % del ancho de la ventana.

10 vértices							
L	τ_0 %	$\max_D(f, g) \cdot 10^{-6}$	$\text{avg}_D(f, g) \cdot 10^{-6}$	$\text{std}_D(f, g) \cdot 10^{-6}$	$\#D$	t_1 (s)	t_2 (s)
0.1	25	0.18223	0.00799	0.02203	1720	0.01937	0.27115
	50	0.01886	0.00379	0.00521	1026	0.02398	0.20189
	75	0.01700	0.00273	0.00361	1009	0.03053	0.2243
5	25	0.00318	0.00016	0.00053	1159	0.01936	0.37509
	50	0.01962	0.00099	0.00207	1367	0.02539	0.34252
	75	0.09546	0.00265	0.01221	951	0.02232	0.13334
10	25	0.00950	0.00011	0.00079	1197	0.02272	0.3319
	50	0.21466	0.00194	0.00962	1013	0.02028	0.3765
	75	0.00458	0.00007	0.00045	1631	0.01457	0.37051
20	25	0.00315	0.00015	0.00056	1637	0.02858	0.46491
	50	0.01037	0.00055	0.00188	1373	0.02596	0.41971
	75	0.10562	0.00169	0.00950	1209	0.02289	0.43047

Cuadro 4.3: Pruebas de filtro de ranqueo para 10 vértices.

50 vértices							
L	τ_0 %	$\max_D(f, g) \cdot 10^{-6}$	$\text{avg}_D(f, g) \cdot 10^{-6}$	$\text{std}_D(f, g) \cdot 10^{-6}$	$\#D$	t_1 (s)	t_2 (s)
0.5	25	6.34872	0.20459	0.66184	8609	0.14571	1.61536
	50	6.92953	0.20098	0.65186	5379	0.11186	1.41705
	75	7.13172	0.19887	0.64402	7550	0.14357	1.69292
25	25	2.25312	0.02891	0.19279	5303	0.39978	3.62654
	50	1.66327	0.02840	0.11617	6376	0.41233	3.4148
	75	3.84786	0.05257	0.26200	7234	0.40226	3.51878
50	25	2.08518	0.03478	0.20232	9484	0.97511	7.25569
	50	0.27437	0.00413	0.02304	9105	0.97644	7.40104
	75	2.54971	0.04411	0.25454	9315	1.04121	7.30954
100	25	1.16647	0.01556	0.09341	6808	1.84715	16.9809
	50	0.28362	0.00390	0.02260	7283	1.97361	17.0943
	75	0.13738	0.00309	0.01469	5452	2.08652	17.0727

Cuadro 4.4: Pruebas de filtro de ranqueo para 50 vértices.

100 vértices							
L	τ_0 %	$\text{máx}_D(f, g) \cdot 10^{-6}$	$\text{avg}_D(f, g) \cdot 10^{-6}$	$\text{std}_D(f, g) \cdot 10^{-6}$	$\#D$	t_1 (s)	t_2 (s)
1	25	257.631	3.82261	15.1911	12223	0.57009	5.70027
	50	243.407	3.51900	14.1835	19593	0.31119	3.81679
	75	257.632	4.01562	16.1977	17330	0.35817	3.83382
50	25	17.8052	0.22251	1.30436	18364	2.37925	16.3116
	50	12.1375	0.08266	0.58795	19344	2.52292	16.4044
	75	11.0282	0.19116	0.91399	18925	2.5554	16.3960
100	25	9.56524	0.11727	0.69078	16060	7.67973	42.2604
	50	0.57470	0.01075	0.04818	17761	8.22792	42.26865
	75	10.7592	0.08805	0.59294	17988	8.91152	42.24934
200	25	0.83450	0.01847	0.07965	9905	21.4739	113.417
	50	0.43027	0.00596	0.03667	11054	24.5056	113.47
	75	1.47271	0.03597	0.15245	19105	23.4575	113.27

Cuadro 4.5: Pruebas de filtro de ranqueo para 100 vértices.

Los cuadros 4.3, 4.4 y 4.5 muestran los resultados para $\text{máx}_D(f, g)$, $\text{avg}_D(f, g)$ y $\text{std}_D(f, g)$ en señales con 10, 50 y 100 vértices; $\#D$ es la cantidad de puntos de evaluación que se toman; además se presentan los tiempos de ejecución para el algoritmo 1 (t_1) y el algoritmo 2 (t_2). Las pruebas fueron realizadas a señales aleatorias con los mismos parámetros de 4.1.1, es decir con componentes $[x_1, \dots, x_n] \sim \text{Unif}(-n, n)$ y $[y_1, \dots, y_n] \sim \text{Unif}(-2n, 2n)$. Por cada número de vértice fijo se filtró con diferentes anchos de ventana, estos a su vez aumentan de acuerdo al número de vértices ya que las señales tienden a tener larga duración.

Las características propias de los filtros de ranqueo se verificaron para el algoritmo de corrimiento, ahora estas mismas se deben corroborar para el método que calcula finitos vértices, es decir el algoritmo 1. Recuerde que estas propiedades son: preservar ciertas discontinuidades de acuerdo al ancho de la ventana y el punto de evaluación, preservar algunos segmentos monótonos, recortar los picos de las señales de entrada y el suavizado.

En la figura 4.8 se presenta una señal filtrada por un ancho $L = 5$ y $\tau_0 = 1$; considerar τ_0 menor que la mediana en este caso resulta en que la discontinuidad en $t = 30$ en la salida esté ubicada en $t = 31,5$; en la figura 4.9 se muestra el efecto de recorte del filtro mediana.

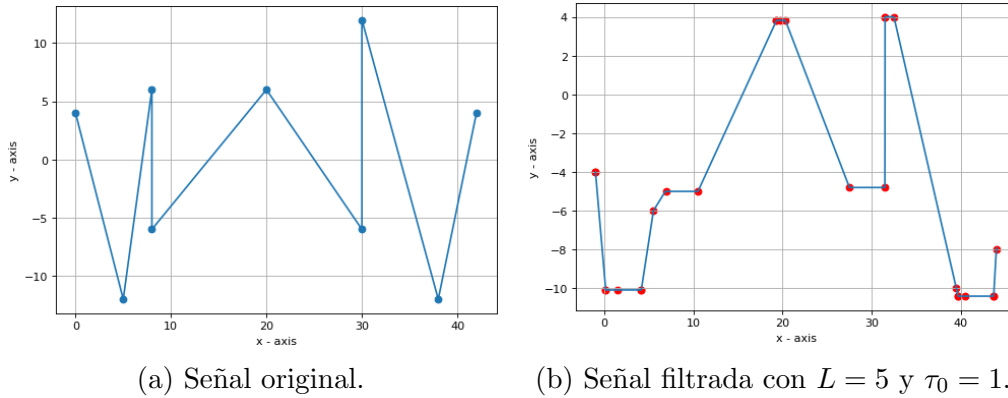


Figura 4.8: Ejemplo en Python de discontinuidad y recorte de picos para algoritmo 1.

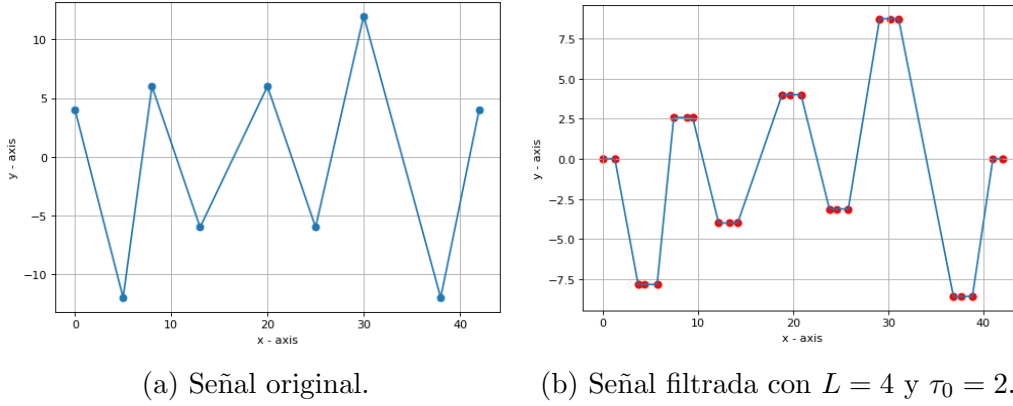


Figura 4.9: Ejemplo en Python de recorte de picos para algoritmo 1.

Note en las figuras 4.10 y 4.11 el efecto de suavizado sobre señales con ruido; la primera es la misma señal de la figura 4.7 pero esta vez con los vértices calculados del algoritmo 1. En la segunda figura las señales de entrada y salida filtrada se superponen y el ruido fue generado de forma aleatoria con distribución normal.

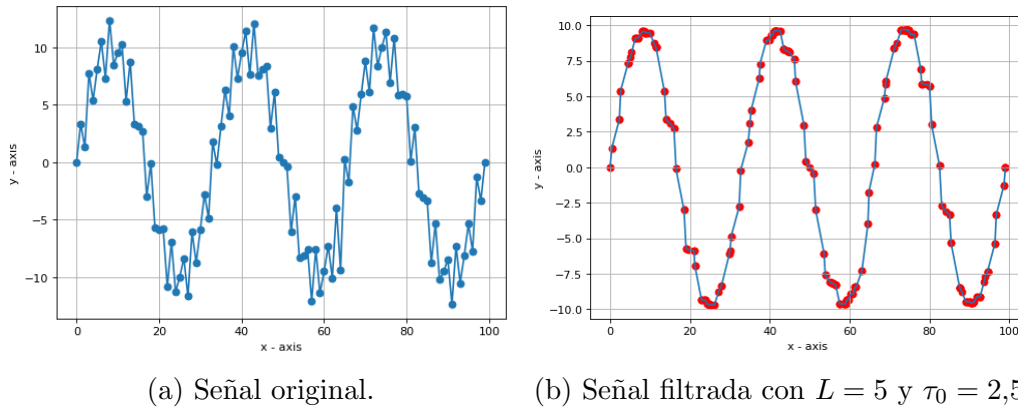
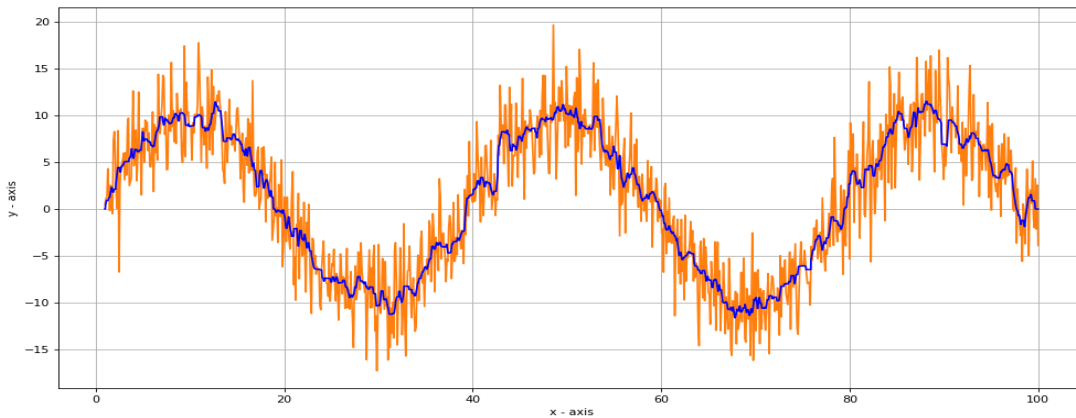


Figura 4.10: Ejemplo en Python de suavizado por cálculo de vértices, algoritmo 1.

Figura 4.11: Señales de entrada y salida filtrada con $L = 1$ y $\tau_0 = 0,5$, algoritmo 1.

4.4. Análisis de resultados y consideraciones prácticas

Los resultados presentados anteriormente para la verificación y evaluación de los algoritmos traen como consecuencia la siguiente valoración. En el caso del **algoritmo de ordenamiento**:

- Las diferencias máximas $\max \mu(f, f^*)$ incrementan con el aumento de vértices, no obstante esta sigue siendo del orden de 10^{-14} , lo que indica que la señal calculada corresponde con el ordenamiento. El mismo fenómeno ocurre con el promedio y la desviación estándar: aumentan con el número de vértices mas siguen en el mismo orden. En general la desviación se mantiene muy por debajo de las demás métricas, con lo cual las diferencias máximas tienden a agruparse alrededor del promedio.
- Claramente el tiempo de ejecución aumenta con el número de vértices. Se obtienen tiempos menores a 0.01 segundos para un número de vértices menor o igual a 50.
- El algoritmo cumple las propiedades descritas en el capítulo 2, suma los segmentos constantes que se presentan en un nivel vertical, las discontinuidades que induzcan una preimagen vacía de algún intervalo implican discontinuidades en el ordenamiento, los vértices y las pendientes corresponden con los del teorema 2.3.1 y en general el número de vértices de la entrada es mayor o igual al número de vértices del ordenamiento.

Para los **métodos de filtros de ranqueo**:

- El algoritmo 2 aproxima bastante bien la salida de los filtros de acuerdo al corrimiento ϵ respecto a la duración de la señal, sin embargo los resultados muestran un aumento sustancial en el tiempo de ejecución para $\epsilon = 0,01$ fijo. Puede aumentarse este valor pero eso implicará mayor error a la salida.
- El protocolo de pruebas para el algoritmo 1 se basa en la medida de error entre dos señales en un dominio discreto aleatorio, interpolando linealmente entre los vértices del algoritmo 1 y tomando la máxima diferencia. Al utilizar una distribución uniforme del intervalo este muestreo permite evaluar en varios puntos de una misma recta y así medir, por ejemplo, la disparidad entre ellas; de esa forma se logra una mejor aproximación a la diferencia total en todo el intervalo.
- El tiempo de ejecución de ambos algoritmos no aumenta significativamente con el cambio de ranqueo τ_0 , mientras que sí lo hace con el aumento del ancho de la ventana. Esto se debe a que entre más aumente la ventana, más vértices se tendrán que ordenar con el algoritmo de ordenamiento.
- La comparación entre los tiempos de ejecución indica que no solo es considerablemente más rápido el algoritmo 1 como era de esperar, si no que además la máxima distancia entre las dos señales en el muestreo sigue manteniéndose esta vez en el orden de 10^{-6} ; este valor tiende a disminuir con el aumento de la ventana, esta también es la tendencia del promedio y la desviación.
- Los algoritmos 1 y 2 cumplen todas las propiedades asociadas a los filtros de ranqueo, incluida la de suavizado y su efecto en señales con ruido.

- El número de vértices de la señal de salida depende tanto de la ventana como del punto de evaluación, puede ser mayor o menor que el número de vértices de la entrada.

En la práctica si lo que se quiere es suavizar una señal con ruido se utilizan ventanas de corta duración, es por esto que se pueden plantear varias estrategias de filtrado, una de ellas es la siguiente: si la señal es de corta duración, se puede realizar un aumento proporcional del tamaño de la misma y así aumentar el ancho de la ventana, esto con el fin de aprovechar al máximo el filtro dado que a pequeñas ventanas aumenta el error de la máxima diferencia.

4.5. Aplicación de los algoritmos

Los filtros de ranqueo tienen ventajas sobre otros filtros de ventana móvil como el promedio, ya que son más inmunes a los picos de gran magnitud que sí afectan el promedio. Ahora bien, una señal discreta puede ser convertida en una señal analógica por interpolación lineal entre sus muestras, y luego la señal analógica puede ser filtrada por los filtros de ranqueo. En un ejemplo sencillo se muestrea uniformemente en tiempos enteros

$$g(t) = f(n) + [f(n+1) - f(n)](t - n), \quad \forall t \in [n, n+1],$$

donde f es la señal que está siendo aproximada. La desventaja del método es que el error puede ser grande ya que la señal en principio puede tomar valores de gran magnitud entre muestras. Aún así, la interpolación entre los datos discretos ofrece un enfoque novedoso para suavizar series de tiempo.

Los filtros de ventana móvil son utilizados en economía y finanzas como herramienta para estudiar tendencias. Del servicio YAHOO! FINANCE se extrajeron las bases de datos que contienen la información sobre el valor de la acción de MICROSOFT (MSFT) e IBM en el último año. Los valores de estas series de tiempo corresponden al cierre del mercado, se normalizaron y se filtró, para ambas series, con una ventana (*look back* en series de tiempo) de 50 días y punto de evaluación la mediana $\tau_0 = 25$ (figuras 4.12 y 4.13).

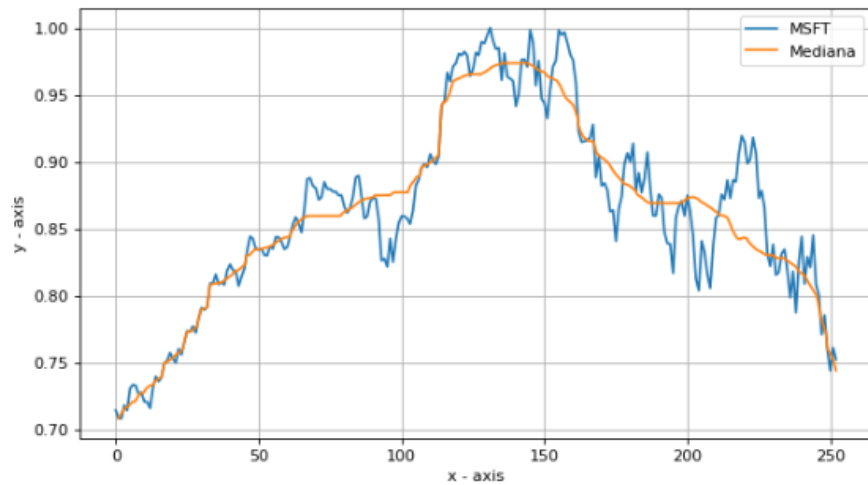


Figura 4.12: Filtro mediana analógico de ancho $L = 50$ de la acción MSFT.

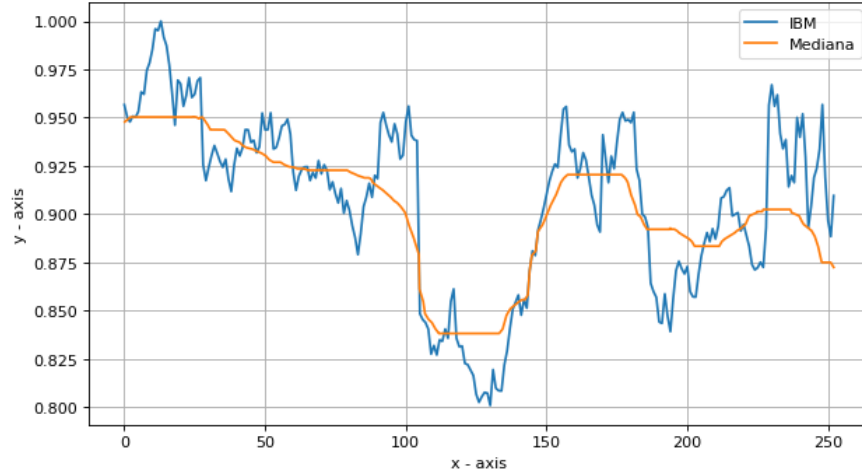


Figura 4.13: Filtro mediana analógico de ancho $L = 50$ de la acción IBM.

Otra posible aplicación se encuentra en la serie de tiempo de la temperatura global proporcionada por NASA y que se puede encontrar en [28]. La base de datos es diaria y va desde 1880 hasta 2016. A priori, luego del filtrado con una ventana amplia, se esperaría una clara tendencia al aumento en el último siglo. En la figura 4.14 se presenta la serie de tiempo y la señal filtrada con $L = 200$ y $\tau_0 = 100$.

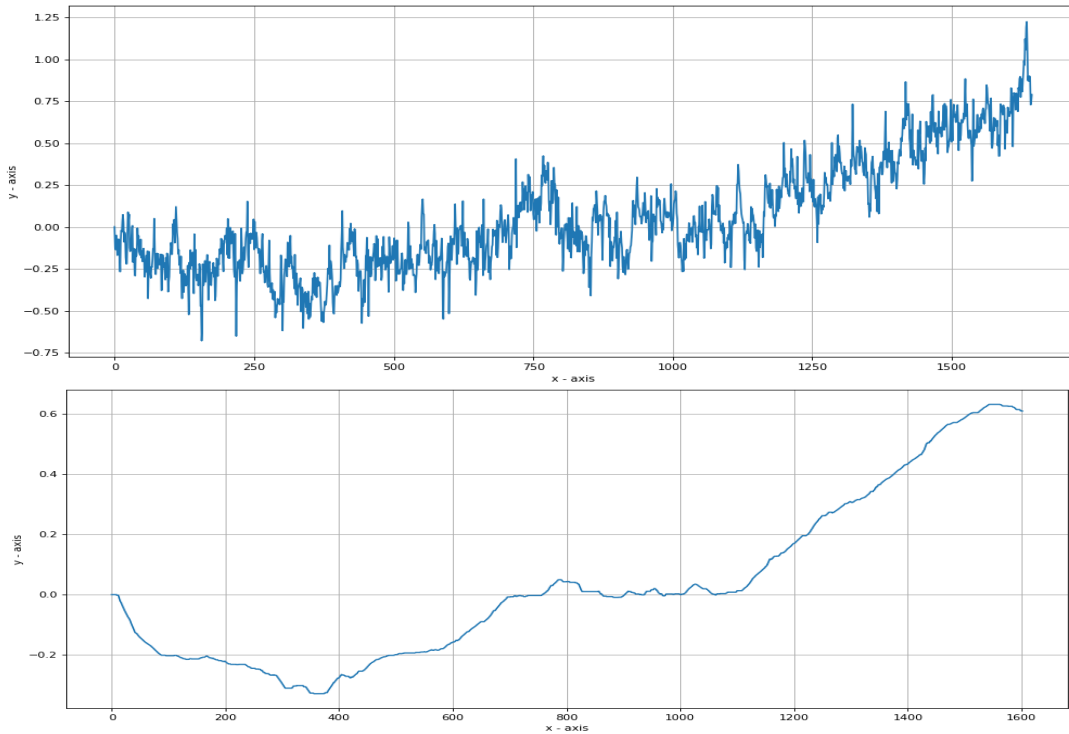


Figura 4.14: Filtro mediana analógico de la temperatura de la tierra 1880-2016 con $L = 200$.

Diferentes puntos de evaluación pueden llegar a ser de interés, por ejemplo el máximo, el mínimo o cualquier otra estadística o porcentaje del ancho de ventana. Por otro lado, los

algoritmos de ordenamiento y evaluación tienen aplicaciones en áreas en las que se necesite obtener la mediana u otra estadística de una señal analógica, e.g. si se quiere conocer la concentración de un medicamento para la cual se mantuvo la mitad del tiempo por encima y la mitad del tiempo por debajo. De la base de datos con la información de la materia particulada 2.5 en el aire de Beijing entre 2014 y 2021, y la aplicación del algoritmo de ordenamiento y funcional, se obtuvo el resultado de la figura 4.15.

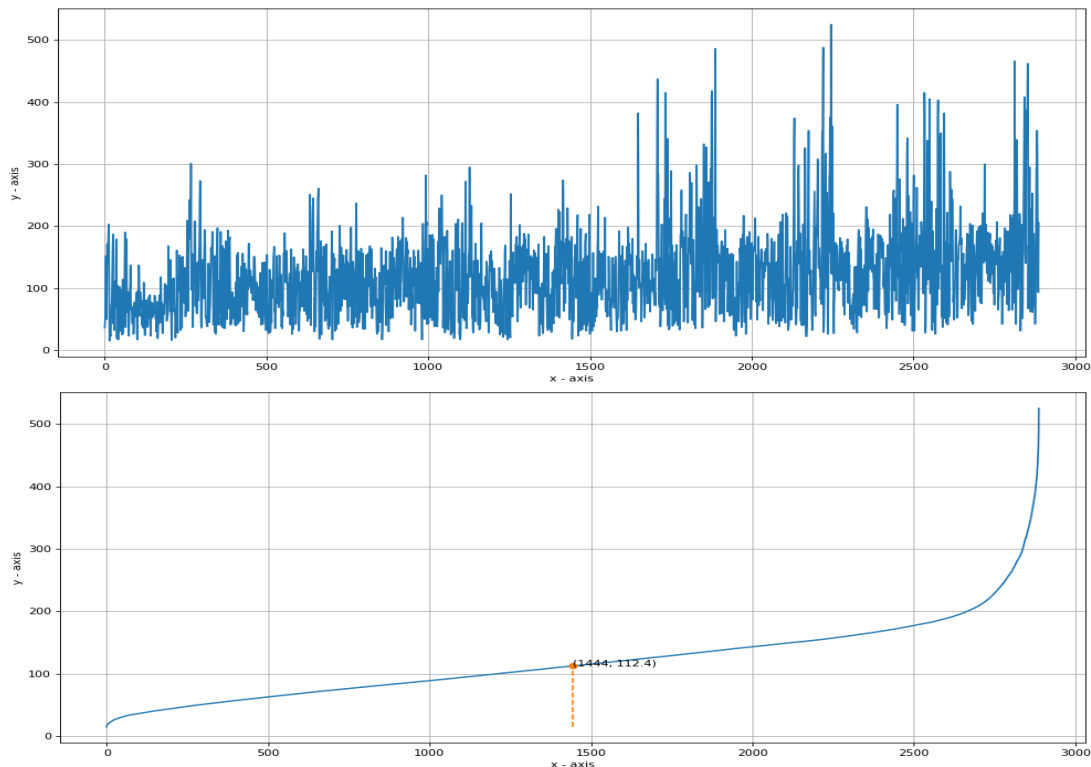


Figura 4.15: Ordenamiento y toma de la mediana para pm25 en Beijing.

Observación. Note cómo los picos de magnitud alta se distribuyen a la derecha del dominio $[0, L]$; esta es la ventaja de los filtros de ranqueo y escoger el punto de evaluación.

La mediana en este caso indica que en ese intervalo de tiempo la ciudad de Beijing estuvo la mitad del tiempo encima del nivel 112,4, que es perjudicial para niños y adultos con problemas respiratorios, vea [29]. Estas son solo algunas aplicaciones de los algoritmos implementados.

Capítulo 5

Conclusiones

En este trabajo se estudiaron los filtros de ranqueo analógicos para entradas lineales a trozos, demostrando múltiples teoremas y desarrollando algoritmos que permitieron su implementación en software. A continuación, se hace una síntesis de las principales ideas y conclusiones del trabajo.

Ordenamiento monótono y algoritmo

En cuanto a la parte teórica de la distribución y el ordenamiento, se probaron tres teoremas que conciernen a las propiedades de estas dos funciones. El primer objetivo del trabajo se demostró en el teorema 2.3.1; se obtuvo una expresión concreta para los vértices del ordenamiento, además, un análisis por casos permitió concluir que f y f^* tienen los mismos niveles verticales, y que los tiempos de los vértices pueden ser calculados midiendo el tiempo que la función original f pasa por debajo de cada nivel. Con lo anterior se pudo diseñar e implementar el algoritmo de ordenamiento de la sección 4.1, que corresponde al segundo objetivo.

La construcción matemática del algoritmo y el protocolo de verificación implementado confirman la validez del mismo. Es de notar que puede que el algoritmo llegue a una partición \mathcal{P}^* que no sea la más gruesa; el caso no tiene una significancia particular, solo implica que se podría obviar un vértice. Igualmente, la verificación tiene una justificación rigurosa: la equimedibilidad nos permite verificar si f^* efectivamente corresponde con el ordenamiento de f ; esto se pudo evidenciar con la medida de error $\mu(f, f^*)$ de la diferencia máxima, que si bien aumenta con el número de vértices de la señal nunca llega a ser significativo, y también con la verificación de las propiedades de segmentos constantes y discontinuidades. Similarmente, el tiempo de ejecución aumenta con el número de vértices como era de esperarse.

Filtros de ranqueo

Se definieron rigurosamente los filtros de orden analógicos por medio de los operadores $W_{t,L}$, \mathbf{o} y el funcional de Stieltjes φ , todos actuando sobre \mathcal{L} ; esto dio paso a considerar otros funcionales además del ranqueo, como por ejemplo aquellos cuya semilla es de clase C^1 , constante a trozos, etc.

El tercer objetivo, que es el teorema 3.4.1, fue un proceso largo que implicó la creación de videos de filtrado utilizando el algoritmo de corrimiento para de esta forma desarrollar intuición sobre el fenómeno. La observación crucial fue el hecho de que al mover la ventana, si la altura de interés se encuentra entre los niveles de los extremos, entonces el tiempo por debajo de esta altura aumenta o disminuye. La demostración llevó a un análisis de cómo pueden ser calculados los vértices de la señal de salida, de acuerdo a unas señales ventaneadas llamadas hitos.

En cuanto al algoritmo de ventana móvil, este es una herramienta útil de visualización y un medio de comparación. Los tiempos de ejecución para este algoritmo son bastante altos y más si se quiere una señal de salida muy precisa, pues en ese caso el corrimiento ϵ disminuye y así aumentan las señales ventaneadas; para el corrimiento y el algoritmo de cálculo de vértices, los tiempos de ejecución aumentan de acuerdo al ancho de la ventana.

El medio de comparación se da al implementar una aproximación a la norma uniforme por un muestreo aleatorio del intervalo, definiendo de nuevo la máxima diferencia, su promedio y desviación. En general se tienen valores bastante bajos del orden de 10^{-6} , lo que quiere decir que los algoritmos producen salidas aproximadamente iguales. Las simulaciones realizadas del filtro conservan las características ya conocidas en el filtro mediana discreto, como por ejemplo cortar picos, preservar segmentos monótonos y ciertas discontinuidades, suavizar señales, etc.

Los objetivos general y específicos formulados en el comienzo del trabajo se cumplieron de manera satisfactoria.

Posibles líneas de investigación

A lo largo del trabajo varias preguntas aún abiertas fueron formuladas y potencialmente pueden llegar a ser líneas de investigación. Una cuestión más bien teórica es la siguiente: si bien los subconjuntos contables de \mathbb{R} no forman una σ -álgebra y por lo tanto la noción usual de medida no está definida para ellos, podría pensarse en una forma alternativa para definir el ordenamiento de funciones cuyo dominio sea enumerable. Por otro lado, puede definirse una clase diferente de filtros con el ordenamiento simétrico.

Es claro que la linealidad a trozos es oportuna para los filtros de ranqueo e incluso combinación lineal finita, pero no es claro para qué otros tipos de funcionales sea ventajoso este tipo de aproximación a la hora de la implementación. También puede pensarse en criterios para tomar una combinación lineal infinita como funcional aplicado al ordenamiento, claramente deberá existir una restricción sobre los coeficientes. Se puede considerar un análisis con diferentes tipos interpolación, e.g. cuadrática, cúbica, etc.

Los algoritmos de ordenamiento y filtrado de ranqueo por cálculo de vértices pueden ser optimizados y mejorar sustancialmente sus tiempos de ejecución. De esta forma se podría pensar en una implementación en tiempo real en un sistema embebido, por ejemplo para aplicaciones en señales biológicas.

Apéndice A

Algoritmos

Los pseudocódigos de los algoritmos diseñados e implementados en el capítulo 4 se presentan en este apéndice.

A.1. Algoritmo de ordenamiento

El algoritmo de ordenamiento calcula los vértices de salida $(\mathbf{x}^*, \mathbf{y}^*)$ a partir de unos vertices de entrada (\mathbf{x}, \mathbf{y}) . La rutina consiste en calcular el tiempo que la señal pasa entre niveles verticales consecutivos y así construir el ordenamiento.

Algorithm 1 Non-decreasing Rearrangment for Piecewise Linear Functions

```
1: procedure PL-REARRANGMENT( $\mathbf{x}, \mathbf{y}$ )
2:    $\mathcal{P}' \leftarrow [y_j]_{j=1}^l$ 
3:   compute  $\mu(f^{-1}(\{y_1\}))$ 
4:   for each pair  $(y_j, y_{j+1})$  do
5:     for each pair of consecutive vertices  $((x_i, y_i), (x_{i+1}, y_{i+1}))$  do
6:       if the line through  $((x_i, y_i), (x_{i+1}, y_{i+1}))$  crosses  $(y_j, y_{j+1})$  then
7:         compute  $m_i$  and  $b_i$ 
8:          $\mu(f^{-1}(y_j, y_{j+1}]) \leftarrow \mu(f^{-1}(y_j, y_{j+1})) + \left| \frac{y_j - b_i}{m_i} - \frac{y_{j+1} - b_i}{m_i} \right|$ 
9:       else if  $y_i = y_{i+1} = y_{j+1}$  and  $x_i \neq x_{i+1}$  then
10:         $\mu(f^{-1}(y_j, y_{j+1}]) \leftarrow \mu(f^{-1}(y_j, y_{j+1})) + (x_{i+1} - x_i)$ 
11:         $\mu(f^{-1}(\{y_{j+1}\})) \leftarrow \mu(f^{-1}(\{y_{j+1}\})) + (x_{i+1} - x_i)$ 
12:       else
13:         continue.
14:       end if
15:     end for
16:   end for
17:   compute time of vertices with  $T_f(y_j) = \mu(f^{-1}(\{y_1\})) + \sum_{k=2}^j \mu(f^{-1}((y_{k-1}, y_k]))$ ,
   and if constant segment add  $T_f(y_j) - \mu(f^{-1}(\{y_j\}))$ 
18:   determine all vertical levels of the rearrangment
19:   return ordered vertices  $\mathbf{x}^*, \mathbf{y}^*$ 
20: end procedure
```

A.2. Algoritmos para ventana móvil

A.2.1. Algoritmo de ventana

El algoritmo de ventana recibe los vértices de una señal de entrada (\mathbf{x}, \mathbf{y}) , el ancho de la ventana W y el tiempo de evaluación t ; retorna los vértices de la señal ventaneada $(\mathbf{x}_t, \mathbf{y}_t)$.

Algorithm 2 Window Operator for Piecewise Linear Signals

```

1: procedure WINDOW( $\mathbf{x}, \mathbf{y}, W, t$ )
2:    $L \leftarrow W/2$ 
3:    $\mathbf{x}_t \leftarrow [t - L/2]$ 
4:    $\mathbf{y}_t \leftarrow []$ 
5:   compute left vertex and add to  $\mathbf{y}_t$ 
6:   for each vertex  $(x_i, y_i)$  do
7:     if  $t - L/2 < x_i < t + L/2$  then
8:        $\mathbf{x}_t.append(x_i)$ 
9:        $\mathbf{y}_t.append(y_i)$ 
10:    else
11:      continue.
12:    end if
13:  end for
14:  compute right vertex and add to  $\mathbf{y}_t$ 
15:  return  $\mathbf{x}_t, \mathbf{y}_t$ 
16: end procedure

```

A.2.2. Algoritmo de funcional de ranqueo

El algoritmo del funcional de ranqueo recibe los vértices $(\mathbf{x}_t, \mathbf{y}_t)$ y el punto de evaluación τ_0 , para calcula el valor $f^*(\tau_0)$.

Algorithm 3 Rank Functional for Windowed Signals

```

1: procedure RANK-FUNCTIONAL( $\mathbf{x}_t, \mathbf{y}_t, \tau_0$ )
2:    $y_d, m_i, b_i \leftarrow 0, 0, 0$ 
3:   for each pair of consecutive vertices  $((x_{i,t}, y_{i,t}), (x_{i+1,t}, y_{i+1,t}))$  do
4:     if  $x_{i,t} = x_{i+1,t} = \tau_0$  then
5:        $y_d \leftarrow y_{i,t}$ 
6:       break
7:     else if  $x_{i,t} < \tau_0 < x_{i+1,t}$  then
8:        $m_i \leftarrow \frac{y_{i+1,t} - y_{i,t}}{x_{i+1,t} - x_{i,t}}$ 
9:        $b_i \leftarrow (-m_i \cdot x_{i,t}) + y_{i,t}$ 
10:       $y_d \leftarrow m_i \tau_0 + b_i$ 
11:    else if  $x_{i,t} \neq x_{i+1,t}$  and  $(x_{i,t} = \tau_0 \text{ or } x_{i+1,t} = \tau_0)$  then
12:      if  $x_{i,t} = \tau_0$  then
13:         $y_d \leftarrow y_{i,t}$ 
14:      else if  $x_{i+1,t} = \tau_0$  then
15:         $y_d \leftarrow y_{i+1,t}$ 
16:      end if
17:      break
18:    else
19:      continue
20:    end if
21:  end for
22:  return  $y_d, (x_{i,t}, y_{i,t}), (x_{i+1,t}, y_{i+1,t})$ 
23: end procedure

```

A.2.3. Algoritmo de filtros de ranqueo con ventana móvil

El algoritmo de corrimiento calcula a partir unos vértices de entrada (\mathbf{x}, \mathbf{y}) asociados a una señal f , un ancho de ventana L y $\tau_0 \in [0, L]$, una lista de vértices $(\mathbf{x}_s, \mathbf{y}_s)$ que corresponden a una señal de salida \tilde{f} . El algoritmo utiliza un corrimiento pequeño $\epsilon > 0$ y los algoritmos diseñados anteriormente.

Algorithm 4 Moving Window Filter by Small ϵ

```

1:  $t \leftarrow x_1$ 
2:  $x_{out}, y_{out} \leftarrow [], []$ 
3: while  $t \leq x_f$  do
4:    $\mathbf{x}_t, \mathbf{y}_t \leftarrow \text{Window}(\mathbf{x}, \mathbf{y}, L, t)$ 
5:    $\mathbf{x}_t^*, \mathbf{y}_t^* \leftarrow \text{PL-Rearrangment}(\mathbf{x}_t, \mathbf{y}_t)$ 
6:    $y_d \leftarrow \text{Rank-functional}(\mathbf{x}_t^*, \mathbf{y}_t^*, \tau_0)$ 
7:    $y_{out}.\text{append}(y_d)$ 
8:    $x_{out}.\text{append}(t)$ 
9:    $t \leftarrow t + \epsilon$ 
10: end while

```

A.3. Algoritmo de cálculo de vértices

El algoritmo para el cálculo de vértices de la señal de salida de nuevo recibe unos vértices de entrada (\mathbf{x}, \mathbf{y}) , un ancho de ventana L y un punto de evaluación τ_0 . Utiliza el método explicado en el teorema 3.4.1 y en la sección 4.3.

Algorithm 5 Analog rank-filter by vertex calculation

```

1:  $t \leftarrow x_1$ 
2:  $x_{out}, y_{out} \leftarrow [], []$ 
3: while  $t \leq x_f$  do
4:    $\mathbf{x}_t, \mathbf{y}_t \leftarrow \text{Window}(\mathbf{x}, \mathbf{y}, L, t)$ 
5:    $\mathbf{x}_t^*, \mathbf{y}_t^* \leftarrow \text{PL-Rearrangment}(\mathbf{x}_t, \mathbf{y}_t)$ 
6:    $y_d \leftarrow \text{Rank-functional}(\mathbf{x}_t^*, \mathbf{y}_t^*, \tau_0)$ 
7:   if  $y_l > y_r$  then
8:     determine location of  $y_d$  w.r.t.  $y_l$  and  $y_r$  and compute vertex or next milestone.
9:     if any add vertex to output  $x_{out}, y_{out}$ 
10:      update  $t$ 
11:   else if  $y_r > y_l$  then
12:     same scenario as before, only that  $y_d$  can go up, compute vertex or next milestone
13:     if any add vertex to output  $x_{out}, y_{out}$ 
14:       update  $t$ 
15:   else if  $y_r = y_l$  then
16:     move to next milestone, i.e. update  $t$ 
17:   end if
18: end while

```

Apéndice B

Demostraciones

Los teoremas sobre las propiedades de la función distribución T_f y el ordenamiento f^* enunciados en el capítulo 2 se demuestran en este apéndice.

B.1. Prueba del teorema 2.2.1

Teorema 2.2.1 (Propiedades de la distribución).

1. T_f es monótona creciente y continua por derecha.
2. $T_f(y_0^-) = T_f(y_0) - \mu(f^{-1}(\{y_0\}))$.
3. Si f es discontinua en $x_0 \in I$, con $y_2 = f(x_0^+)$ y $y_1 = f(x_0^-)$, y además $f^{-1}((y_1, y_2)) = \emptyset$, entonces T_f es constante en el intervalo (y_1, y_2) con valor $T_f(y_1)$.

Demostración.

1. Si $y_i \leq y_j$ entonces

$$\{x \in I: f(x) \leq y_i\} \subseteq \{x \in I: f(x) \leq y_j\},$$

luego como la medida μ es monótona, se sigue que $T_f(y_i) \leq T_f(y_j)$ y así T_f es monótona creciente. Para la continuidad por derecha sea $y_0 \in [A, B]$ y defina

$$A_n := \left\{x \in I: y_0 < f(x) \leq y_0 + \frac{1}{n}\right\};$$

note que $\{A_n\}_{n \in \mathbb{N}}$ es una sucesión decreciente luego por el teorema 1.1.3 se sigue que

$$\begin{aligned} \lim_{n \rightarrow \infty} T_f\left(y_0 + \frac{1}{n}\right) &= \lim_{n \rightarrow \infty} \mu\left(\left\{x \in I: f(x) \leq y_0 + \frac{1}{n}\right\}\right) \\ &= \lim_{n \rightarrow \infty} \mu(\{x \in I: f(x) \leq y_0\}) + \mu(A_n) \\ &= T_f(y_0) + \lim_{n \rightarrow \infty} \mu(A_n) \\ &= T_f(y_0) + \mu\left(\bigcap_{n=1}^{\infty} A_n\right) \\ &= T_f(y_0), \end{aligned}$$

puesto que $\cap_{n=1}^{\infty} A_n = \emptyset$; por lo tanto T_f es continua por derecha.

2. Sea $y_0 \in [A, B]$ y defina

$$A_n := \left\{ x \in I : y_0 - \frac{1}{n} < f(x) \leq y_0 \right\};$$

de nuevo $\{A_n\}_{n \in \mathbb{N}}$ es una sucesión decreciente y por el teorema 1.1.3 se tiene:

$$\begin{aligned} \lim_{n \rightarrow \infty} T_f \left(y_0 - \frac{1}{n} \right) &= \lim_{n \rightarrow \infty} \mu \left(\left\{ x \in I : f(x) \leq y_0 - \frac{1}{n} \right\} \right) \\ &= \lim_{n \rightarrow \infty} \mu (\{x \in I : f(x) \leq y_0\}) - \mu (A_n) \\ &= T_f(y_0) - \lim_{n \rightarrow \infty} \mu (A_n) \\ &= T_f(y_0) - \mu \left(\bigcap_{n=1}^{\infty} A_n \right) \\ &= T_f(y_0) - \mu (f^{-1}(\{y_0\})), \end{aligned}$$

ya que $\cap_{n=1}^{\infty} A_n = f^{-1}(\{y_0\})$.

3. Sea $u \in (y_1, y_2)$, entonces $T_f(u) = T_f(y_1) + \mu(f^{-1}((y_1, u])) = T_f(y_1)$, pues por hipótesis $f^{-1}((y_1, u]) = \emptyset$.

□

B.2. Prueba del teorema 2.2.2

Teorema 2.2.2 (Propiedades del ordenamiento).

1. f^* es monótona creciente.
2. $s < f^*(\tau)$ si y solo si $T_f(s) < \tau$.
3. f^* es continua por izquierda.
4. f y f^* son equimedibles, es decir $T_f = T_{f^*}$.
5. Si T_f es discontinua en y_0 , entonces f^* es constante en el intervalo $(T_f(y_0^-), T_f(y_0))$ con valor y_0 .
6. $T_{T_f}(\tau) = f^*(\tau) - A + \mu(T_f^{-1}(\{\tau\}))$.
7. $f^*(\tau_0^+) - f^*(\tau_0) = \mu(T_f^{-1}(\{\tau_0\}))$.

Demostración.

1. Sean $0 \leq \tau \leq u$; note que

$$\{y \in [A, B] : T_f(y) \geq u\} \subseteq \{y \in [A, B] : T_f(y) \geq \tau\},$$

luego tomando ínfimo se sigue que $f^*(\tau) \leq f^*(u)$, y así f^* es monótona creciente.

2. En primer lugar suponga $s < f^*(\tau) = \inf \{y \in [A, B] : T_f(y) \geq \tau\}$, luego se tiene que $s \notin \{y \in [A, B] : T_f(y) \geq \tau\}$ y así $T_f(s) < \tau$. Para el recíproco suponga $T_f(s) < \tau$. Si por el contrario $s \geq f^*(\tau)$, como T_f es monótona y también $T_f(f^*(\tau)) \geq \tau$, se sigue entonces que

$$T_f(s) \geq T_f(f^*(\tau)) \geq \tau,$$

lo cual es una contradicción, luego $s < f^*(\tau)$.

3. Considere la sucesión $\{\tau_0 - \frac{1}{n}\}_{n \in \mathbb{N}}$ que converge a τ_0 por abajo. Como $\tau_0 - \frac{1}{n} < \tau_0$, entonces $f^*(\tau_0 - \frac{1}{n}) \leq f^*(\tau_0)$ para todo $n \in \mathbb{N}$, i.e. la sucesión es acotada. Más aún, es creciente pues $f^*(\tau_0 - \frac{1}{n+1}) \geq f^*(\tau_0 - \frac{1}{n})$. Con lo anterior se concluye que:

$$\lim_{n \rightarrow \infty} f^*\left(\tau_0 - \frac{1}{n}\right) = \sup \left\{ f^*\left(\tau_0 - \frac{1}{n}\right) : n \in \mathbb{N} \right\}.$$

Por simplicidad defina $D := \{f^*(\tau_0 - \frac{1}{n}) : n \in \mathbb{N}\}$. Bajo esta notación $f^*(\tau_0)$ es cota superior de D . Sea z una cota superior arbitraria de D , entonces $z \geq f^*(\tau_0 - \frac{1}{n})$ para todo $n \in \mathbb{N}$; luego

$$T_f(z) \geq T_f\left(f^*\left(\tau_0 - \frac{1}{n}\right)\right) \geq \tau_0 - \frac{1}{n}, \quad \forall n \in \mathbb{N},$$

tomando límite tenemos que $T_f(z) \geq \tau_0$, y utilizando el ítem (2) se obtiene $f^*(\tau_0) \leq z$. Se concluye que $f^*(\tau_0^-) = f^*(\tau_0)$ y así es continua por izquierda.

4. Por el ítem (2) se tiene:

$$\{\tau \in [0, \mu(I)] : f^*(\tau) \leq y\} = \{\tau \in [0, \mu(I)] : T_f(y) \geq \tau\} = [0, T_f(y)],$$

por lo tanto

$$T_{f^*}(y) = \mu(\{\tau \in [0, \mu(I)] : f^*(\tau) \leq y\}) = \mu([0, T_f(y)]) = T_f(y).$$

5. Como T_f es monótona creciente la imagen inversa $T_f^{-1}((T_f(y_0^-), T_f(y_0))) = \emptyset$. Sea $\tau \in (T_f(y_0^-), T_f(y_0))$. Se mostrará que

$$f^*(\tau) = \inf \{y \in [A, B] : T_f(y) \geq \tau\} = y_0.$$

Por simplicidad $D := \{y \in [A, B] : T_f(y) \geq \tau\}$. Observe que $y_0 \in D$, pues $\tau < T_f(y_0)$. Para ver que y_0 es cota inferior considere $z \in D$; entonces $T_f(z) \geq \tau > T_f(y_0^-)$. Como la imagen inversa $T_f^{-1}((T_f(y_0^-), T_f(y_0))) = \emptyset$ entonces $T_f(z) \notin (T_f(y_0^-), T_f(y_0))$, y así $T_f(z) \geq T_f(y_0)$ y por lo tanto $z \geq y_0$.

6. De nuevo por el ítem (2):

$$\begin{aligned} T_{T_f}(\tau) &= \mu(\{y \in [A, B]: T_f(y) \leq \tau\}) = \mu(\{y \in [A, B]: T_f(y) < \tau\}) + \mu(T_f^{-1}(\{\tau\})) \\ &= \mu(\{y \in [A, B]: y < f^*(\tau)\}) + \mu(T_f^{-1}(\{\tau\})) \\ &= f^*(\tau) - A + \mu(T_f^{-1}(\{\tau\})) \end{aligned}$$

7. Por simplicidad denote $j_{f^*}(\tau_0) := f^*(\tau_0^+) - f^*(\tau_0)$. Como f^* es monótona creciente, entonces el salto se puede expresar como:

$$f^*(\tau_0^+) - f^*(\tau_0) = \inf \{f^*(y) - f^*(x) : x < \tau_0 < y\}.$$

Utilizando T_{T_f} como en el ítem (6) tenemos:

$$j_{f^*}(\tau_0) = \inf \{\mu(T_f^{-1}([x, y))) : x < \tau_0 < y\}.$$

Defina $G := \{\mu(T_f^{-1}([x, y))) : x < \tau_0 < y\}$. Note que $T_f^{-1}(\{\tau_0\}) \subseteq T_f^{-1}([x, y])$ para todo $x < \tau_0 < y$, con lo cual $\mu(T_f^{-1}(\{\tau_0\})) \leq \mu(T_f^{-1}([x, y]))$, y así $\mu(T_f^{-1}(\{\tau_0\}))$ es una cota inferior en G . Ahora, si w es una cota inferior de G , entonces $w \leq \mu(T_f^{-1}([x, y]))$ para todo $x < \tau_0 < y$, luego expandiendo:

$$w \leq \mu(T_f^{-1}([x, \tau_0])) + \mu(T_f^{-1}([\tau_0, y]))$$

tomando límite cuando $x \rightarrow \tau_0$ y $y \rightarrow \tau_0$ tenemos que $w \leq \mu(T_f^{-1}(\{\tau_0\}))$; por lo tanto $j_{f^*}(\tau_0) = \mu(T_f^{-1}(\{\tau_0\}))$.

□

B.3. Prueba del teorema 2.2.3

Teorema 2.2.3. *El ordenamiento tiene la misma integral que la función original, esto es:*

$$\int_I f(x) \, dx = \int_0^{\mu(I)} f^*(\tau) \, d\tau$$

Demostración. Se mostrará el resultado para funciones positivas ya que $(f^+)^* = (f^*)^+$ y $(f^-)^* = (f^*)^-$, por lo que utilizando la definición de integral para funciones medibles arbitrarias el resultado se seguirá directamente.

Si g es positiva sobre I entonces

$$\begin{aligned} \int_I g(x) \, dx &= \int_I \int_0^{g(x)} 1 \, ds \, dx \\ &= \int_I \int_0^\infty \chi_{[0, g(x)]}(s) \, ds \, dx. \end{aligned}$$

Aplicando el teorema de Fubini:

$$\begin{aligned}
\int_I \int_0^\infty \chi_{[0, g(x)]}(s) \, ds \, dx &= \int_0^\infty \int_I \chi_{[0, g(x)]}(s) \, dx \, ds \\
&= \int_0^\infty \left(\int_I \chi_{I \cap \{\tau: g(\tau) \geq s\}}(x) \, dx \right) \, ds \\
&= \int_0^\infty \mu(\{\tau \in I: g(\tau) \geq s\}) \, ds \\
&= \int_0^\infty \mu([0, \mu(I)] \cap \{x \in [0, \infty): g^*(x) \geq s\}) \, ds \\
&= \int_0^\infty \chi_{[0, \mu(I)]}(\tau) \left(\int_0^\infty \chi_{\{x \in [0, \infty): g^*(x) \geq s\}}(\tau) \, ds \right) \, d\tau \\
&= \int_0^\infty \chi_{[0, \mu(I)]}(\tau) \left(\int_0^\infty \chi_{[0, g^*(\tau)]}(s) \, ds \right) \, d\tau \\
&= \int_0^\infty \chi_{[0, \mu(I)]}(\tau) \left(\int_0^{g^*(\tau)} 1 \, ds \right) \, d\tau \\
&= \int_0^\infty g^*(\tau) \chi_{[0, \mu(I)]}(\tau) \, d\tau \\
&= \int_0^{\mu(I)} g^*(\tau) \, d\tau
\end{aligned}$$

□

Bibliografía

- [1] J. W. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.
- [2] H. Longbotham and A. Bovik, “Theory of order statistic filters and their relationship to linear fir filters,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 2, pp. 275–287, 1989.
- [3] N. Gallagher and G. Wise, “A theoretical analysis of the properties of median filters,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1136–1141, 1981.
- [4] S. G. Tyan, *Median Filtering: Deterministic Properties*, pp. 197–217. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981.
- [5] A. Bovik, T. Huang, and D. Munson, “A generalization of median filtering using linear combinations of order statistics,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 6, pp. 1342–1350, 1983.
- [6] A. Restrepo and A. Bovik, “Adaptive trimmed mean filters for image restoration,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 8, pp. 1326–1337, 1988.
- [7] J. Ramírez-Angulo, A. Díaz-Sánchez, R. González-Carvajal, A. López-Martin, and J. M. Rocha-Pérez, “Modular discrete and cmos integrated implementations of high-speed analog rank-order filters,” *Circuits, Systems, and Signal Processing*, vol. 37, pp. 5637–5646, Dec 2018.
- [8] A. Wongjan, A. Julsereewong, and T. Junsing, “Analog median filtering circuit using cmos three-input max/min cell,” in *2018 3rd International Conference on Control and Robotics Engineering (ICCRE)*, pp. 155–159, 2018.
- [9] T.-T. Bui and T. Shibata, “A vlsi implementation of rank-order searching circuit employing a time-domain technique,” *Journal of Engineering*, vol. 2013, 01 2013.
- [10] P. Tymoshchuk, “Design of parallel rank-order filtering system based on neural circuits of discrete-time,” in *2021 IEEE 16th International Conference on the Experience of Designing and Application of CAD Systems (CADSM)*, pp. 16–20, 2021.

- [11] J. Poikonen and A. Paasio, "A ranked order filter implementation for parallel analog processing," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 5, pp. 974–987, 2004.
- [12] U. Cilingiroglu and L. Dake, "Rank-order filter design with a sampled-analog multiple-winners-take-all core," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 8, pp. 978–984, 2002.
- [13] J. Fitch, E. Coyle, and N. Gallagher, "The analog median filter," *IEEE Transactions on Circuits and Systems*, vol. 33, no. 1, pp. 94–102, 1986.
- [14] P. Ferreira, "Sorting continuous-time signals and the analog median filter," *IEEE Signal Processing Letters*, vol. 7, no. 10, pp. 281–283, 2000.
- [15] P. Ferreira, "Sorting continuous-time signals: analog median and median-type filters," *IEEE Transactions on Signal Processing*, vol. 49, no. 11, pp. 2734–2744, 2001.
- [16] H. Longbotham and A. Bovik, "Relating analog and digital order statistic filters," in *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, pp. 1526–1529 vol.3, 1988.
- [17] A. Kawala-Sterniuk, M. Podpora, M. Pelc, M. Blaszczyzyn, E. J. Gorzelanczyk, R. Martinek, and S. Ozana, "Comparison of smoothing filters in analysis of eeg data for the medical diagnostics purposes," *Sensors (Basel, Switzerland)*, vol. 20, p. 807, Feb 2020. 32024267[pmid].
- [18] Y. Tsutsui, S. Awamoto, K. Himuro, Y. Umezu, S. Baba, and M. Sasaki, "Characteristics of smoothing filters to achieve the guideline recommended positron emission tomography image without harmonization," *Asia Oceania journal of nuclear medicine & biology*, vol. 6, no. 1, pp. 15–23, 2018. 29333463[pmid].
- [19] Y. A. Kropotov, A. A. Belov, and A. Y. Proskuryakov, "Smoothing filtering of observed samples acoustic signals and interference in audio exchange telecommunications," in *2018 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, pp. 1–5, 2018.
- [20] M. Bottema, "Deterministic properties of analog median filters," *IEEE Transactions on Information Theory*, vol. 37, no. 6, pp. 1629–1640, 1991.
- [21] J. Lukeš and J. Malý, *Medidas e integrales*. Traducción y revisión Humberto Rafeiro, Andrés Vargas. - Primera edición. - Bogotá: Editorial Pontificia Universidad Javeriana, 2017.
- [22] H. Royden, *Real Analysis*. Collier Macmillan international editions, Macmillan, 1968.
- [23] S. Axler, *Measure, Integration & Real Analysis*. Springer Nature, 2020.
- [24] T. M. Apostol, *Mathematical analysis; 2nd ed.* Addison-Wesley series in mathematics, Reading, MA: Addison-Wesley, 1974.

- [25] R. Castillo and H. Rafeiro, *An Introductory Course in Lebesgue Spaces*. CMS Books in Mathematics, Springer International Publishing, 2016.
- [26] G. Hardy and J. Littlewood, “A maximal theorem with function-theoretic applications,” *Acta Mathematica*, vol. 54, no. none, pp. 81 – 116, 1930.
- [27] D. Díaz, “Analog order filters - thesis.” <https://github.com/ddiaz99/Analog-Order-Filters-Thesis>, 2022.
- [28] “Giss surface temperature analysis.” <https://data.giss.nasa.gov/gistemp/>.
- [29] “Beijing air pollution: Real-time air quality index.” <https://aqicn.org/city/beijing/>.