

**SEPARACIÓN DE FLUJO BASE EN LA CUENCA ALTA DEL RÍO DE ORO
UTILIZANDO ALGORITMOS NUMÉRICOS**

DIANA MARCELA MACHUCA GONZÁLEZ

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA CIVIL
BUCARAMANGA**

2012

**SEPARACIÓN DE FLUJO BASE EN LA CUENCA ALTA DEL RÍO DE ORO
UTILIZANDO ALGORITMOS NUMÉRICOS**

DIANA MARCELA MACHUCA GONZÁLEZ

**Trabajo de grado presentado como requisito para optar al título de
Ingeniero Civil**

Director

**SULLY GÓMEZ ISIDRO
Ingeniera Civil, Ph.D.**

Codirector

**JOSE ALFREDO RUEDA
Ingeniero Civil**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA CIVIL
BUCARAMANGA**

2012

*A Dios, por la fortaleza para afrontar las dificultades,
A mis padres, por su amor incondicional, constante apoyo y dedicación,
A mi familia por su aliento, a mis profesores por sus enseñanzas
y a mis amigos por su motivación.*

Diana Marcela Machuca González

AGRADECIMIENTOS

Especialmente a la I.C PhD Sully Gómez Isidro, por todo su apoyo, confianza y dedicación. También al IC Jose Alfredo Rueda y al IC Juan Carlos Sarmiento por los aportes en el proyecto.

CONTENIDO

INTRODUCCIÓN.....	14
CAPÍTULO 1	15
1. ESTACIÓN HIDROMÉTRICA AUTOMÁTICA DEL RÍO DE ORO	15
1.1 UBICACIÓN DE LA ESTACIÓN EL CONQUISTADOR	16
1.2 DATOS DE CAUDAL.....	18
1.2.1 OBTENCIÓN DE LOS DATOS DE CAUDAL:.....	18
1.2.2 CONVERSIÓN A CAUDALES DIARIOS:	18
1.2.3 ANÁLISIS DE LOS VALORES MÁXIMOS DE CAUDAL:.....	20
1.2.4 RECESIÓN EN EL RÍO DE ORO	22
CAPÍTULO 2	23
2. SEPARACIÓN DE FLUJO BASE	23
2.1 ESTUDIO DE LA RECESIÓN	23
2.1.1 INTERPRETACIÓN DE LAS CURVAS DE RECESIÓN Y SUS CONSTANTES	24
2.1.2.1 ANÁLISIS DE C	30
2.2 SEPARACIÓN DE FLUJO BASE CON MÉTODOS DE ALGORITMOS.....	31
2.3 ALGORITMO DE NATHAN Y MCMAHON	31
2.3.1 TÉCNICA DEL FILTRO DIGITAL RECURSIVO AUTOMATIZADO:.....	32
2.3.2 VALOR DEL PARÁMETRO DE FILTRO B:.....	33
2.3.3 PROCEDIMIENTO REALIZADO POR EL PROGRAMA BASEFLOW FILTER: 34	

<u>2.3.3.1</u>	<u>CURVA MAESTRA DE RECESIÓN AUTOMATIZADA</u>	<u>34</u>
<u>2.3.4</u>	<u>ANÁLISIS Y REPRESENTACIÓN GRÁFICA DE LOS PASOS DEL ALGORITMO.....</u>	<u>38</u>
<u>2.3.5</u>	<u>APLICACIÓN DEL ALGORITMO DE SEPARACIÓN DE NATHAN Y MCMAHON</u>	<u>45</u>
<u>2.3.5.1</u>	<u>PORCENTAJES DE FLUJOS BASE DADOS POR LOS PASOS DEL ALGORITMO:</u>	<u>45</u>
<u>2.3.5.2</u>	<u>COMPARACIÓN DE LOS PASOS DE SEPARACIÓN DEL MÉTODO DE NATHAN Y MCMAHON</u>	<u>46</u>
<u>2.3.5.3</u>	<u>FLUJO BASE PARA LA CUENCA ALTA DEL RÍO DE ORO:</u>	<u>47</u>
	<u>CAPITULO 4</u>	<u>48</u>
<u>3.</u>	<u>INTERFAZ GRÁFICA EN MATLAB CREADA PARA EL SOFTWARE BASEFLOW FILTER.....</u>	<u>48</u>
<u>3.1</u>	<u>INSTRUCCIONES DE USO DE LA INTERFAZ GRÁFICA (VER ANEXO A)</u>	<u>48</u>
<u>3.2</u>	<u>DESCRIPCIÓN</u>	<u>50</u>
<u>3.2.1</u>	<u>DESCRIPCIÓN PARA OPERATION1 DEL MENÚ OPERATION</u>	<u>50</u>
<u>3.2.2</u>	<u>DESCRIPCIÓN PARA OPERATION2 DEL MENÚ OPERATION</u>	<u>51</u>
	<u>CAPITULO 5</u>	<u>52</u>
	<u>CONCLUSIONES.....</u>	<u>52</u>
	<u>RECOMENDACIONES.....</u>	<u>53</u>
	<u>BIBLIOGRAFÍA.....</u>	<u>54</u>
	<u>ANEXOS</u>	<u>58</u>

LISTA DE TABLAS

TABLA 1. Parámetros de recesión para la cuenca alta del río de Oro obtenidos mediante el Método de la Curva Maestra. (<i>La autora</i>).....	28
TABLA 2. Parámetros de recesión para la cuenca alta del río de Oro obtenidos mediante el Método de Recesión Individual. (<i>La autora</i>).....	28
TABLA 3. Diagrama de flujo del algoritmo para automatizar la CMR [20].....	37

LISTA DE FIGURAS

FIGURA 1.	Localización estación El Conquistador sobre la CSRO [28].....	16
FIGURA 2.	Ruta de acceso a la estación El Conquistador [13].....	17
FIGURA 3.	Caudales diarios octubre de 2008 - marzo de 2011 (<i>La autora</i>)....	19
FIGURA 4.	Períodos en la estación el Conquistador [28].....	21
FIGURA 5.	Componentes del Hidrograma [3]	23
FIGURA 6.	Curva Maestra de Recesión El Conquistador 2008 -2011 (<i>La autora</i>) 25	
FIGURA 7.	CMR Recesiones Representativas. (<i>La autora</i>).....	28
FIGURA 8.	Separación de flujo base del caudal [20]	33
FIGURA 9.	% Caída de flujo base con los pasos del filtro [20].....	38
FIGURA 10.	Representación gráfica PASO 1 (hacia adelante). (<i>La autora</i>)	39
FIGURA 11.	Representación gráfica PASO 2 (hacia atrás). (<i>La autora</i>).....	40
FIGURA 12.	Representación gráfica PASOS 1 y 2. (<i>La autora</i>).....	41
FIGURA 13.	Representación gráfica PASO 3 (hacia adelante). (<i>La autora</i>)	42
FIGURA 14.	Representación gráfica PASOS 1, 2 Y 3. (<i>La autora</i>).....	43
FIGURA 15.	Representación gráfica de los PASOS en la Estación El Conquistador. (<i>La autora</i>).....	44

LISTA DE ANEXOS

ANEXO A.	MANEJO DEL SOFTWARE Y LA INTERFAZ GRAFICA	58
ANEXO B.	ESTACIÓN AUTOMÁTICA DE AFORO	67
ANEXO C.	CAUDALES MÁXIMOS Y MÍNIMOS	68
ANEXO D.	PARÁMETROS DE RECESIÓN POR RECESIÓN INDIVIDUAL.....	70
ANEXO E.	ENVOLVENTE MÍNIMA DE CAUDALES.....	71
ANEXO F.	CÓDIGO FUENTE DE LA INTERFAZ GRÁFICA.....	72

RESUMEN

TITULO: Separación de flujo base en la cuenca alta del río de oro utilizando algoritmos numéricos*

AUTOR(A): Diana Marcela Machuca González**

PALABRAS CLAVES: Cuenca Superior del Río Oro (C.S.R.O.); Separación de Flujo Base; Algoritmos; Recesión; Escorrentía

CONTENIDO:

La presente investigación muestra la aplicación de los algoritmos numéricos utilizados por el programa Baseflow filter como modelo de separación de flujo base en la estación El Conquistador, ubicada en la subcuenca del río de Oro, al sur oriente de la cuenca superior del río Lebrija.

La separación de flujo base se llevó a cabo utilizando datos de caudales desde el 2 de octubre de 2008 hasta el 31 de marzo de 2011, los cuales fueron registrados en proyectos de grado anteriores. Se estimó la componente de flujo base entre un 71 y 81 % del caudal total; para poder visualizar y determinar fácilmente las separaciones de flujo base, se desarrolló una interfaz gráfica que representa temas de interés asociados a la salida de datos del software Baseflow filter, tales como el hidrograma, la Curva Maestra de Recesión, entre otros.

Los pasos 1 y 2 de la técnica de separación de flujo base de Nathan and McMahon, presentaron valores que se encuentran en un rango aceptable a comparación de otros métodos, mientras que el paso 3 desenchaja, subestimando el flujo base. Para este modelo se estimó que el flujo base corresponde en promedio al 76% del caudal medido en la estación de El Conquistador, lo cual llama la atención sobre la importancia de los acuíferos de la cuenca en su interrelación con el río.

* Proyecto de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Civil.

Director: Sully Gómez Isidro, IC, Ph.D

Codirector: Jose Alfredo Rueda, IC

ABSTRACT

TITLE: Baseflow Separation in the upper basin of the Río de Oro using numerical algorithms*

AUTHOR: Diana Marcela Machuca González**

KEY WORDS: Upper Basin of the Río de Oro; Baseflow Separation; Algorithms; Recession; Runoff

DESCRIPTION:

This research shows the application of the numerical algorithms used by the Baseflow filter program as a baseflow separation model at the station El Conquistador, located in the Río de Oro subbasin, south east of the upper basin of the Río Lebrija.

The baseflow separation was performed using streamflow data from the October 2, 2008 until March 31, 2011, which were recorded in previous degree projects. It is estimated the baseflow component between 71 and 81% of the streamflow; for visualizing and easily determine the baseflow separations, developed a graphical interface that represents topics of interest associated with the output of the Baseflow filter software such as hydrograph, Master Recession Curve, among others.

The baseflow separation technique based numerical algorithms by Nathan and McMahon, showed values that are within an acceptable range compared with other methods. For this model base flow was estimated as average 76% of the streamflow measured at the station El Conquistador, wich draws attention to the importance of aquifers in the basin in its relationship with the river.

* Undergraduate Research Project

** Physical Mechanical Engineering Faculty. Civil Engineering School.

Director: Sully Gómez Isidro, CE, Ph.D

Codirector: Jose Alfredo Rueda, CE

INTRODUCCIÓN

La Cuenca Superior del Río de Oro (CSRO) es considerada una reserva hídrica para el Área Metropolitana de Bucaramanga y ha sido objeto de diferentes estudios sobre identificación y disponibilidad de fuentes de aguas subterráneas, lo que ha implicado el análisis de diversas variables siendo una de estas la interacción entre las corrientes superficiales y las aguas subterráneas por lo cual el presente trabajo se concentra en hallar tal relación mediante la separación del flujo base en la estación El Conquistador en la subcuenca del río de Oro aplicando los algoritmos de separación de flujo descritos por Nathan y McMahon utilizando datos de caudales diarios (2008-2011).

El informe describe en su primer capítulo, la estación utilizada y se indica la obtención, el procesamiento y análisis de la información recopilada; en el segundo capítulo, se analiza la separación de flujo base, la Curva Maestra de Recesión y la constante de recesión en la cuenca alta del río de Oro, asimismo, se explican los algoritmos numéricos utilizados en el software Baseflow filter y se muestra su aplicación y resultados; en el capítulo tres, se explica el desarrollo de la interfaz gráfica creada y desarrollada durante el presente proyecto de grado y por último, en el capítulo cuatro, se dan las conclusiones y recomendaciones correspondientes.

En los anexos se encuentran las instrucciones para el manejo de la interfaz gráfica, su respectivo código fuente, las características de la estación, las gráficas utilizadas para los análisis de recesión, entre otros.

CAPÍTULO 1

Para realizar la separación de flujo base en la cuenca alta del Río de Oro se requiere de información hidrológica de caudal en la cuenca sobre uno o más puntos de control. En este capítulo, se describirá la estación utilizada, la información recopilada, su obtención y respectivo análisis.

1. ESTACIÓN HIDROMÉTRICA AUTOMÁTICA DEL RÍO DE ORO

La estación hidrométrica automática del río de Oro fue instalada en el año 2006 como proyecto de grado de los estudiantes Díaz & Rincón [12], por medio del convenio entre la UIS-Grupo Predicción Hidroclimática (GPH) y la empresa prestadora de servicios del municipio de Piedecuesta, Santander - Piedecuestana de Servicios. La estación fue monitoreada durante el año 2007 en nuevo proyecto de grado a cargo de Hernández & Saltarín [17]. Durante el periodo comprendido entre agosto de 2007 y octubre de 2008 la estación no estuvo en funcionamiento. Después de octubre de 2008 y hasta enero de 2009 el funcionamiento de la estación estuvo a cargo del proyecto llevado a cabo por Flórez & Rueda [13]. Estos proyectos mencionados además de niveles de agua, estudiaron parámetros como la temperatura y presión atmosférica. Con la ayuda del ADV Flow Tracker se tomaron medidas de caudal y utilizando estos datos se creó la curva de calibración que relaciona la altura con el nivel del agua del río.

Desde febrero de 2009 hasta julio de 2010 la estación fue monitoreada por el proyecto que desarrollaron los estudiantes Aguilar & Caballero [2]; en este proyecto se le da continuación a los estudios de los parámetros mencionados en proyectos anteriores y se agrega un nuevo parámetro a medir que es la conductividad eléctrica; para medir este parámetro y mejorar los demás parámetros se instala en la estación un nuevo instrumento llamado CTD Diver. A partir de agosto de 2010 hasta febrero de 2011, la estación fue monitoreada por

Cervantes & Fajardo [8], quienes hicieron el análisis y correlación de los datos proporcionados por el CTD Diver, Baro y Diver, para identificar la variación temporal de propiedades físico-químicas como temperatura, presión, niveles del agua y electro-conductividad. Los datos de caudal obtenidos en el proyecto de Fajardo & Cervantes [8] se reunieron con los obtenidos por Flórez & Rueda [13] y fueron utilizados en el presente trabajo, con el fin de realizar y estudiar la separación de flujo base en la cuenca alta del río de Oro por medio de los algoritmos numéricos del programa Baseflow filter.

1.1 UBICACIÓN DE LA ESTACIÓN EL CONQUISTADOR

La estación se encuentra ubicada en la sub cuenca del río de Oro, al sur de la cuenca superior del río Lebrija. El contorno de la subcuenca es bastante irregular y su cauce principal se forma por la unión de los ríos Oro y Hato o Lato, los cuales nacen cerca al alto de Picacho.

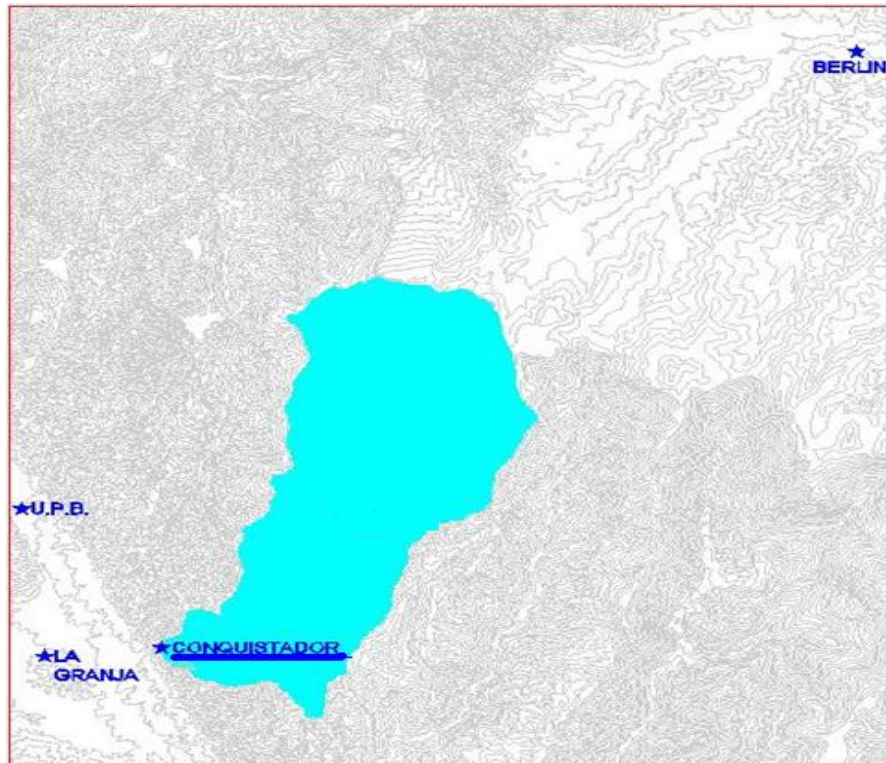


FIGURA 1. Localización estación El Conquistador sobre la CSRO [28]

La estación se encuentra en Piedecuesta a una altura de aproximadamente 1105 msnm y unas coordenadas de 6°59'41.36" N; 73°02'10.91" O [13], la sección de aforo se encuentra aproximadamente a 40m aguas debajo de allí. La pendiente del cauce de dicha zona es constante y el tramo es recto, por lo cual fue el sitio preciso para su ubicación.

Para acceder a la estación desde la ciudad de Bucaramanga se debe tomar la vía autopista de Piedecuesta hasta el puente vehicular de Piedecuesta y atravesar el barrio cabecera hasta el margen del río y de ahí en adelante se realiza un trayecto a pie de aproximadamente 25 minutos por la montaña en una trocha existente. La estación hidrométrica automática del río de Oro está capacitada para el registro continuo de parámetros hidrológicos del río tales como la temperatura, el nivel de agua y la presión atmosférica [13].



FIGURA 2. Ruta de acceso a la estación El Conquistador [13]

1.2 DATOS DE CAUDAL

1.2.1 Obtención de los datos de caudal:

Los datos de caudal usados en la presente investigación fueron calculados con los datos de niveles de agua medidos en la estación El Conquistador durante el período 02 de agosto de 2008 hasta el 31 de marzo de 2011 y por medio de la ecuación obtenida de la curva de calibración realizada por Flórez & Rueda [13] y mejorada después por Aguilar & Caballero [2] y Fajardo & Cervantes [12] quienes realizaron nuevos aforos:

$$Q = 1.056E^{-4}H^{2.519} \quad \text{Ecuación final de calibración Conquistador}$$

1.2.2 Conversión a caudales diarios:

Debido a que la estación de aforo El Conquistador corresponde a una estación automática con una configuración de toma de datos cada 3 minutos, los datos de caudal calculados, por tanto, en intervalos de tiempo de 3 minutos fueron promediados en caudales diarios para poder realizar el archivo de entrada al software Baseflow filter. Asimismo, se utilizó la información de caudales medios diarios ya que se requería de una transformación de la información a un intervalo común y que generara la mínima variación a los datos.

A continuación se muestra el hidrograma con los caudales medios diarios a partir del 02 de octubre de 2008 hasta el 31 de marzo de 2011 de la estación automática del río de Oro:

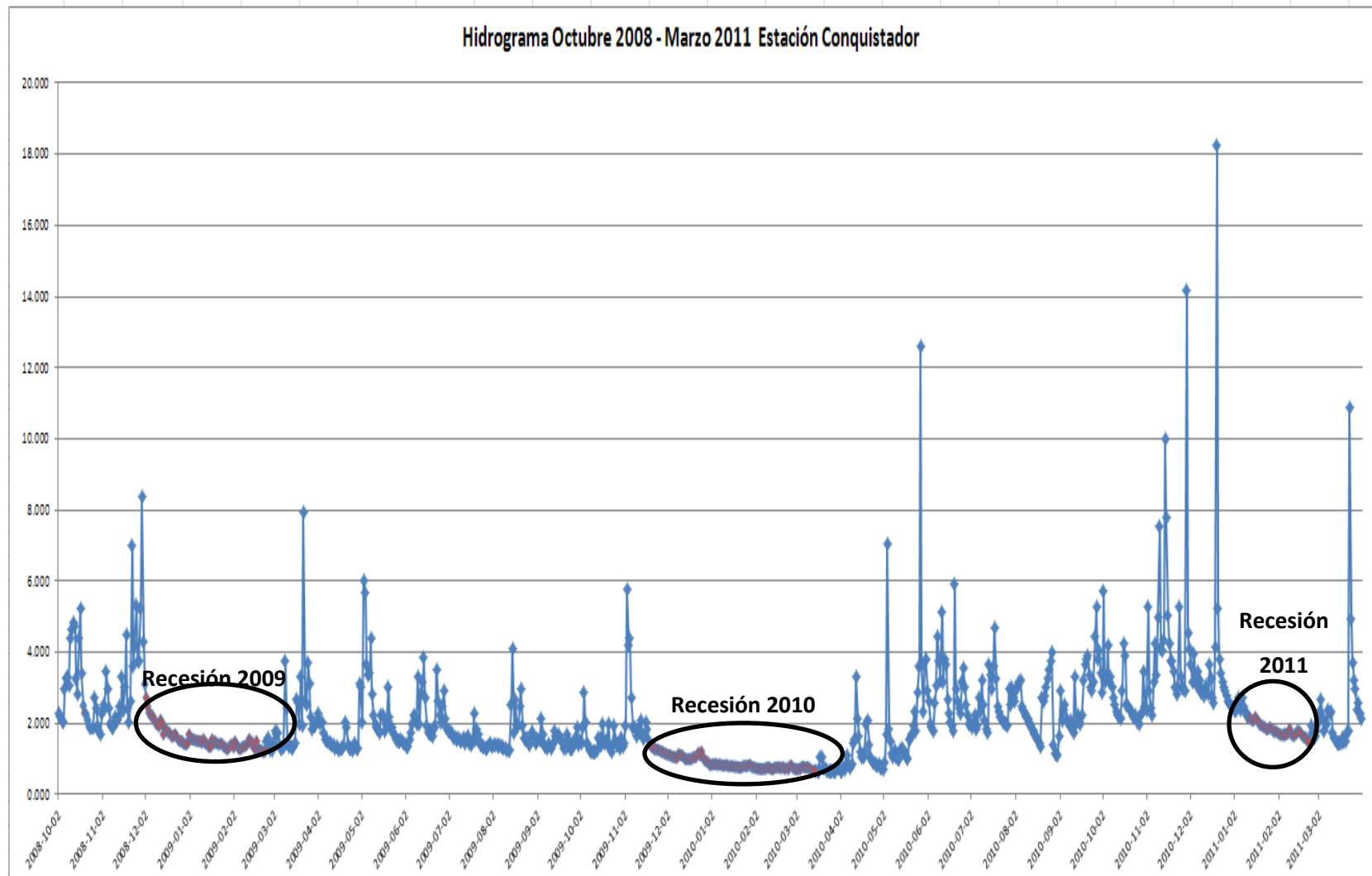


FIGURA 3. Caudales diarios octubre de 2008 - marzo de 2011 (La autora)

1.2.3 Análisis de los valores máximos de caudal:

En el anexo C se muestran los valores máximos y mínimos de caudal para cada mes del período estudiado.

El caudal alcanzó sus valores máximos el 23 de Marzo de 2011 entre las 7 y las 9 de la noche en el transcurso de una avalancha, en donde según la curva de calibración se llegó a un caudal de 103,6 m³/seg. Asimismo, los mayores valores de caudal medio diario se registraron el 27 de mayo, el 29 de noviembre y el 20 de diciembre de 2010 con un caudal de 12.618, 14.171 y 18.25 m³/seg, respectivamente; el promedio de caudal del 23 de Marzo de 2011 fue de 10.906 m³/seg.

También, se puede observar una época de lluvias bastante fuerte en los meses de octubre y noviembre de 2008 y 2010; períodos en donde se puede apreciar la respuesta rápida del río debido al aumento del nivel en muy poco tiempo. En el resto de tiempo hubo lluvias de menor importancia que no generaron caudales muy altos.

Las secciones en las que se interrumpen los registros son el tiempo en que los operadores de la estación bajan los datos acumulados en la memoria interna de los equipos, sin embargo, debido a la necesidad de un archivo completo de datos se ajustaron algunos valores para así dar continuidad a la información.

En el año 2009 se puede diferenciar claramente una temporada lluviosa que se extiende desde el 16 de abril al 3 de noviembre y una temporada seca desde el 18 de noviembre al 25 de abril del 2010 como se aprecia en la figura:

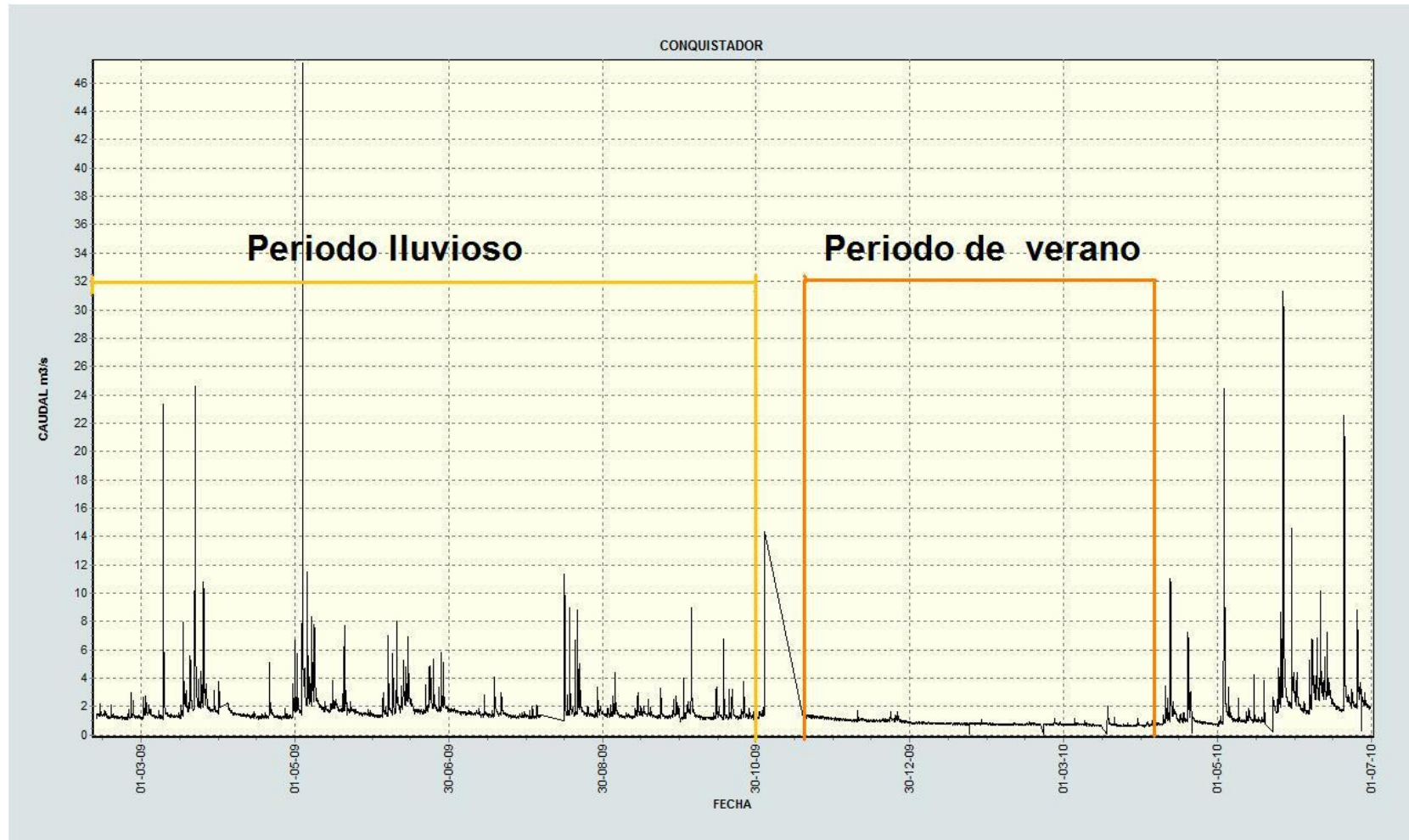


FIGURA 4. Períodos en la estación el Conquistador [28].

1.2.4 RECESIÓN EN EL RÍO DE ORO

La recesión en el río de oro es lenta, como muestra la Figura 3. Para analizar la curva de recesión del río se aprovechó las tres épocas de recesión más largas del período de tiempo estudiado:

- El primer período se presentó desde el 15 de diciembre de 2008 al 21 de febrero de 2009 para un total de 65 días aproximadamente. En los meses de diciembre de 2008 y enero de 2009 se observa una época de no lluvias bastante amplia de 45 días, con tan solo pequeñas precipitaciones el 31 de diciembre de 2008 y el 11 de enero de 2009, las cuales se evidencian en el registro de los caudales, es decir, después de la temporada de lluvias los niveles alcanzaron un valor mínimo de 41.7 cm el día 13 de enero de 2009.

- El segundo período es el más larga y se registra claramente durante el tiempo comprendido entre los últimos meses del año 2009 y en los primeros meses del año 2010, más exactamente del 16 de noviembre de 2009 hasta el 16 de marzo de 2010 para un total de 120 días aproximadamente. Los registros más bajos de caudal se obtuvieron en el mes de Mayo de 2009 llegándose a medir menos de 0,5 m³/seg; los niveles mínimos de agua han sido registrados en esta recesión, alcanzando niveles de hasta 35 centímetros, con presencia de eventos de lluvia los niveles del agua varían de acuerdo con la intensidad del evento pudiendo llegar a un nivel hasta de 239 centímetros según los registros entre Diciembre del 2009 y Abril de 2010.

- El tercer período de recesión es la más corta, del 08 de enero al 01 de febrero de del 2011, es decir, de 24 días aproximadamente; es una época de poca lluvia como podemos notar en la Figura 8 en la cual están todos los caudales medios diarios registrados.

CAPÍTULO 2

2. SEPARACIÓN DE FLUJO BASE

El caudal base se define como el caudal que se presenta antes, durante y después de la adición del volumen precipitado, esto considera al agua que se introduce a la corriente por precipitaciones anteriores o bien por la infiltración de agua desde los acuíferos a la corriente (flujo subterráneo), entonces el “flujo base es considerado como el resultado de la descarga del agua subterránea en la corriente del río, mientras la escorrentía directa es considerada como el resultado del flujo superficial o cercano a la superficie” [9].

2.1 ESTUDIO DE LA RECESIÓN

La separación del flujo base de una corriente hidrográfica inicia con la identificación de los puntos en los cuales el escurrimiento directo comienza y finaliza. El punto de inicio del escurrimiento directo (escurrimiento directo es el agua que proviene del evento de lluvia, durante o cuando ya ha transcurrido algún tiempo después de que acabo de llover) es fácilmente reconocido puesto que es el tiempo en que el flujo empieza a incrementarse, mientras los puntos de finalización o de terminación del escurrimiento directo se suele tomar como el punto de mayor curvatura de la curva de recesión [3].

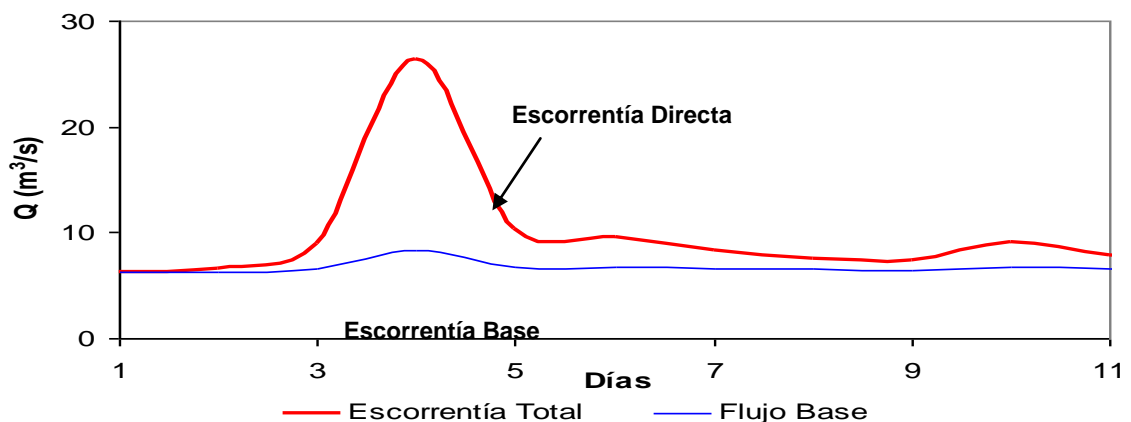


FIGURA 5. Componentes del Hidrograma [3]

Es muy difícil conocer con precisión el punto final del escurrimiento directo de un hidrograma, esto hace que la separación del flujo base del directo no sea una labor sencilla, con el fin de determinar el escurrimiento directo se analizan varios hidrogramas y se seleccionan aquellos tramos en donde solo existe escurrimiento base, estos tramos seleccionados se dibujan posteriormente en papel semilogarítmico de tal forma que sus extremos inferiores sean tangentes a una línea, esta línea resultante se conoce como Curva Maestra de Recesión.

El punto final del escurrimiento directo del hidrograma se localiza suponiendo la curva maestra de recesión dibujada en papel aritmético a la misma escala que el hidrograma con la curva de recesión del hidrograma. El punto final del escurrimiento se encuentra donde ambas líneas se separa. La pendiente de la línea resultante de la curva maestra de recesión corresponde al valor de la constante de recesión, que para el caso de este estudio es de aproximadamente 0.025 (figura 9).

2.1.1 Interpretación de las Curvas de Recesión y sus constantes

La precipitación interrumpe frecuentemente el periodo de recesión, en consecuencia cada serie de descargas produce una serie de segmentos de recesión de duraciones variables. Un segmento de recesión es un tramo de la curva de recesión, el cual está determinado por un caudal inicial y una longitud de recesión. Se han desarrollado varios métodos para construir una curva de recesión maestra a partir de las recesiones más cortas. Un gran problema es la alta variabilidad encontrada en el comportamiento de la recesión de segmentos individuales. Los segmentos representan diferentes etapas en el proceso de salida del flujo y un efecto físico a corto plazo o variación estacional en el comportamiento de la recesión se adhiere al problema de derivar una curva de recesión característica.

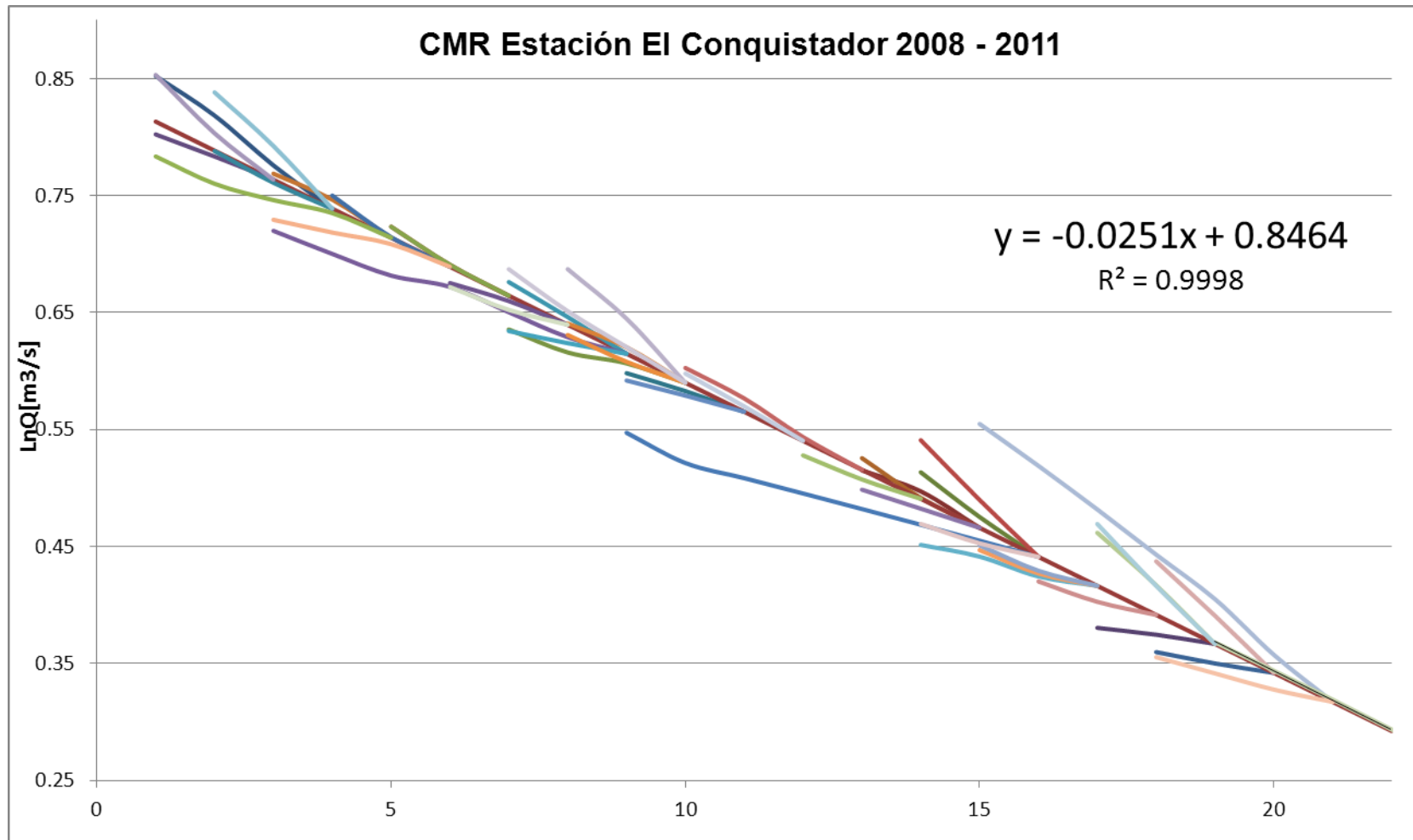


FIGURA 6. Curva Maestra de Recesión El Conquistador 2008 -2011 (*La autora*)

2.1.1.1 Obtención de los Parámetros de Recesión

Una vez separados los segmentos de recesión se procede a calcular los parámetros caudal inicial y constante de recesión para la cuenca alta del río de Oro; antes de desarrollar el método de la Curva Maestra es necesario descartar la serie de caudales con pendientes muy altas y las que se presenten datos de caudales consecutivos muy repetidos.

2.1.1.2 Curva Maestra de Recesión y constantes

Debido a las diferencias que presenta cada uno de los segmentos, se puede decir que el método realiza un promedio de las características de recesión de todos los segmentos, ya que se trabajan con períodos de diferente longitud y constante de recesión.

Las curvas de recesión maestra ajustadas presentan un buen factor de correlación mayor a 0.98, en la Figura 7 se muestran las curvas más representativas y la ecuación que las describe.

Cabe mencionar que se puede interpretar de diferentes formas la constante de recesión, ésta es la pendiente con que decrece la curva de recesión, en este caso estaría expresada en unidades (T^{-1}); pero otros autores [2] han encontrado que el inverso de la constante de recesión ($1/\alpha$) en unidades (T), llamado para este proyecto C (días) puede ser interpretado como el tiempo promedio de permanencia de una gota que ingresa al acuífero.

La estación hidrométrica El Conquistador presenta coeficientes de recesión C de 34, 40 y 49 días, más bajos si se comparan con los obtenidos de otras estaciones cercanas. Los valores de C bajos, como los presentes en esta estación, significan una descarga más rápida del acuífero y los C altos una mayor regulación en los aportes que los acuíferos hacen a las corrientes de agua.

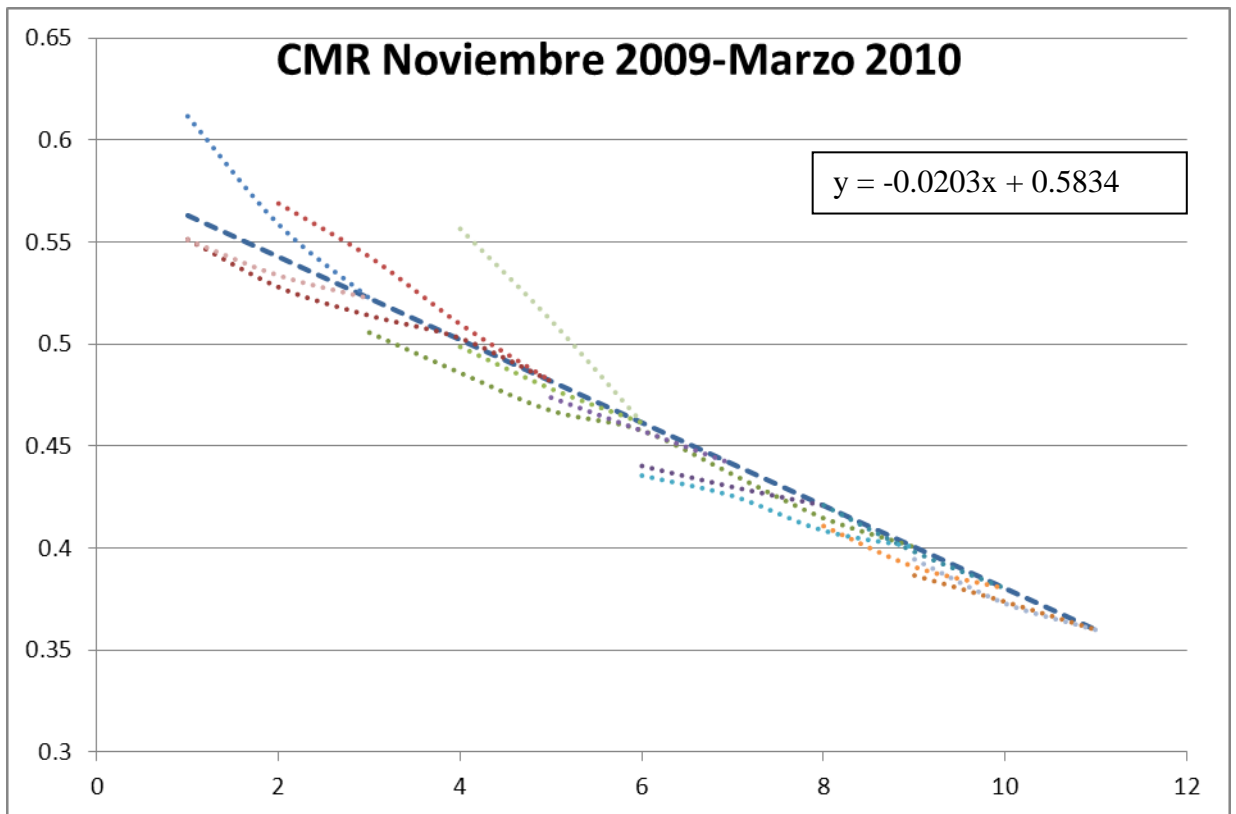
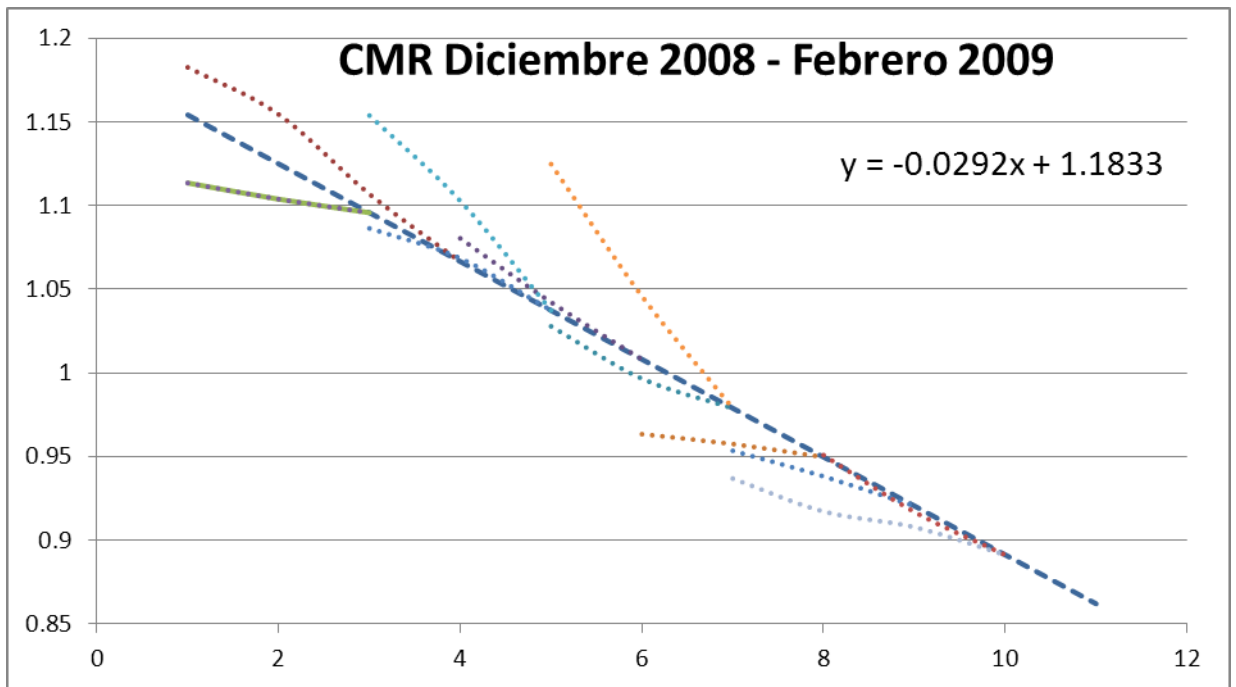


FIGURA 7. CMR Recesiones Representativas. (La autora)

RECESIÓN MAESTRA				
ESTACIÓN	α (días ⁻¹)	C días	Qo[m3/s]	R
EL CONQUISTADOR	-0.02921548	34	1.1833	0.99914964
	-0.0203	49	0.5834	0.9980481
	-0.0252	40	0.8386	0.99969995

Tabla 1. Parámetros de recesión para la cuenca alta del río de Oro obtenidos mediante el Método de la Curva Maestra. (La autora)

2.1.1.3 Recesión individual

Un primer análisis de las gráficas de los segmentos permite establecer que cuando estos se presentan en periodos secos más extensos tienen caudales bajos y pendientes suaves, estas características se presentan con más frecuencia en los años Niño.

Los segmentos de flujo base con longitudes cortas (días) presentan menores coeficientes de recesión, mientras que los que para valores altos la longitud de recesión aumenta. En general, el comportamiento de los segmentos es más paralelo cuando tienen menor longitud. En el anexo E se muestra la tabla de resultados obtenidos por este método.

RECESIÓN INDIVIDUAL			
α (días ⁻¹)	C días	Qo[m3/s]	R
-0.025417219	51	0.82800732	0.97046105

Tabla 2. Parámetros de recesión para la cuenca alta del río de Oro obtenidos mediante el Método de Recesión Individual. (La autora)

2.1.2 Comparación de las Constantes de Recesión con algunas características físicas de la cuenca

Repetidamente se ha dicho que el régimen de un río depende de un variado conjunto de factores que, a su vez, no son independientes entre sí. Entre estos factores cabe numerar las precipitaciones, la temperatura, el relieve, la vegetación, el tipo de suelo y la geología. En cuencas con dimensiones moderadas o pequeñas menores de 2000 Km^2 , la geología puede jugar un papel importante al hacer que haya trasvases subterráneos entre cuencas próximas pues muchas veces no coinciden las divisiones de las aguas superficiales y las de las aguas subterráneas [2].

Se intentara hacer un análisis cualitativo entre el coeficiente de agotamiento y algunas características de la cuenca subterránea; según Diosdado Pérez en 1995 para una cuenca subterránea cuya alimentación se deba a las precipitaciones, sin que vierta agua hacia otras cuencas o las reciba de estas, el coeficiente de recesión α depende de varios factores:

- Es tanto más pequeño, cuanto mayor sea el volumen de agua subterránea almacenada en el acuífero.
- Es función inversa de la porosidad eficaz o del coeficiente de almacenamiento.
- Es inversamente proporcional a la conductividad hidráulica.

Un mayor volumen significa una mayor regulación en la capacidad de almacenamiento representado en tasas de recesión más lentas. Un mayor coeficiente de almacenamiento significa mejores condiciones para aportar agua hacia las corrientes, o mayor facilidad para el drenaje. Los coeficientes de almacenamiento menores están asociados a los depósitos cuaternarios ya que estos se conforman principalmente de combinaciones de arenas limos y arcillas.

En cuanto a la conductividad hidráulica la relación nos permite ver que es mayor en formaciones de tipo ígneo metamórfico, en estos casos la conductividad se puede ver favorecida por las condiciones existentes en las fallas, fisuras y diaclasas.

2.1.2.1 Análisis de C

Al realizar la curva maestra de recesión de los datos de la estación El Conquistador, debido a que se presentó inversos de coeficientes de recesión altos, (C entre 39 y 51 días), con C calculado como $1/\alpha$, significa que en esta zona, la regulación del aporte de los acuíferos a las corrientes es más eficiente. Las variaciones de los coeficientes de recesión para esta subcuenca se deben a: la precisión de los caudales medidos, lo cual depende del sistema usado en la medición de caudales; en el proceso de separación de flujo base se pudo haber incluido parte de la escorrentía subsuperficial; también se deben a las condiciones que antecedieron al inicio de la recesión como por ejemplo, la humedad del suelo.

Durante el desarrollo del proyecto se analizó que la cuenca alta del Río de Oro presenta al parecer buenas condiciones para el almacenamiento del recurso subterráneo; alta pluviosidad, estructura geológica favorable a la infiltración debido a la permeabilidad secundaria que le confiere el alto fracturamiento de las rocas ígneo metamórficas. Además se obtuvo valores altos de C (días), lo que indica que regula eficientemente el volumen almacenado.

Una vez identificado el punto final del escurrimiento directo resta trazar la línea de separación entre el flujo base y el directo, para este propósito se analizará el método de Nathan y McMahon utilizado por el software Baseflow filter y que se explica a continuación.

2.2 SEPARACIÓN DE FLUJO BASE CON MÉTODOS DE ALGORITMOS

La separación de flujo base en registros históricos de caudales requieren el desarrollo de algoritmos que permitan llevar a cabo este cálculo en el largo plazo.

Estos métodos permiten dar una estimación mejor del flujo base ya que tienen en cuenta el comportamiento del flujo base similar al hidrograma y no un comportamiento lineal como los métodos gráficos y de intervalo; además permiten hacer la separación para cualquier tipo de evento.

2.3 ALGORITMO DE NATHAN Y MCMAHON

El software Baseflow filter está basado en una técnica automatizada de separación de flujo base que utiliza un filtro digital y ha sido probada contra otras técnicas automatizadas y métodos de separación manuales; también, frente a los flujos base medidos en seis cuencas hidrográficas, obtuvo un coeficiente de determinación general mensual de 0,86 [6].

El objetivo del software es la determinación de las características del flujo base a partir de bases de datos de caudales diarios. El resultado de dicho análisis será el porcentaje del flujo base en función del caudal y la constante de recesión de la CMR. Estos parámetros han demostrado ser útiles en la evaluación de las propiedades de acuíferos o para la entrada a los modelos hidrológicos [6].

Una excelente revisión de la automatización del flujo base y análisis de la recesión está dada por Nathan y McMahon (1990), quienes analizaron una técnica de separación de flujo base basada en un filtro digital recursivo que se encontró que era un método rápido y objetivo de la separación continua del flujo base.

2.3.1 Técnica del Filtro Digital Recursivo Automatizado:

Este procedimiento de separación de flujo base descrito por Nathan y McMahon (1990) fue originalmente utilizada en el análisis de señales y procesamiento (Lyne y Hollick, 1979, quienes aparentan haber sido los primeros en sugerir el uso de un filtro digital).

Aunque la técnica no tiene una base física real, es importante destacar que, sin embargo, la técnica proporciona una estimación objetiva y repetible de un índice de flujo base que es fácilmente automatizado. El filtrado de la escorrentía superficial (señales de alta frecuencia) a partir de flujo base (señales de baja frecuencia) es análoga a la filtración de señales de alta frecuencia en el análisis de señal y el procesamiento.

La ecuación del filtro es:

$$q_t = \beta q_{t-1} + \frac{1 + \beta}{2} (Q_t - Q_{t-1})$$

Donde q_t es la escorrentía superficial filtrada (respuesta rápida) en el paso de tiempo t (un día), Q_t es el caudal original y β es el parámetro de filtro (0.925).

El flujo base, b_t , se calcula con la ecuación:

$$b_t = Q_t - q_t$$

El valor de β igual a 0,925 se determinó por Nathan y McMahon (1990) y Arnold et al. (1995) para obtener resultados reales en comparación con las técnicas de separación manual. A continuación se explica su significado.

2.3.2 Valor del parámetro de filtro β :

Una inspección visual de varios conjuntos de datos indicó que el valor del parámetro de filtro que produjo la separación de flujo base más aceptable estaba en el intervalo de **0.9 a 0.95**.

En consecuencia, se decidió aplicar el filtro a los datos de todo el conjunto con los tres valores de los parámetros de 0.9, 0.925 y 0.95, y el valor de parámetro óptimo fue elegido entonces como 0.925 después de comparar los tres conjuntos de resultados. Los tres diferentes parámetros de filtro también fueron evaluados para su uso en la regionalización. Los resultados igualmente indicaron que de los tres valores de los parámetros de filtro a prueba un valor igual a 0,925 parece el más adecuado. En consecuencia, la técnica de un filtro digital con un parámetro de filtro fijado en **0.925** fue adoptada como proporcionar la mejor estimación del índice de flujo base. Así, se encontró que el uso de un sencillo filtro digital era un método rápido y objetivo de la separación continua de flujo base. Los resultados obtenidos usando esta técnica dieron respuestas similares a los obtenidos utilizando una técnica tradicional gráfica para un intervalo de variabilidad de flujo y tamaños de captación.

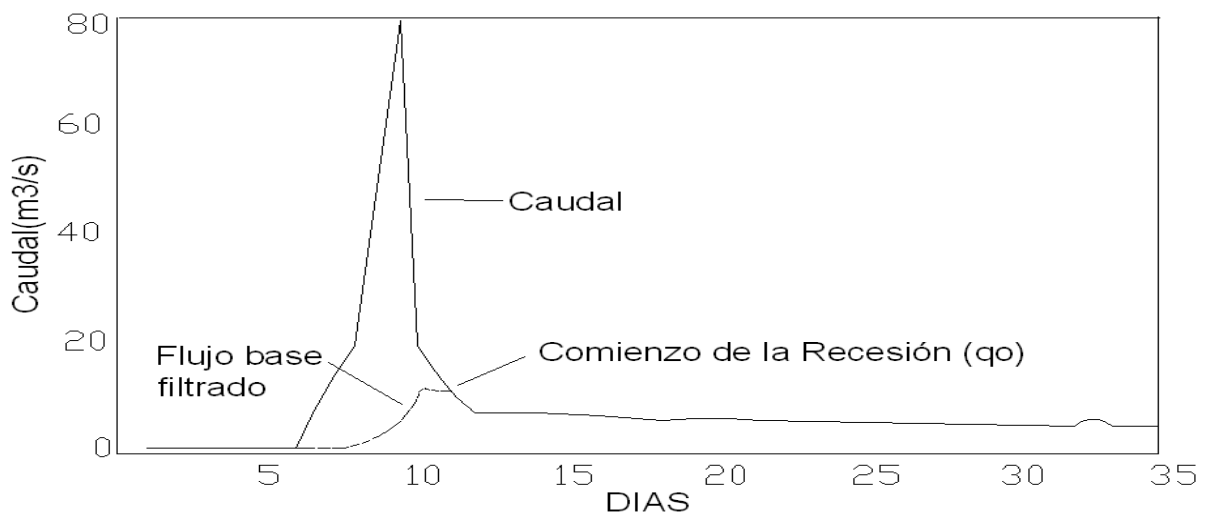


FIGURA 8. Separación de flujo base del caudal [20]

2.3.3 Procedimiento realizado por el programa Baseflow filter:

El programa empieza por definir las variables que va a utilizar con sus respectivas especificaciones de acuerdo al tipo de dato y crea vectores a los que se les asignan el valor de las variables que lee de los archivos de entrada .prn y .out, haciendo el manejo de errores en caso de alguna incoherencia en los datos. Inicialmente crea los vectores año, mes y día, dividiendo la fecha correspondiente a cada caudal y haciendo un procedimiento cuando el año es bisiesto para mayor exactitud.

Luego, declara el parámetro de filtro $\beta=0.925$ y empieza a realizar los cálculos correspondientes a los tres pasos de separación de flujo base, los cuales se explican gráficamente en el numeral **3.3.4**. Seguidamente, hace la sumatoria de los caudales y de los valores de flujo base hallados para cada paso, así, calcula las fracciones de flujo base resultantes del paso 1, 2 y 3, por ejemplo, fr1 es igual a la sumatoria del flujo base calculado por el paso 1 dividido sobre la sumatoria del caudal total.

Por otra parte, el software hace un procedimiento semiautomático que selecciona el segmento de recesión y lo dibuja en orden descendente en escala semilogarítmica. Este procedimiento ha hecho que el método Matching Strip consuma menos tiempo, sin embargo es aun subjetivo. El procedimiento se describe a continuación:

2.3.3.1 Curva Maestra de Recesión Automatizada

El procedimiento para automatizar totalmente un método para predecir la pendiente de las recesiones se describe como sigue: los datos de caudal diario del archivo de entrada se pasan a través del programa de filtro. Por lo general, la

recesión se calcula a partir del punto en el hidrograma, donde se supone que todo el flujo de la superficie ha dejado de existir o por convención [13]:

$$D = A^{0.2}$$

donde D =días después del pico del hidrograma donde la escorrentía superficial cesa.

El programa calcula y selecciona las constantes de recesión del caudal (alpha) para cada uno de los tramos de recesión, así: primero, el filtro utiliza el punto donde el vaor de flujo base del primer paso se reúne con el registro de caudal y calcula todas las pendientes existentes ajustando que el número de días del tramo esté entre el rango de ndmin y ndmax escogido por el usuario en el archivo de entrada .lst.

De esta forma, se llena el vector alpha (i) = (Ln (Q/Qo))/nd, donde nd generalmente es tomado de 10 días, ya que Nathan y MacMahon hicieron énfasis en la importancia de seleccionar un intervalo de tiempo tan largo como fuese posible al usar el método de correlación, debido al elemento subjetivo involucrado al ajustar una curva a los puntos. Ellos acentuaron además que un pequeño rango en la pendiente representa una porción significativa del rango de caudal, por lo tanto la incertidumbre involucrada al aplicar este método puede ser considerable.

Luego, escoge sólo las pendientes alpha que sean positivas dado que la resta de logaritmos de caudal para hacer la variación en Y se hizo inicial – final, separando así, los tramos de recesión significativos; también escoge sólo los tramos que pertenecen a enero, febrero, noviembre y diciembre debido a que son los meses donde la evapotranspiración es baja y es mejor su estimación. Estos segmentos son entonces ordenados por rango de menor a mayor. El registro es entonces buscado para determinar si hay alguna falta de coincidencia entre estos segmentos de descarga ordenados derivados del registro de caudal y los pasos del filtro.

A estos tramos seleccionados, se les calcula las coordenadas X y Y para realizar el análisis de regresión de las pendientes alpha, para esto, crea una matriz en donde guarda todos los valores de caudal que conforman cada tramo de recesión contabilizando el número total de recesiones que encontró, asimismo, guarda en un vector, el índice de recesión de cada tramo. La regresión lineal la realiza con el punto medio de cada tramo, encontrando así, el intercepto Y y la pendiente de la Curva Maestra de Recesión, es decir la constante de recesión α representativa.

Ahora, el programa busca el valor del punto más bajo de cada tramo de recesión, es decir, el valor de flujo más pequeño y crea la secuencia de dónde ese valor cae sobre el eje X, reorganizando la matriz anteriormente hecha; cuando se encuentra un vacío, el programa proyecta una regresión lineal desde los últimos puntos de flujo existentes en el segmento para establecer el rango numérico del espacio vacío. A través de este medio, los segmentos de caudal del filtro pueden ser ajustados a su posición adecuada en la Curva Maestra de Recesión (CMR) para su posterior análisis. Cuando existen suficientes datos anuales, las diferencias son pequeñas y las proyecciones necesarias son por lo general no más de 10 días.

Si no hay vacíos en las series de caudal construidas o después de que se han ajustado, los registros se analizan para calcular la pendiente (factor Alpha) o constante de recesión del caudal. Ya que hay a menudo una dispersión considerable en los flujos de recesión, se creó una técnica para recoger la menor pendiente de la curva. Asimismo, se halla el índice de recesión k, nombrado como Baseflow days en el archivo de salida .dat, que representa el tiempo por ciclo logarítmico de recesión de caudal, o en otras palabras, el tiempo en días que tarda el caudal en reducirse en un factor de 0.1. A continuación, se representa la automatización de la Curva Maestra de Recesión realizada por Baseflow filter a través de un diagrama de flujo.

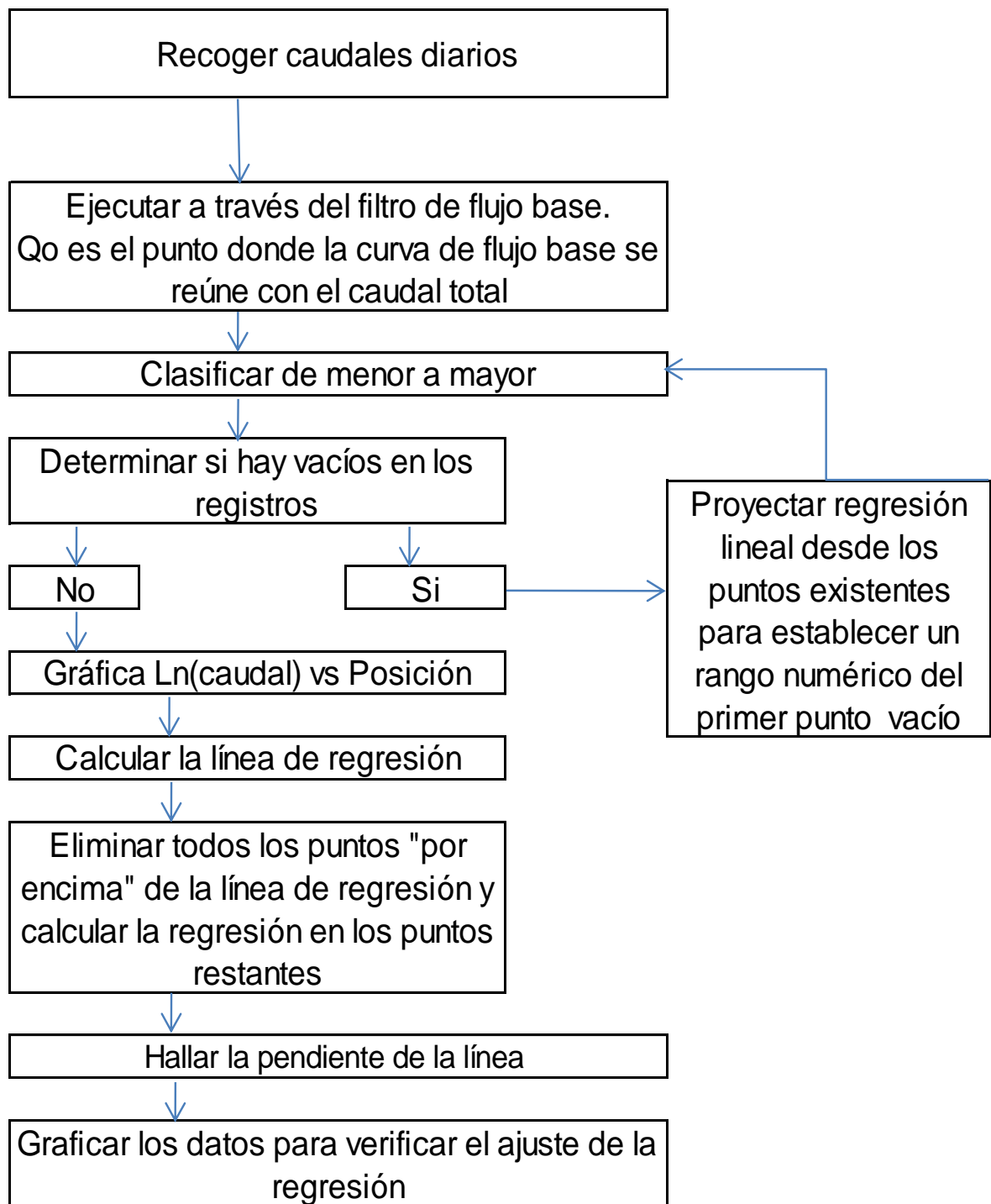


TABLA 3. Diagrama de flujo del algoritmo para automatizar la CMR [20]

2.3.4 ANÁLISIS Y REPRESENTACIÓN GRÁFICA DE LOS PASOS DEL ALGORITMO

El filtro es pasado sobre los datos de caudal tres veces (hacia delante, hacia atrás y hacia adelante), en cada pasada se calcula un valor de flujo base que se representan en el archivo de salida como paso 1, paso 2 y paso 3 .

En general, cada pasada se traducirá en menos flujo base como un porcentaje del flujo total, debido a que cada paso se basa en los resultados del anterior, sólo el paso 1 trabaja con los valores de caudales; la reducción promedio en el flujo base para los pasos dos y tres a partir del uno se muestra a continuación:

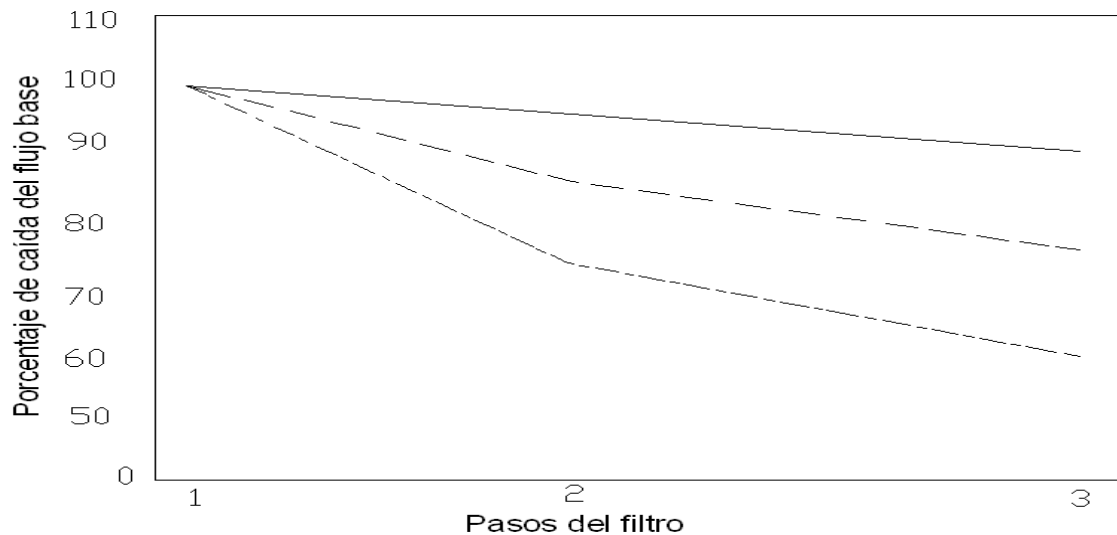


FIGURA 9. % Caída de flujo base con los pasos del filtro [20]

El flujo base se reduce aproximadamente un 17 por ciento en el segundo paso y un 10 por ciento adicional en el tercer paso. Esta opción da al usuario más flexibilidad para ajustar la separación a las condiciones más aproximadas del lugar. En general, el paso inverso se hizo para anular cualquier distorsión de fase de los datos debido al paso por delante del filtro. Se puede observar que el tercer paso recorrido comparado con los anteriores tiende a subestimar el caudal base, esto se debe al modelo matemático y físico con que se planteó este método.

A continuación se representará las operaciones realizadas por cada paso:

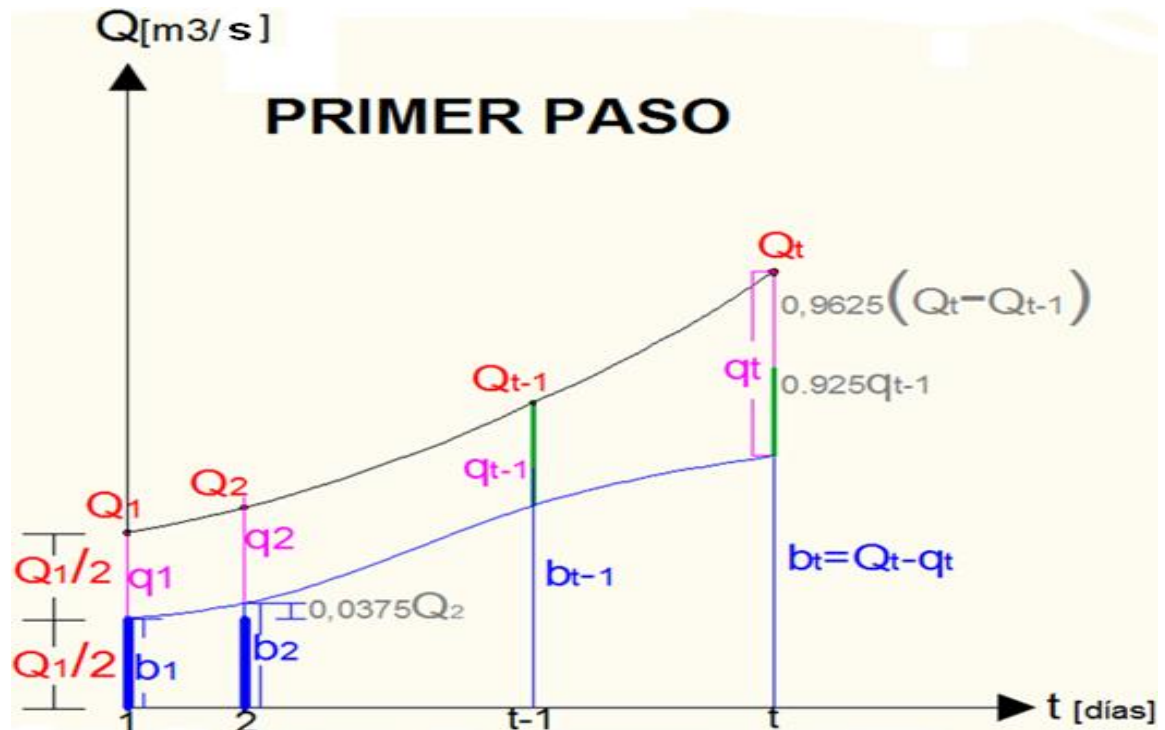


FIGURA 10. Representación gráfica PASO 1 (hacia adelante). (La autora)

2.3.4.1 PASO 1 (hacia adelante)

Para el primer día: $q_1 = Q_1 * 0.5$ y $b(1,1) = Q_1 - q_1$

Este valor corresponde también al flujo base calculado para los pasos 2 y 3, es decir, los tres pasos empiezan con el mismo valor de flujo base que corresponde a la mitad del caudal en el primer día, así:

$$b(2,1) = b(1,1) = Q_1/2$$

$$b(3,1) = b(1,1) = Q_1/2$$

En general, el paso 1 calcula el flujo base así:

$$\text{Volumen escurrentía } q_t = 0.925 * q_{t-1} + 0.9625 * (Q_t - Q_{t-1})$$

$$\text{Flujo base } b(1,t) = Q_t - q_t$$

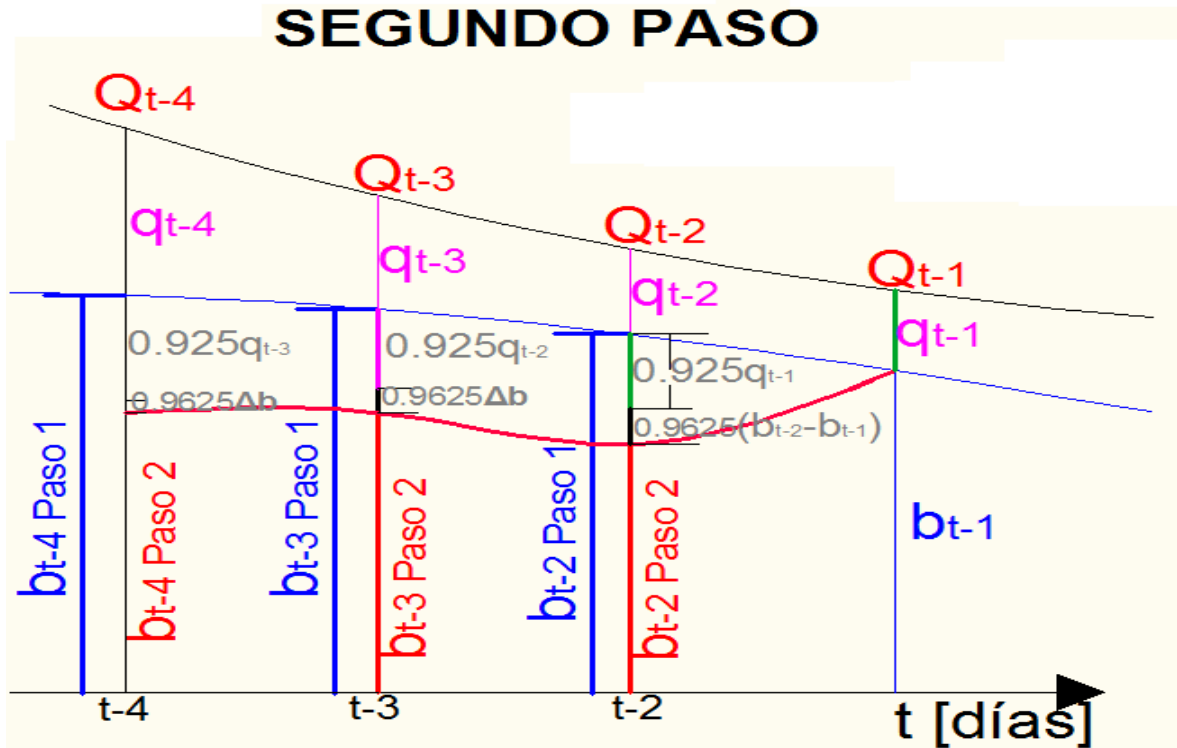


FIGURA 11. Representación gráfica PASO 2 (hacia atrás). (La autora)

2.3.4.2 PASO 2 (hacia atrás)

$$b(2, i-1) = b(1, i-1)$$

Debido a que por motivos de programación, el último valor calculado para todos los pasos es igual a cero, el paso 2 empieza a calcular el flujo base desde el penúltimo lugar, es decir, desde la posición $i-1$ el cual lo iguala al flujo base calculado en el primer paso en esa misma posición y a partir de la posición $i-2$, lo calcula con las siguientes ecuaciones:

$$\text{Volumen escorrentía } q_t = 0.925 * q_{t+1} + 0.9625 * (b_t - b_{t+1})$$

$$\text{Flujo base } b(2, t) = b(1, t) - q_t$$

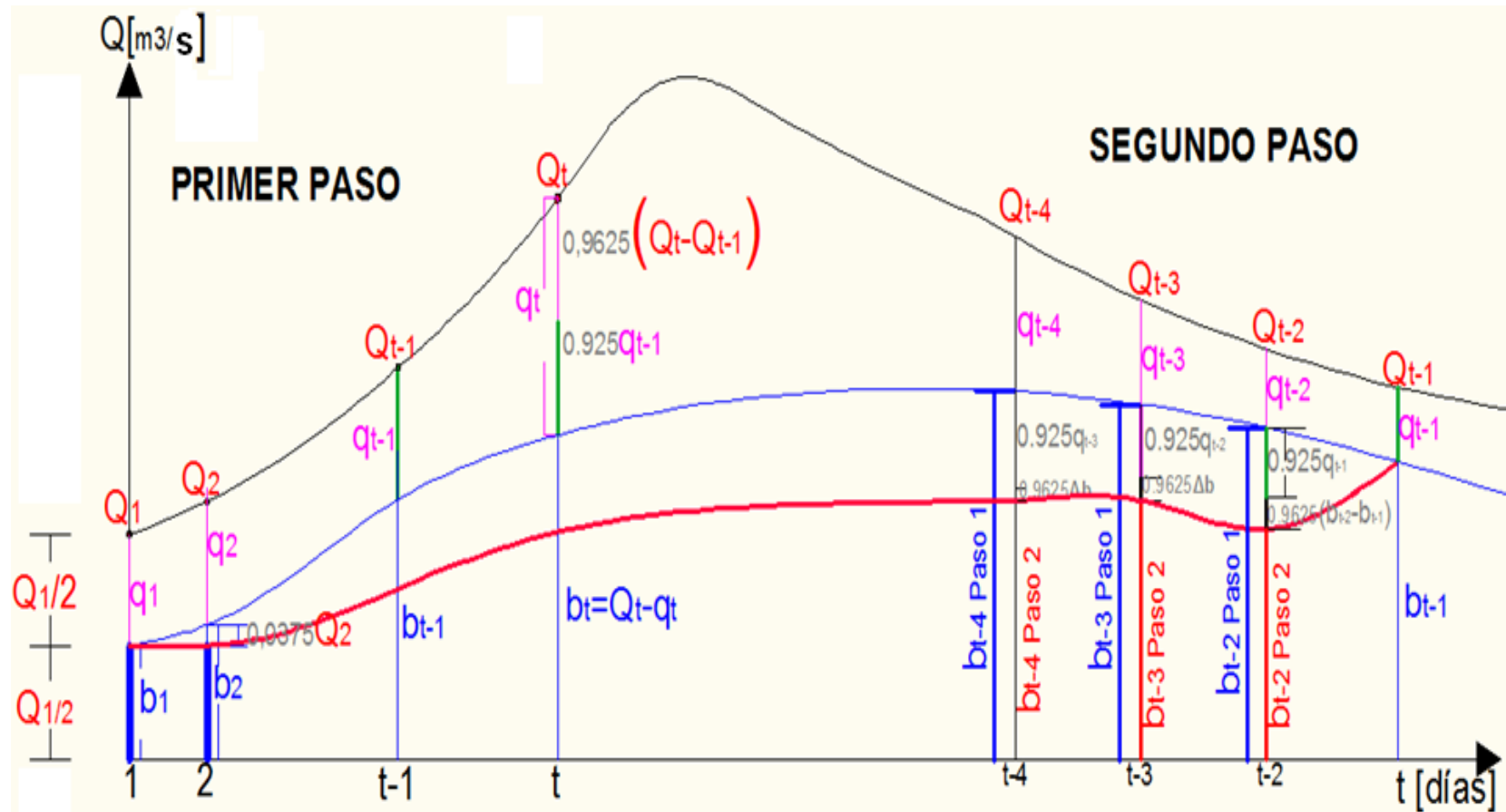


FIGURA 12. Representación gráfica PASOS 1 y 2. (La autora)

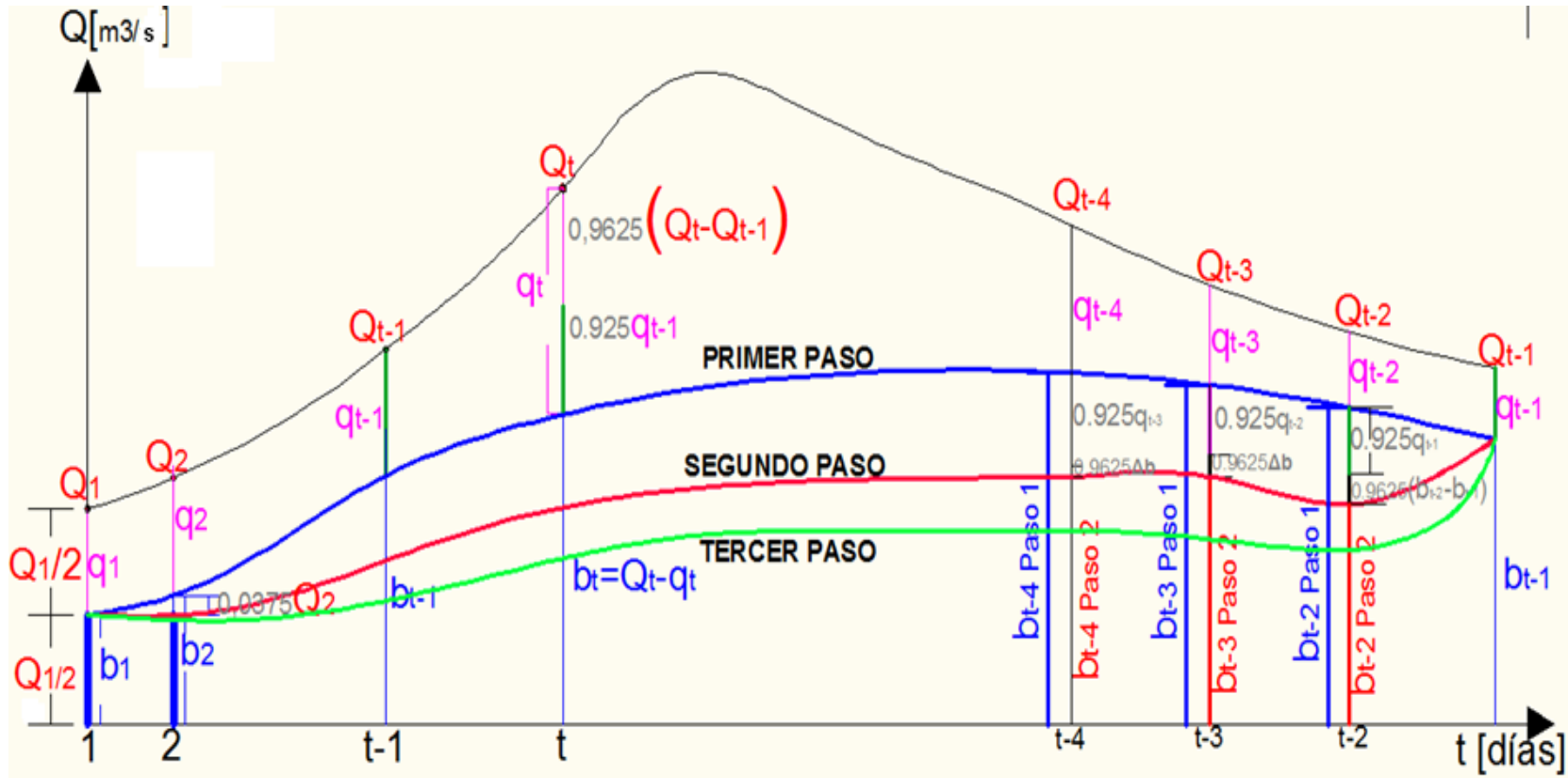


FIGURA 14. Representación gráfica PASOS 1, 2 Y 3. (La autora)

Los tres pasos empiezan y terminan con el mismo valor de flujo base, los cuales se calculan en el paso 1. El flujo base se reduce aproximadamente un 17% en el segundo paso y un 10% adicional en el tercer paso.

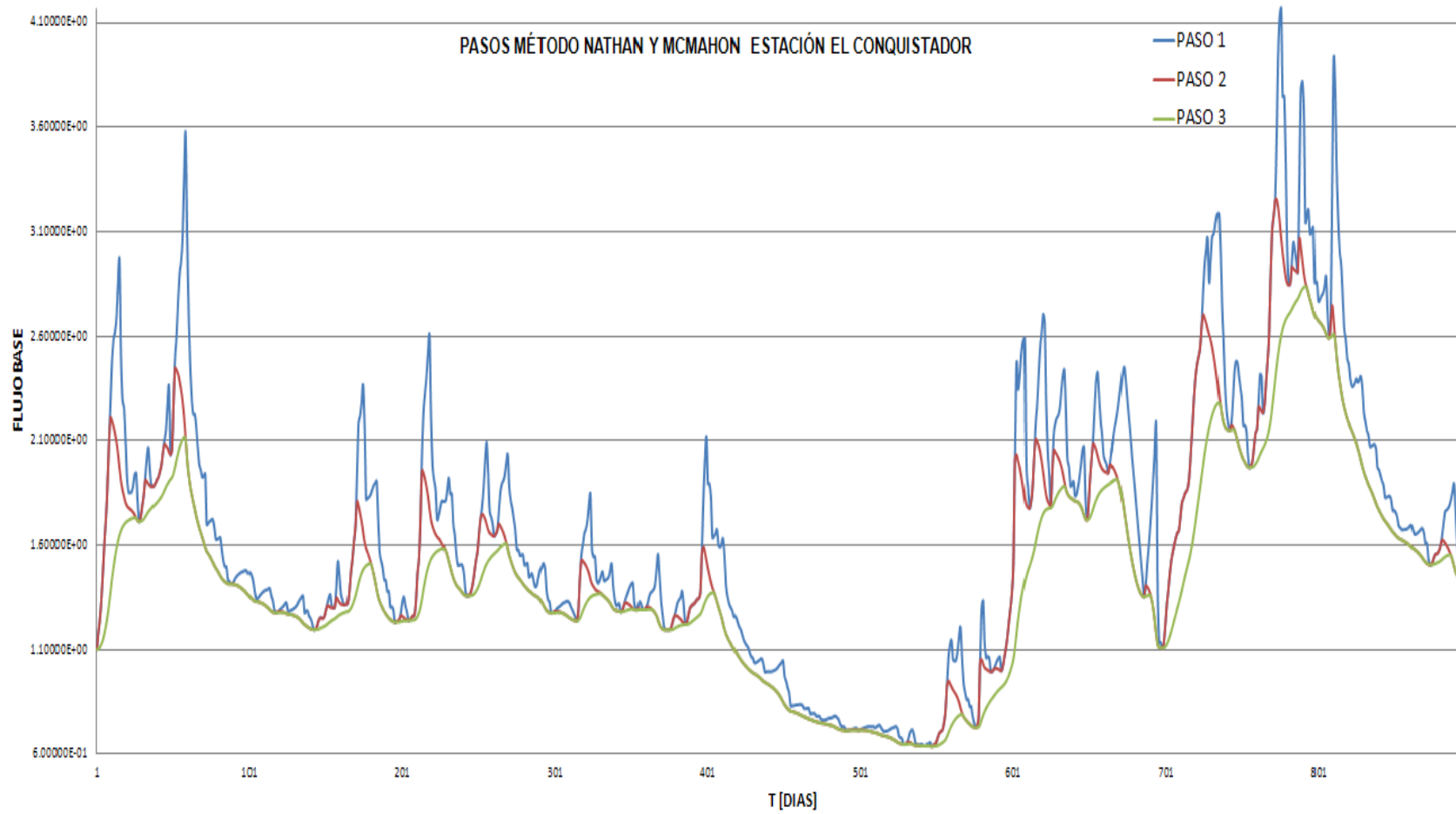


FIGURA 15. Representación gráfica de los PASOS en la Estación El Conquistador. *(La autora)*

2.3.5 Aplicación del Algoritmo de Separación de Nathan y McMahon

Antes de llevar a cabo la separación de flujo base se determinó la constante de recesión la cual se halló determinando la curva maestra de recesión (figura 9); se encontró una constante de recesión $\alpha = 0.025$.

El análisis de caudal para la Cuenca alta del río de Oro se llevó a cabo mediante el software Baseflow filter el cual se basa en el método de separación de flujo base de Nathan y McMahon (N & M), que involucra tres pasos. Todos los pasos dieron un porcentaje grande de la componente de flujo base, entre 70 y 80 por ciento en promedio; el primer paso arrojó los mayores porcentajes, es decir, entre 75 y 84 % mientras que el tercer paso lo subestima con valores entre 61 y 74%.

La figura 17 presenta los resultados de las componentes de flujo base y flujo directo a partir de los caudales diarios en la cuenca alta del río de Oro durante el período de agosto de 2008 a marzo de 2011, según lo calculado por las ecuaciones de Nathan y McMahon.

2.3.5.1 Porcentajes de flujos base dados por los pasos del algoritmo:

- En el período de octubre a diciembre de 2008 los flujos base calculados por los pasos 1, 2 y 3 respectivamente del método de Nathan & McMahon contribuyeron el **75%, 67% y 61%** del caudal total medido en la estación El Conquistador.
- Durante el año 2009, el flujo base fue estimado de **84%, 77% y 74%** del caudal total según los pasos 1, 2 y 3 respectivamente.
- En el año 2010 la contribución del flujo base fue del **76%, 69% y el 63%** del flujo total, cuando se estiman por los pasos 1, 2 y 3, respectivamente.
- Para los tres primeros meses del 2011, es decir, para el período de enero a marzo los porcentajes de flujo base estimados fueron de **83%, 79% y 73%**

2.3.5.2 Comparación de los pasos de Separación del método de Nathan y McMahon

En la figura 18 se muestra una comparación de los tres pasos de separación de flujo base en la estación El Conquistador durante el período de 02 de agosto de 2008 al 31 de marzo de 2011, donde se observa el comportamiento del flujo base de cada paso y en el numeral anterior se presenta los valores en porcentaje arrojados correspondientes a cada paso.

Para determinar cuál de los pasos utilizados en el algoritmo de separación se asemejaba mejor a las condiciones del flujo total del río, se contó con la ayuda de datos de flujo base obtenidos por el modelo hidrológico Tetis realizado por Juan Carlos Forero como tesis de maestría. Dentro de los reportes que muestra este modelo existe un archivo que corresponde al resultado final de la modelación, donde se encuentra por una parte la lista de datos medidos en la estación de aforo del río y por otra parte está el caudal base, el cual es un elemento del modelo, por cada intervalo de tiempo. Desde un comienzo, los resultados se esperaron diferentes teniendo en cuenta que en el modelo se corren diferentes eventos de simulación con diferentes condiciones en la cuenca, sin embargo, se trató de hacer una aproximación con los datos suministrados con el propósito de hacer un acercamiento al comportamiento del flujo base y así, obtener el paso más aproximado.

De esta forma, teniendo el porcentaje promedio de flujo base dado por la separación del modelo Tetis, se procedió a comparar con los pasos del algoritmo para ver cuál tenía el mejor comportamiento y se encontró que el valor que más representa a la cuenca alta del río de Oro está entre los pasos 1 y 2, entonces, se puede aproximar que el porcentaje de Flujo base para el período de octubre de 2008 a marzo de 2011 más probable es de **76.5%**, lo cual representa un componente importante de flujo base.

2.3.5.3 Flujo base para la cuenca alta del Río de Oro:

Debido a que se ha comprobado que el porcentaje más certero está en el rango del primer y segundo paso, se puede afirmar que se encontraron componentes de flujo base del **71%** en 2008 y aumentó al **80.5%** en 2009, mientras que dio un **72.5%** y **81%** en 2010 y 2011, respectivamente.

El promedio de flujo base en los tres años analizados fue de **76.25%**. Esto demuestra que gran parte del caudal del río de Oro es suministrado por la descarga del agua subterránea, es decir, según los resultados, el flujo base es el componente dominante en la cuenca alta del río de Oro. Estos resultados corroboran con estudios ya realizados en varias subcuencas de la Cuenca Superior del río Lebrija [16], donde los componentes de flujo base son casi siempre de más de 65%.

En este estudio, la indicación de un mayor flujo base puede ser debido al flujo de los acuíferos, entre lo que puede ser el resultado de su formación geológica. Esto también puede indicar la existencia de aportes de flujo provenientes de cuencas vecinas. En este trabajo no se llegó a indagar sobre el origen de estos flujos, pero existe la posibilidad de que su procedencia corresponda a flujos subterráneos de la zona de la mesa de Berlín, por la existencia de algunos afloramientos en la parte alta de la cuenca, pero para corroborar estas sospechas, se requiere de otros estudios más específicos, como es el estudio con isótopos y análisis de muestras en la zona.

Asimismo, la pendiente de recesión coincide notablemente con el análisis de la Curva Maestra de Recesión (figura 9) hecho anteriormente. Para el período completo de estudio (2008 – 2011) Baseflow filter da como resultado una pendiente Alpha de **-0.0269** y en el análisis manual un valor de **-0.0251**, obteniendo un porcentaje de error total de 6.6%.

CAPITULO 4

3. INTERFAZ GRÁFICA EN MATLAB CREADA PARA EL SOFTWARE BASEFLOW FILTER

La creación de la interfaz gráfica se realizó con el fin de visualizar la información resultante del software Baseflow filter y poder ejecutarlo dentro de un programa de fácil acceso como Matlab. Para su comprobación se requirió convertir la información existente de la estación El Conquistador, a los formatos y características que requiere el software Baseflow Filter **(Ver Anexo A)**.

3.1 INSTRUCCIONES DE USO DE LA INTERFAZ GRÁFICA (Ver Anexo A)

En la pantalla inicial aparece un menú denominado Operations, que incluye dos submenús que se encargan de mostrar las gráficas, botones y demás objetos correspondientes a la opción elegida. Cada uno de los elementos está oculto y al presionar cualquier submenú se hacen visibles.

Cuando se presiona la opción Operation1 del menú Operations, el usuario tiene la opción de visualizar el hidrograma, los pasos calculados de flujo base por el software Baseflow filter, la Curva Maestra de Recesión, las fechas de los tramos de recesión más largos e información adicional sobre las gráficas en un período de tiempo específico.

El usuario digita el período de tiempo que desea visualizar, que debe ser mayor o igual a dos años y al presionar el botón GRAFICAR HIDROG se muestra en la primera gráfica el hidrograma con la separación de flujo base hecha por los tres pasos, además la flecha del cursor cambia para que el usuario pueda seleccionar los tramos de recesión más largos, dando clic izquierdo sobre el hidrograma en donde se ubiquen los segmentos a elegir, teniendo en cuenta, siempre de

delimitarlos, es decir, dar clic donde comienza y termina el tramo. Para deshabilitar esta opción, después de haber escogido los segmentos, se da clic derecho sobre la gráfica y en el cuadro de texto que aparece abajo denominado Fechas, aparecen las fechas de inicio y fin de cada tramo seleccionado anteriormente. También se muestran los valores de flujo base calculados a partir de los datos de los archivos de entrada (caudales.prn) y salida (caudales.out).

En la sección denominada Días recesión, el usuario puede ingresar el tamaño a escoger para los tramos de recesión y seguidamente, al presionar el botón GRAFICAR TRAMOS, se visualizarán los segmentos que forman la curva maestra de recesión y a partir de ella se calcula la pendiente MCR, que sería la constante de recesión.

Al terminar, es necesario oprimir el botón reiniciar para borrar los elementos en la pantalla actual y con ello, iniciar una nueva graficación.

En la opción Operation2 del menú Operations, se visualiza el hidrograma, flujo base, curva maestra de recesión e información adicional sobre las gráficas para un año determinado. En la sección Año se ingresa el año a visualizar y luego al presionar el botón GRAFICAR HIDROG, igualmente, se muestra la gráfica correspondiente al hidrograma y flujo base, y porcentaje de flujo base. Por otro lado, en la sección Días de recesión, se ingresa el número de días para los tramos de recesión y al presionar el botón GRAFICAR TRAMOS se pueden visualizar la curva maestra de recesión con los segmentos seleccionados, la fecha de inicio del tramo más representativo, la cantidad de días que dura y la pendiente calculada a partir de la CMR. Se debe tener en cuenta que primero se debe presionar el botón Graficar Hidrog antes de ejecutar Graficar Tramos.

Por último, se debe oprimir el botón reiniciar para borrar las gráficas y poder cambiar de pantalla al elegir otra opción en el menú.

3.2 DESCRIPCIÓN

3.2.1 Descripción para Operation1 del menú Operation

El botón GRAFICAR HIDROG, busca el archivo de entrada 'caudales.prn' y el de salida 'caudales.out', toma cada una de las columnas de los archivos y los organiza en dos matrices separadas, sin embargo como el formato de la fecha en los resultados del archivo de salida varía en posiciones, es necesario ubicar los valores de flujo base en las posiciones correctas dentro de una matriz.

Para seleccionar los años definidos por el usuario dentro de las cajas de texto de Período, se toma el año inicial y se le suma uno en uno hasta llegar al año final (por ejemplo, si se ingresa 2009 a 2011 como período, en el vector se almacena 2009, 2010 y 2011), de tal forma que se almacenen en un vector y se obtiene el lapso de tiempo a escoger y se comparan con los datos del archivo de entrada y de salida para elegir los datos correspondientes al período definido que luego serán utilizados por el botón GRAFICAR TRAMOS para hacer la extracción de los tramos de recesión.

Por otro lado, los valores de flujo base se calculan teniendo en cuenta la siguiente fórmula: $\% \text{ de Flujo base} = (\text{Volumen de Flujo base} / \text{Volumen de escorrentía}) * 100$. También dentro de éste botón se encuentra las instrucciones necesarias para capturar las posiciones x y y obtenidas después de dar clic dentro de la gráfica que corresponden a la fecha y al caudal, posteriormente se imprime la fecha en un listbox y al dar clic derecho se deshabilita la opción de captura de datos a través del ratón.

El botón GRAFICAR TRAMOS, toma los datos correspondientes al período especificado por el usuario y a partir de ello, se separan los tramos ascendentes y descendentes a partir de los puntos de inflexión, éstos se almacenan en vectores siempre y cuando cumplan con los días de recesión que especificó el usuario.

Acto seguido, se utiliza nuevamente puntos de inflexión para retirar algunas secciones iniciales de los tramos descendentes y obtener los tramos de recesión. Ahora se toma cada tramo de recesión y se verifica cuáles son tramos representativos (es decir, si el tamaño del tramo de recesión supera los 10 días).

Si existen tramos representativos, se calcula la pendiente promedio de los puntos que forman el tramo y con base a ésta pendiente, se grafica la recta que pasa por los puntos finales de todos los tramos de recesión, de lo contrario, se toman los segmentos finales de cada tramo de recesión, se calcula la pendiente promedio y se imprimen los resultados.

El botón REINICIAR borra todas las gráficas, los valores contenidos en los cuadros de texto y las variables para que no afecten los resultados de un nuevo procedimiento.

3.2.2 Descripción para Operation2 del menú Operation

Para Operation2 se realiza el mismo procedimiento que para Operation1, solo que esta vez, se toma el año ingresado por el usuario en la sección Año y se comparan con los datos del archivo de entrada y de salida para elegir los datos adecuados, que luego serán utilizados por el botón GRAFICAR TRAMOS para hacer la extracción de los tramos de recesión. Además, para el tramo representativo (es decir, si el tamaño del tramo de recesión supera los 10 días) se guarda la fecha de inicio y el número de días que demora el tramo para luego imprimirlo en las cajas de texto correspondientes.

CAPITULO 5

CONCLUSIONES

- En el estudio de análisis de caudal en la cuenca alta del río de Oro, se encontró que las componentes de flujo base anuales oscilaron entre 71% y 81%, es decir que el caudal es principalmente alimentado con el agua subterránea. El hecho de que hubo registros de caudal diario continuo en la estación hidrométrica El Conquistador, incluso durante los largos períodos sin lluvia, confirma esta importante contribución al caudal, indicando la existencia de aportes de flujo de cuencas vecinas.
- Finalizada esta investigación, se ha realizado la aplicación del software Baseflow filter a series de datos de caudales en la estación El Conquistador por tres años con los cuales se desarrolló la validación del algoritmo de separación de flujo base de Nathan y McMahon. En general, el primer y segundo paso dieron como resultado siempre una estimación razonable de la componente de flujo base, pudiendose concluir que el valor más certero de flujo base se encuentra dentro del rango del paso 1 al 2, mientras que el tercer paso lo subestima.
- Se pudo ejecutar satisfactoriamente el software Baseflow filter dentro de Matlab y se creó la interfaz gráfica que permite representar la salida de datos que genera. El usuario entonces puede visualizar fácilmente los resultados además de interactuar con la interfaz ajustando las fechas de recesión que considera convenientes a lo largo del eje de la ordenada y también puede elegir el número de días de recesión que desea que tengan los tramos para el ajuste de la Curva Maestra de Recesión.

- Se encontró que la constante de recesión predominante en la cuenca alta del Río de Oro es de 0.025 aproximadamente. Esto representa que cuenta con buenas condiciones para el almacenamiento del recurso subterráneo, alta pluviosidad, estructura geológica favorable a la infiltración. Además, al estudiar los parámetros de recesión, se obtuvo valores altos de C, lo que indica que regula eficientemente el volumen almacenado.

RECOMENDACIONES

- Tanto las características hidrológicas como las geológicas de la subcuenca, pueden haber contribuido a una recarga relativamente alta de las aguas subterráneas que a su vez ofrecen un alto componente de flujo base a la corriente. Por lo tanto es aconsejable usar un modelo calibrado con base física que sea capaz de identificar la fuente de flujo base, además de su respectiva cuantificación. Se recomienda realizar estudios más específicos, como es el estudio con isótopos y análisis de muestras para averiguar la procedencia de los flujos subterráneos que alimentan la zona.
- La cantidad, continuidad y mejoramiento de la calidad de la información de la estación El Conquistador son aspectos de vital importancia para la aplicación de proyectos similares en la región. Para obtener un mejor registro de caudales se recomienda hacer revisión periódica de los instrumentos para evitar fallas en las mediciones y un adecuado mantenimiento para obtener series consistentes y homogéneas que permitan mejorar las estimaciones de flujo base en la cuenca.
- Debe continuarse con el estudio de algoritmos que reflejen en forma aún más realista el porcentaje del flujo base aportado al caudal como elemento fundamental del modelo hidrológico de la cuenca.

BIBLIOGRAFÍA

- [1]. **Acosta, M.R.** (2006), Aplicación del Modelo Hidrológico Conceptual NAM. Precipitación - Escorrentía en la Cuenca del Río de Oro, *Tesis de Pregrado, UIS, Bucaramanga.*
- [2]. **Aguilar, J.J., y Caballero, C.R.** (2010). Calibración e Instalación de un CTD Diver en la Estación Hidrométrica Automática del Río de Oro. *Tesis de Pregrado. UIS, Bucaramanga.*
- [3]. **Aparicio, F.** (1987). Fundamentos de hidrología de superficie., *Noriega editores Limusa.*
- [4]. **Arenas, J.P. y Ruiz, J.S.** (2005), Estudio de la Variación del Régimen de Lluvias en la Cuenca Superior del Río Lebrija, *Tesis de Pregrado, UIS, Bucaramanga.*
- [5]. **Ardila, D.M. y Calderón L.C.** (2006), Generación de Tormenta de Diseño Para las Estaciones Meteorológicas de la Cuenca del Río de Oro, *Tesis de Pregrado, UIS, Bucaramanga.*
- [6]. **Arnold J.G., Allen P.M., Muttiah R. and Bernhardt G.** (1995), Automated Baseflow Separation And Recession Analyses Techniques, *Ground Water* 33(6): 1010-1018.
- [7]. **Arnold, J.G. and Allen P.M.** (1999). Automated methods for estimating baseflow and ground water recharge from streamflow records. *Journal of the American Water Resources Association* 35(2): 411-424.

- [8]. **Cervantes, R.E. & Fajardo, E.A.** (2011), Análisis Y correlación de la información del CTD en la estación automática del río de Oro. *Tesis de Pregrado, UIS, Bucaramanga.*
- [9]. **Chapman, T. G.,** (1999). A comparison of algorithms for stream flow recession and baseflow separation. *Hydrological Processes. 13, 701-714.*
- [10]. **Chow, V. T.** (1994) Handbook of applied hydrology. *McGraw-Hill. New York.*
- [11]. **Coronado, W. & Mejía, C.** (2006) Cartografía geológica y caracterización estructural con fines hidrogeológicos en el sector nororiental de la cuenca del Río de Oro. *UIS, Bucaramanga.*
- [12]. **Díaz A. W. y Rincón W. Y.** (2006), Montaje y calibración de una estación hidrométrica automática para medición de niveles en el Río de oro, *Tesis de Pregrado, UIS, Bucaramanga.*
- [13]. **Flórez, W. y Rueda, J.A.,** (2009). Curva de Calibración y Funcionamiento de la Estación Hidrométrica Automática del Río de Oro. *Tesis de Pregrado, UIS, Bucaramanga.*
- [14]. **Gómez, S., Colegial, J.,** (2003). Interacción entre sistemas hidrogeológicos para el estudio del fenómeno de recarga en el macizo de Santander e identificación de acuíferos. *Tesis de Pregrado, UIS, Bucaramanga.*
- [15]. **González N.A. y Sossa L.C.** (2005), Eventos Extremos de Precipitación: Caso de Estudio Febrero 12-05 Girón, *Tesis de Pregrado, UIS, Bucaramanga.*
- [16]. **Guzmán, J.A., A. Anaya. E. Niño. S. Gómez. y J.D. Colegial.,** 2005; Cuenca Experimental del Río Sucio Municipio de Tona. *Informe Técnico.*

Grupo en Predicción y Modelamiento Hidroclimático, Universidad Industrial de Santander, Bucaramanga.

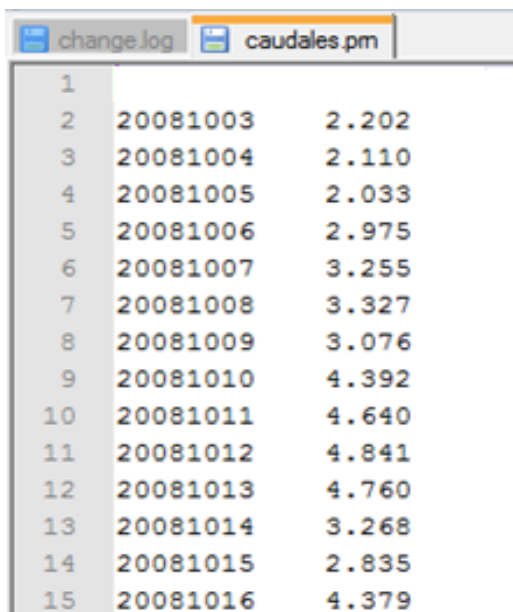
- [17]. **Hernández, L.I. y Saltarín D.A.** (2007), Montaje, Operación y Calibración de la estación Hidrométrica Automática Para Medición de Niveles en el Río de Oro, Estación Conquistador Alto, *Tesis de Pregrado, UIS, Bucaramanga.*
- [18]. **Jaramillo M.** (2003), Estudio de la Interacción Acuífero-Río Como Soporte Metodológico en la Estimación de Caudales Mínimos, *Tesis de Maestría, Universidad Nacional de Colombia, Medellín.*
- [19]. **López W.G. y R.M. Osorio** (2005), Modelo Digital de Elevación para la Cuenca del Río de Oro, *Tesis de Pregrado, UIS, Bucaramanga.*
- [20]. **Nathan R. J. y McMahon T. A.** (1990) Evaluation of automated Techniques for baseflow and recession analyses. *Universidad de Melbourne, Melbourne, Australia*
- [21]. **Osorio, L. & Salazar, O.** (2006) Cartografía geológica y caracterización estructural de la subcuenca del Río Tona, macizo de Santander, con fines hidrogeológicos. *Tesis de Pregrado, UIS, Bucaramanga.*
- [22]. **Salazar, J.L., y Quitian, D. O.,** (2009). Elaboración de Un Modelo Digital de Elevación y Delimitación del Tipo de Vegetación Presente en la Cuenca Superior del Río de Oro. *Tesis de Pregrado, UIS, Bucaramanga.*
- [23]. **Serrano, A.M., y Balaguera, G.,** (2006). Separación de Flujo Base en la Estacion Café Madrid (C.S.R.L.). *Tesis de Pregrado, UIS, Bucaramanga.*

- [24]. **Singh, V. P.** (1988) Hydrologic Systems, Vol. 1. Rainfall-Runoff Modeling. *Prentice-Hall, Inc. Englewood Cliffs, New Jersey.*
- [25]. **Sloto,R. and Croase, M.** (1996) *Hysep: A computer program for streamflow hydrograph separation and análisis.* En: *U:S: Geological Survey Water Resources, 1996. 1-7*
- [26]. **Soil Conservation Service** (1972) Hydrology. National Engineering Handbook. Section 4, *U.S. Department of Agriculture, Washington D.C.*
- [27]. **Ulloa, A. M. & Gómez, S.** (2007). Proyecto de investigación, Geológica e hidrogeológica en la región central del departamento de Santander. *Bucaramanga: INGEOMINAS. U.I.S.*
- [28]. ***Forero, J.C.** (2012). Modelo Hidrológico Distribuido de la Cuenca Superior del Río De Oro. *Tesis de Posgrado, UIS, Bucaramanga.*
- [29]. ***CDMB,** (2007) Sistema de Información (SIA), SIG

ANEXOS

ANEXO A. MANEJO DEL SOFTWARE Y LA INTERFAZ GRAFICA

- Formato de las archivos de entrada **caudales.prn** y **file.lst**



1		
2	20081003	2.202
3	20081004	2.110
4	20081005	2.033
5	20081006	2.975
6	20081007	3.255
7	20081008	3.327
8	20081009	3.076
9	20081010	4.392
10	20081011	4.640
11	20081012	4.841
12	20081013	4.760
13	20081014	3.268
14	20081015	2.835
15	20081016	4.379

El archivo **caudales.prn** debe contener los registros de caudales diarios con su respectiva fecha separados por al menos 1 espacio o 1tab y puede ser creado en Excel e importado a block de notas o notepad ++; se debe guargar siempre como caudales.prn, así se ajustará al formato requerido para archivo de entrada al programa.

La primera línea debe ir vacía o puede contener texto como el nombre del río por ejemplo, ya que esa línea no será tenida en cuenta para el cálculo.

El formato de fecha debe ser “año mes día” sin ningún espacio, éste se puede ajustar en Excel, seleccionando las celdas, dando click izquierdo, en formato de celdas – Número – Personalizada – Tipo: yyyyymmdd - Aceptar

```
file: Bloc de notas
Archivo Edición Formato Ver Ayuda
!!Input for baseflow program:
  10 !NDMIN: minimum number of days for alpha calculation
  10 !NDMAX: maximum number of days for alpha calculation
  1 !IPRINT: daily print option (0-no; 1-yes)

!!Daily stream data files
caudales.prn      caudales.out
```

El archivo **file.lst** debe contener las siguientes variables:

NDMIN: Número mínimo de días utilizados para calcular el valor de alfa. La configuración predeterminada para NDMIN es de 10.

NDMAX: Número máximo de días utilizados para calcular el valor de alfa. La configuración predeterminada para NDMAX es de 300.

IPRINT: código de impresión para los valores de resumen de las fracciones de flujo base, alfa y días de flujo base. Además de la salida estándar, el usuario puede elegir para imprimir los valores filtrados para flujo de base para los tres pases realizados por el programa, así:

0 No imprimir los valores diarios - 1 Imprimir los valores diarios

Si IPRINT se establece en 1, el archivo de salida **caudales.out** debe ser definido.

El número debe ser la primera cosa en la línea y que éste debe ser seguido por un espacio para separarlo de cualquier otra cosa escrita en la línea.

Las variables NDMIN, NDMAX y IPRINT deben estar definidas en las líneas 2, 3 y 4, respectivamente. Asimismo, el nombre de los archivos de entrada y salida deben ocupar un máximo de 15 espacios, es decir, caudales.out debe estar escrito a partir del espacio 17 para que el programa los pueda identificar adecuadamente, así:

caudales.prn caudales.out

- La carpeta debe contener los siguientes archivos, al hacer doble click sobre hysep.m (archivo tipo Matlab Code) se abrirá Matlab

Nombre	Fecha de modifica...	Tipo	Tamaño
bflow	12/06/2006 01:07 ...	Aplicación	413 KB
caudales	23/04/2012 09:11 a...	CorelDRAW X5 Gr...	15 KB
file	13/04/2012 11:45 a...	Archivo LST	1 KB
hysep	23/04/2012 06:52 a...	Archivo ASV	46 KB
hysep	23/04/2012 09:30 a...	MATLAB Figure	25 KB
hysep	25/04/2012 04:28 ...	MATLAB Code	46 KB

- Al lado izquierdo aparecerán los archivos de entrada, si no se visualizan, se puede abrir la carpeta haciendo click en Desktop - Current Folder.

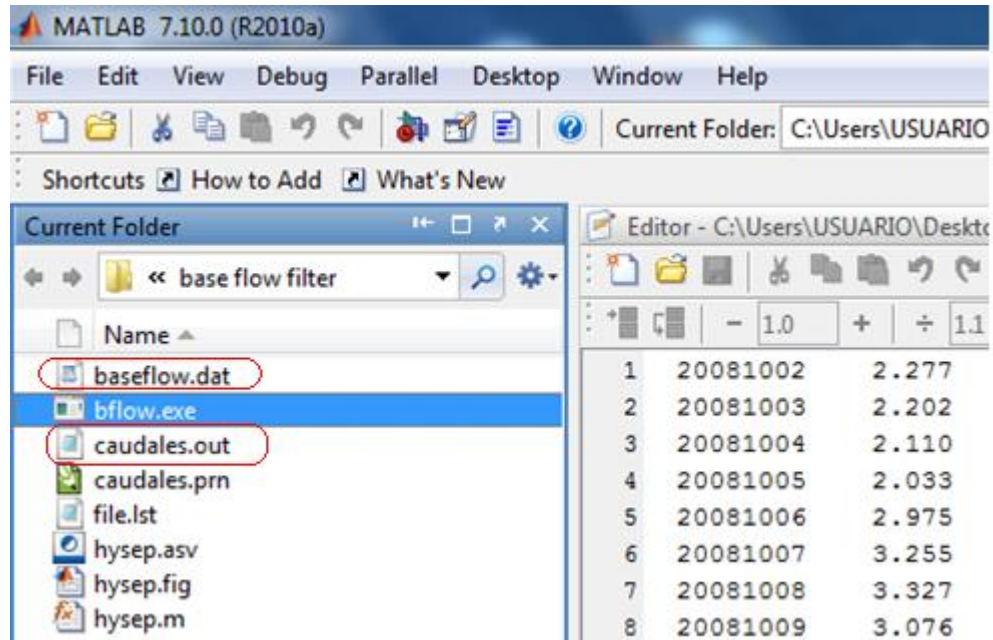
The screenshot shows the MATLAB 7.10.0 (R2010a) interface. On the left, a file explorer window titled 'Current Folder' shows the contents of a folder named '<< base flow filter'. The files listed are:

- bflow.exe
- caudales.prn
- file.lst
- hysep.asv
- hysep.fig
- hysep.m

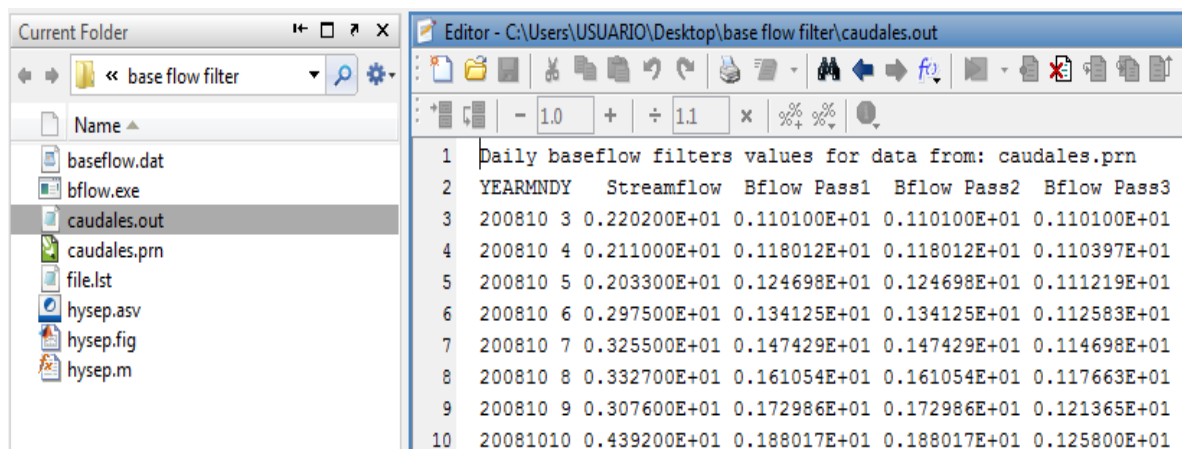
On the right, the MATLAB Editor window shows a table with 8 rows of data:

1	20081002	2.277
2	20081003	2.202
3	20081004	2.110
4	20081005	2.033
5	20081006	2.975
6	20081007	3.255
7	20081008	3.327
8	20081009	3.076

- Se debe ejecutar bflow.exe dándole doble click, al hacerlo aparecerán los archivos de salida .out y .dat



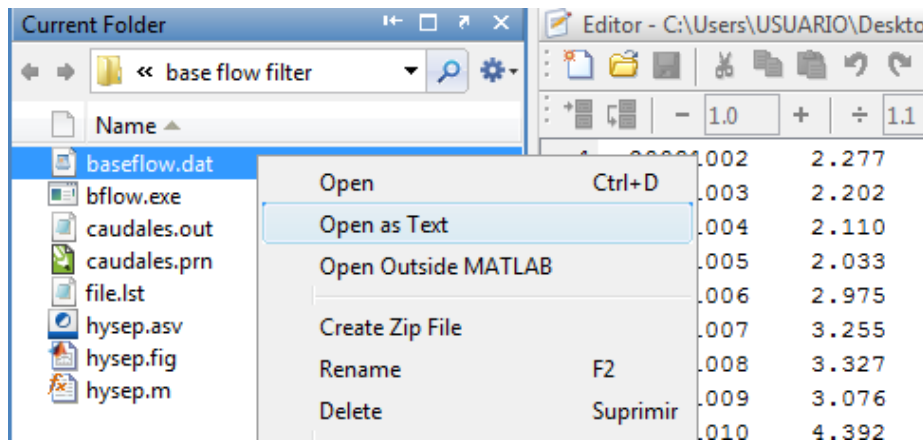
- Se puede visualizar el archivo de salida caudales.out al darle doble click, mientras que para el el archivo de salida .dat se debe dar click derecho – open as text.



Ejemplo de la salida de datos (.out) del software (La autora)

Este archivo contiene la fecha, el valor de caudal diario del archivo de entrada y el valor de flujo base que es estimado por cada uno de los tres pasos del algoritmo para cada valor de caudal diario dado. Se observa el caudal total Q_T (m^3/seg) y los tres pasos de flujo base Q_B (m^3/seg) para cada día del año según el formato año-mes-día (yyyymmdd).

- Archivo de salida (.dat) del software

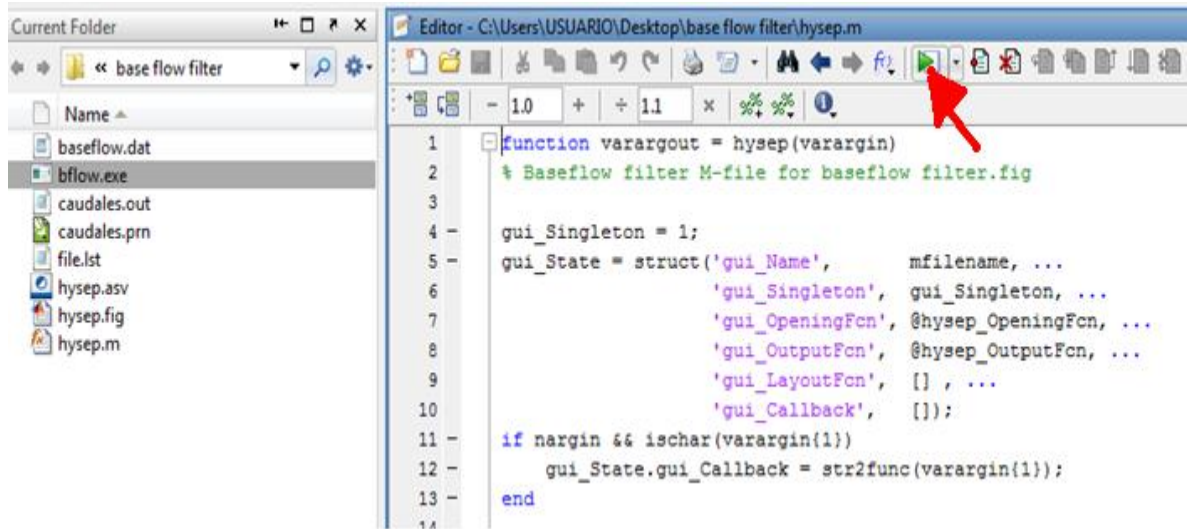


1	Baseflow data file: this file summarizes the fraction of streamflow that is contributed by						
2							
3	Gage file	Baseflow Fr1	Baseflow Fr2	Baseflow Fr3	NPR	Alpha Factor	Baseflow Days
4	caudales.prn	0.80	0.73	0.69	4	0.0269	85.3993
5							

Archivo de salida (.dat) del software (*La autora*)

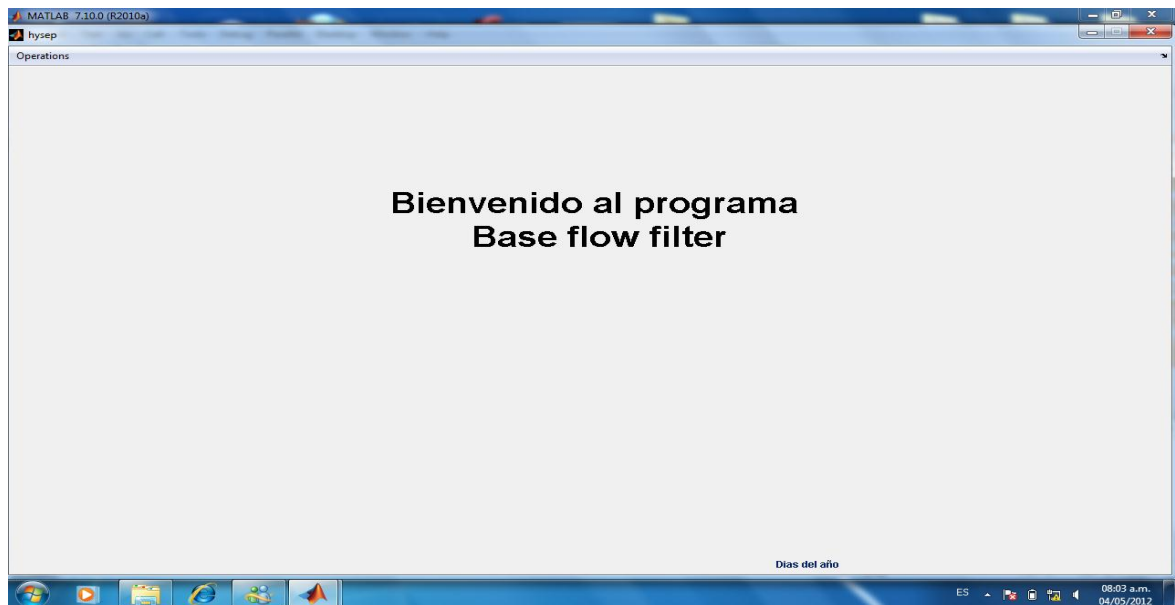
Este archivo contiene las fracciones de caudal que son aportadas por el flujo de base estimados para cada paso del método (Baseflow Fr), NPR que es el número de recesiones individuales que se utilizan para calcular la curva maestra recesión, Alpha Factor que es la constante de recesión y Baseflow days que es el tiempo por ciclo logarítmico de recesión de caudal, o en otras palabras, el tiempo en días que tarda el caudal en reducirse en un factor de 0.1.

- Luego de haber ejecutado el software en Matlab, se puede observar los resultados a través de la interfaz gráfica creada en el presente proyecto. Al drale click al botón Play señalado en la siguiente figura:

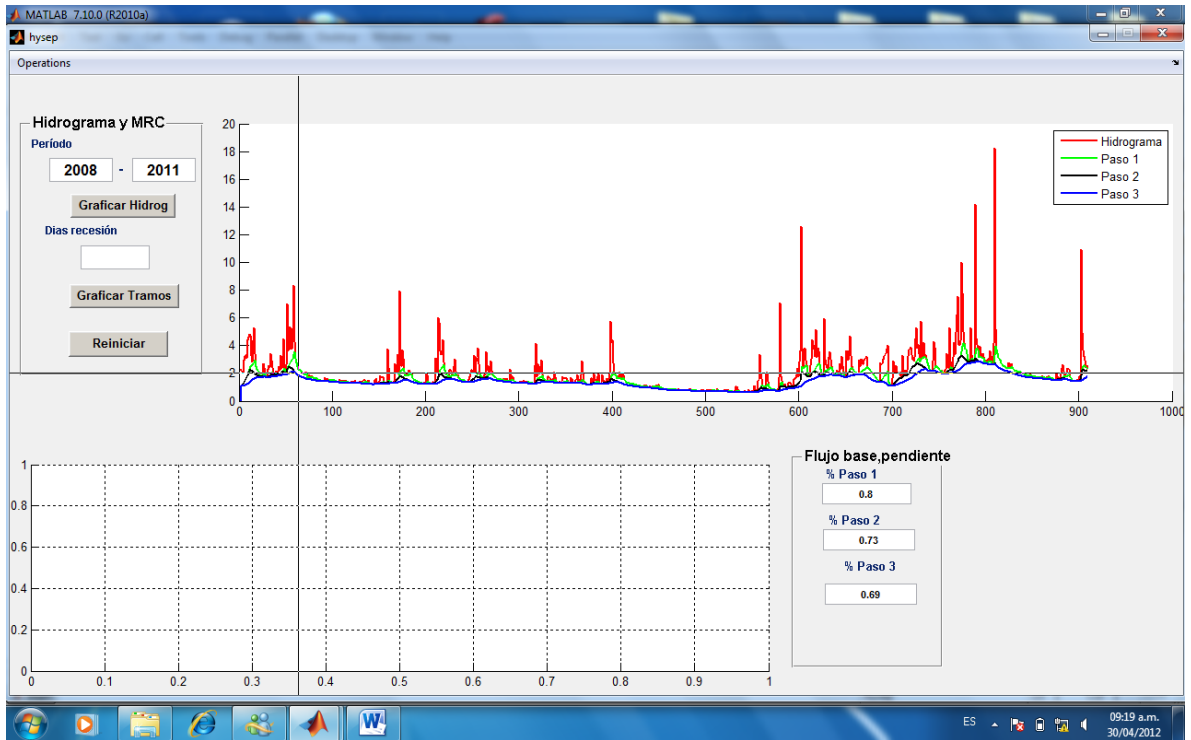


- A continuación se presenta la representación gráfica correspondiente a la explicación del **Capítulo 4**.

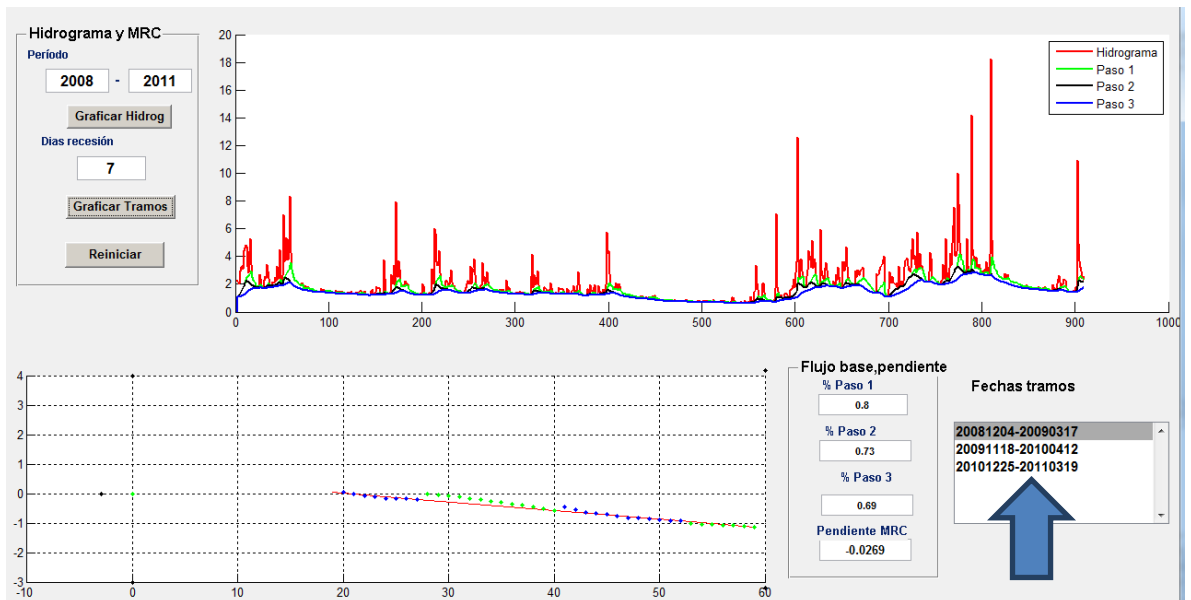
Presentación de la interfaz: **Operation1** para visualizar todo el período completo



- Escogencia de las fechas de recesión por el usuario:

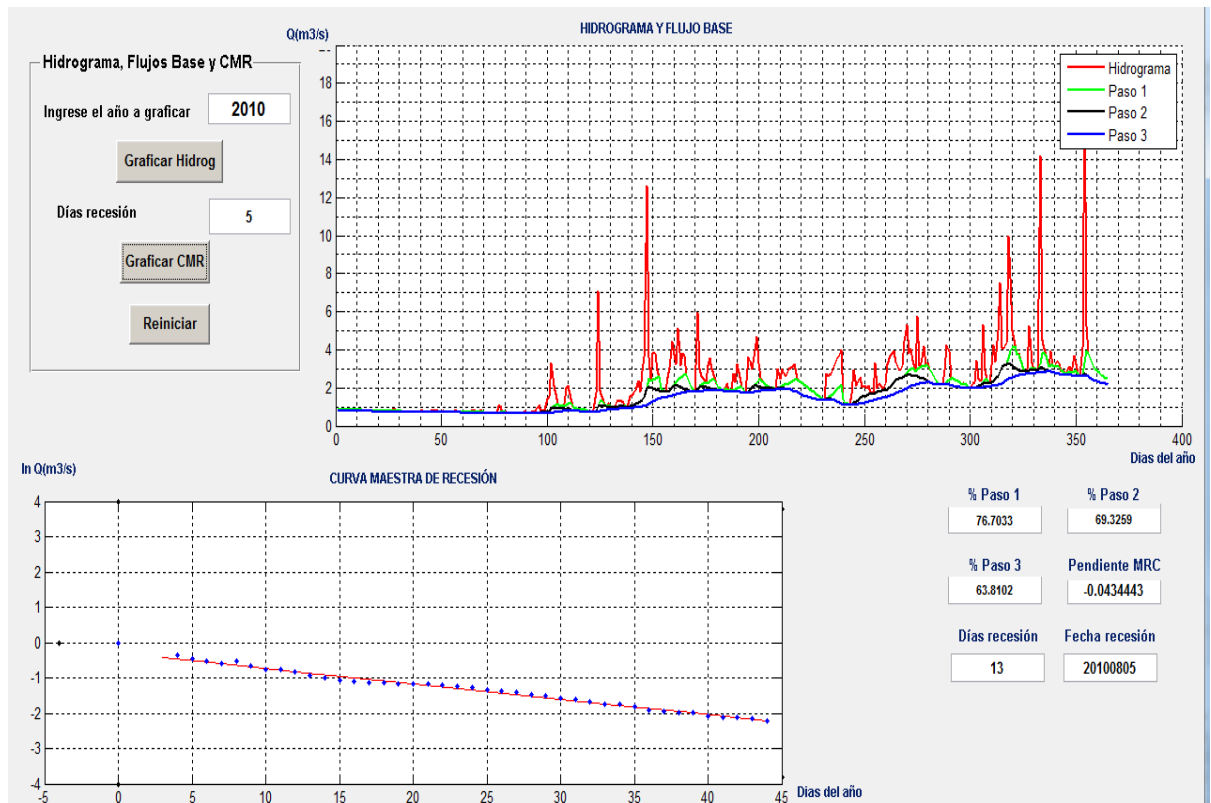


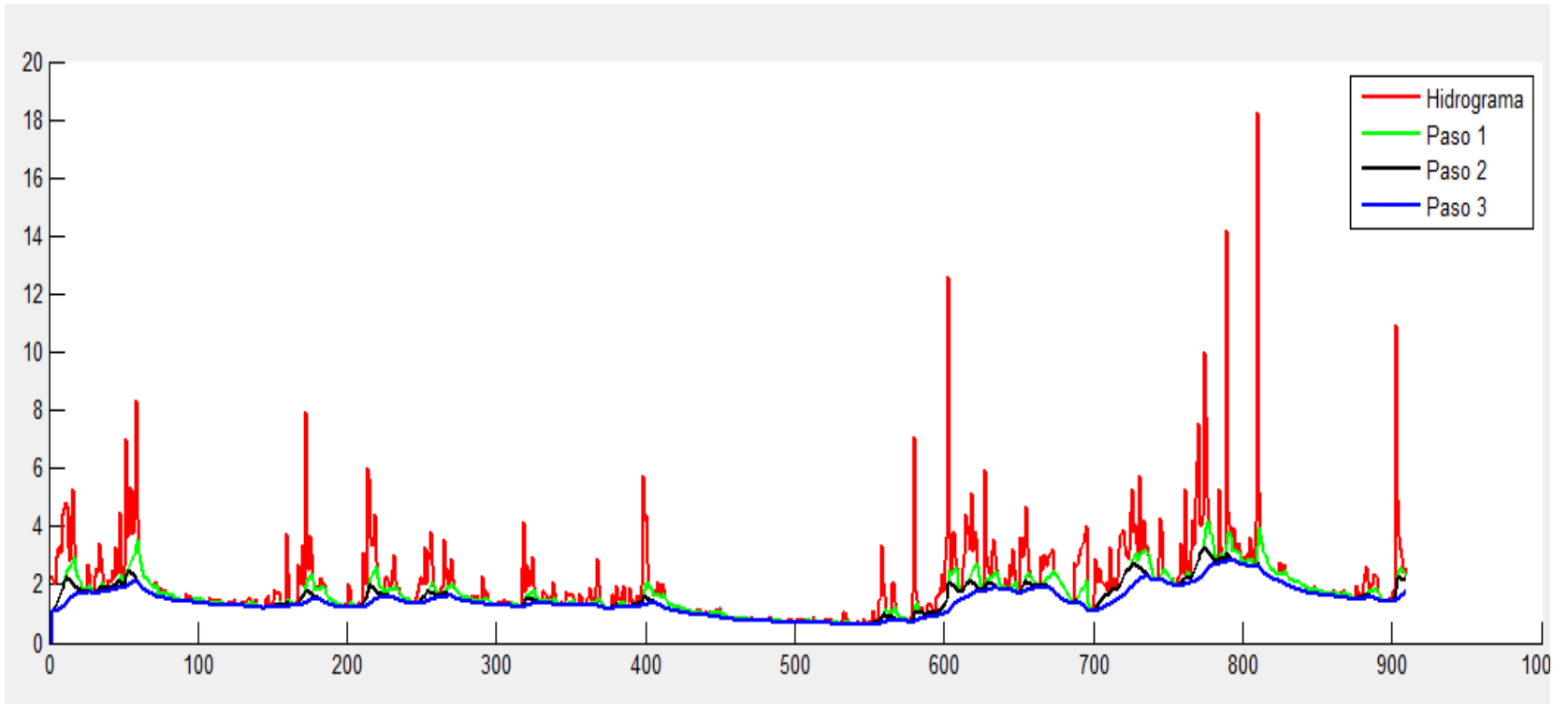
- Las fechas de recesión se encuentran señaladas en la siguiente figura:



- **Operation 2:**

De igual forma se realizó una interfaz gráfica que muestra la información deseada para el año que el usuario desee, así, por ejemplo, para el año 2010 se muestra el hidrograma, los tres pasos de separación de flujo base con sus respectivos porcentajes, la Curva Maestra de recesión, el valor de la constante de recesión y la fecha de inicio de la recesión más representativa.





Gráfica de Separación de flujo base estación el Conquistador 2008 -2011

Fuente: La autora

ANEXO B. ESTACIÓN AUTOMÁTICA DE AFORO

Para la toma de datos de caudal sobre la corriente del Río de Oro se utilizó la información suministrada por la estación de aforo instalada por intermedio del convenio realizado entre la UIS y la Piedecuestana de Servicios. [12], [13], [17].



Estación de aforo Conquistador y equipo [28]

ANEXO C. CAUDALES MÁXIMOS Y MÍNIMOS

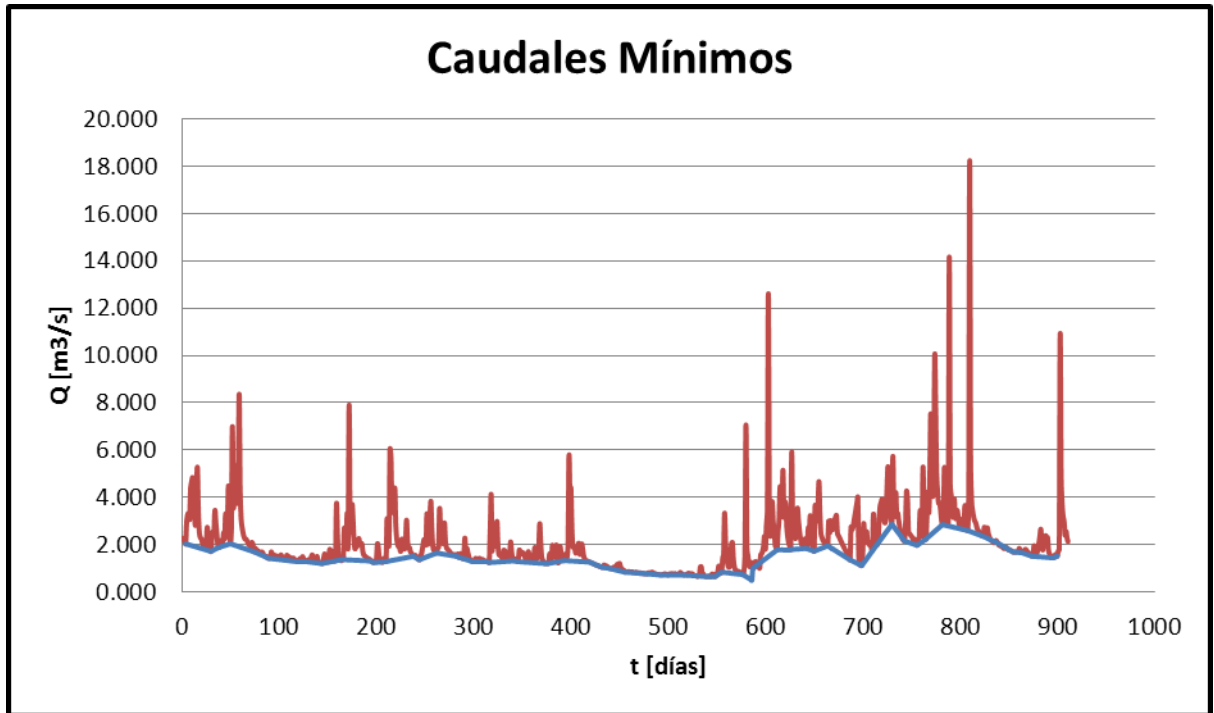
			Q [m ³ /s]
2008	Octubre	Máximo	5.253
		Mínimo	1.714
2008	Noviembre	Máximo	8.364
		Mínimo	1.881
2008	Diciembre	Máximo	3.106
		Mínimo	1.416
2009	Enero	Máximo	1.693
		Mínimo	1.277
2009	Febrero	Máximo	1.630
		Mínimo	1.199
2009	Marzo	Máximo	7.909
		Mínimo	1.301
2009	Abril	Máximo	3.096
		Mínimo	1.232
2009	Mayo	Máximo	6.028
		Mínimo	1.465
2009	Junio	Máximo	3.837
		Mínimo	1.358
2009	Julio	Máximo	2.281
		Mínimo	1.276
2009	Agosto	Máximo	4.124
		Mínimo	1.351
2009	Septiembre	Máximo	2.116
		Mínimo	1.282
2009	Octubre	Máximo	2.890
		Mínimo	1.195
2009	Noviembre	Máximo	5.747
		Mínimo	1.148
2009	Diciembre	Máximo	1.920

		Mínimo	0.658
2010	Enero	Máximo	1.363
		Mínimo	0.511
2010	Febrero	Máximo	2.346
		Mínimo	0.503
2010	Marzo	Máximo	2.240
		Mínimo	0.514
2010	Abril	Máximo	13.307
		Mínimo	0.532
2010	Mayo	Máximo	38.705
		Mínimo	0.479
2010	Junio	Máximo	30.709
		Mínimo	1.350
2010	Julio	Máximo	19.600
		Mínimo	1.423
2010	Agosto	Máximo	3.036
		Mínimo	0.809
2010	Septiembre	Máximo	14.576
		Mínimo	0.974
2010	Octubre	Máximo	17.401
		Mínimo	1.204
2010	Noviembre	Máximo	42.990
		Mínimo	1.920
2010	Diciembre	Máximo	48.251
		Mínimo	0.801
2011	Enero	Máximo	4.690
		Mínimo	1.423
2011	Febrero	Máximo	4.264
		Mínimo	0.848
2011	Marzo	Máximo	104.270
		Mínimo	1.197

ANEXO D. PARÁMETROS DE RECESIÓN POR RECESIÓN INDIVIDUAL

ESTACIÓN EL CONQUISTADOR							
SERIE	FECHA	α (días ⁻¹)	Q[m3/s]	C días	LNQ	R ²	R
1	2008-10-02	-0.0378051	0.8290492	26	-0.18747584	0.89601	0.94658
2	2008-12-15	-0.0245419	0.6447717	41	-0.43885904	0.93598	0.96746
3	2008-12-18	-0.0361371	0.6725097	28	-0.39673872	0.90909	0.95346
4	2008-12-28	-0.006859	0.6029174	146	-0.50597502	0.89749	0.94736
5	2009-01-08	-0.0165134	0.7465302	61	-0.29231922	0.91459	0.95634
6	2009-01-13	-0.0297957	0.7728283	34	-0.2576984	0.95038	0.97488
7	2009-01-20	-0.0088878	0.9037987	113	-0.10114858	0.98915	0.99456
8	2009-02-18	-0.0152061	0.8850858	66	-0.12207067	0.93266	0.96574
9	2009-03-12	-0.0193448	0.3430384	52	-1.06991302	0.95156	0.97548
10	2009-04-06	-0.0246335	0.5714518	41	-0.55957516	0.93594	0.96744
11	2009-04-23	-0.0273664	0.4927308	37	-0.70779235	0.88074	0.93848
12	2009-05-11	-0.030412	0.9983893	33	-0.00161198	0.97505	0.98745
13	2009-05-29	-0.0295052	0.8565048	34	-0.15489533	0.95326	0.97635
14	2009-06-17	-0.0178448	1.09595	56	0.09162156	0.99130	0.99564
15	2009-07-02	-0.0306416	1.1888238	33	0.17296442	0.89941	0.94837
16	2009-08-03	-0.025336	1.0721603	39	0.06967558	0.91480	0.95645
17	2009-08-06	-0.015149	1.1685866	66	0.15579502	0.95346	0.97645
18	2009-10-07	-0.0498473	1.3065219	20	0.26736854	0.91133	0.95464
19	2009-11-20	-0.0173757	0.3396801	58	-1.07975094	0.93033	0.96454
20	2009-11-25	-0.0175117	0.4239789	57	-0.85807159	0.95344	0.97644
21	2009-12-01	-0.0097039	0.5108833	103	-0.67161404	0.97509	0.98746
22	2009-12-03	-0.0204029	0.5793846	49	-0.54578878	0.93038	0.96456
23	2009-12-10	-0.0133489	0.6921457	75	-0.36795881	0.96385	0.98176
24	2009-12-27	-0.0290341	0.6385436	34	-0.44856526	0.93598	0.96746
25	2010-01-07	-0.0185737	0.635484	54	-0.45336839	0.88074	0.93848
26	2010-01-15	-0.0163126	0.6885109	61	-0.37322408	0.94977	0.97456
27	2010-01-18	-0.0116747	0.7755052	86	-0.25424063	0.94156	0.97034
28	2010-01-29	-0.0152983	0.8891791	65	-0.11745662	0.93579	0.96736
29	2010-02-11	-0.0173128	0.9208228	58	-0.08248766	0.97312	0.98647
30	2010-02-28	-0.0142994	1.0086819	70	0.00864448	0.99307	0.99653
31	2010-03-10	-0.0474953	1.1117011	21	0.10589138	0.95346	0.97645
32	2010-04-23	-0.0447802	0.0561421	22	-2.87986914	0.92228	0.96036
33	2010-06-03	-0.0497284	0.7755081	20	-0.25423688	0.94797	0.97364
34	2010-07-24	-0.0133696	0.948158	75	-0.0532341	0.97308	0.98645
35	2010-08-06	-0.0396091	1.2096216	25	0.19030756	0.95327	0.97635
36	2010-08-13	-0.0476916	1.1440893	21	0.13460893	0.97489	0.98736
37	2010-10-18	-0.0251393	1.5762692	40	0.45506076	0.99271	0.99635
38	2010-10-24	-0.0485708	1.5788481	21	0.45669554	0.92841	0.96354
39	2011-01-08	-0.0512634	0.9052901	20	-0.09949979	0.97309	0.98645
40	2011-01-20	-0.0127369	0.8141334	79	-0.20563109	0.95361	0.97653
41	2011-01-27	-0.0287361	0.8708551	35	-0.13827971	0.89749	0.94736
42	2011-01-30	-0.0140808	0.9367303	71	-0.06535989	0.89801	0.94764
43	2011-02-10	-0.0161217	0.566306	62	-0.56862072	0.95326	0.97635
44	2011-02-16	-0.0323589	0.684221	31	-0.37947428	0.97293	0.98637
PROMEDIO		-0.0254172	0.8280073	51			0.9704611

ANEXO E. ENVOLVENTE MÍNIMA DE CAUDALES



Estación el Conquistador	
Tr	Qmín
5	0.19631283
10	0.10080294
20	0.0500051
50	0.01988207

Para hallar la envolvente, primero se definió un periodo de 15 días como ventana con la cual se hace una búsqueda móvil de mínimos, luego se interpoló linealmente para obtener una serie de igual cantidad de datos que la serie original. Con $q(t)$ y q_0 , t y k , se halló el caudal mínimo para un período de retorno de 5, 10, 20 y 50 años. Para esto se ajustan los t a la distribución de probabilidades Lognormal.

ANEXO F. CÓDIGO FUENTE DE LA INTERFAZ GRÁFICA

```
function varargout = hysep(varargin)
% Baseflow filter M-file for baseflow filter.fig

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @hysep_OpeningFcn, ...
                  'gui_OutputFcn',  @hysep_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before hysep is made visible.
function hysep_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% Choose default command line output for hysep
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes hysep wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = hysep_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject     handle to figure
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject     handle to pushbutton1 (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)

% Se asegura que exista el archivo caudales.out sino, ejecuta el bflow.exe
% para generarlo.
if fopen('caudales.out') < 0
    system('bflow.exe') %Ejecuta
end
```

```

%Extrae todos los datos del archivo de entrada
[fech,val]=textread('caudales.prn','%s %f',-1);
%Extraer todos los datos del archivo de salida
[f,ad,s,b1,b2,b3,b4]=textread ('caudales.out','%s %f %f %f %f %f
%f','headerlines',2);
ad1(1,1)=0;
ad1(1,2)=0;
ad1(1,3)=0;
ad1(1,4)=0;
handles.fech=fech;
%Organiza el vector en las columnas correctas,debido a que cuando se leen
%los datos del archivo de salida, las columnas varíasn-
for i=1:length(fech)
    if length(f{i})==6
        ad1(i+1,1)=s(i);
        ad1(i+1,2)=b1(i);
        ad1(i+1,3)=b2(i);
        ad1(i+1,4)=b3(i);
    else
        if length(f{i})==8
            ad1(i+1,1)=ad(i);
            ad1(i+1,2)=s(i);
            ad1(i+1,3)=b1(i);
            ad1(i+1,4)=b2(i);
        else
            if ad(i)>0 && ad(i)<=9
                ad1(i+1,1)=b1(i);
                ad1(i+1,2)=b2(i);
                ad1(i+1,3)=b3(i);
                ad1(i+1,4)=b4(i);
            else
                ad1(i+1,1)=s(i);
                ad1(i+1,2)=b1(i);
                ad1(i+1,3)=b2(i);
                ad1(i+1,4)=b3(i);
            end
        end
    end
end
end
end

%Extrae datos de entrada y las posiciones de la matriz de acuerdo al
%periodo ingresado por el usuario y los guarda para luego graficarlos
n=1;
can=0;
cont=[0 0 0 0];
sum=[0 0 0 0];
a = get(handles.edit1,'String');
for m=1:length(fech)
    handles.pru(m)=strncmp(fech{m}, a, 4); %Verifica que la fecha corresponda a
la que ingresó el usuario y la guarda
    if handles.pru(m)==1
        handles.prov(n)=m; % Guarda las posiciones de las fechas que el usuario
ingresó
        can=can+1;
        grafic(n,1)=val(m); %Almacena el valor de la columna del hidrograma
(valores decimales)en una nueva matriz
        if val(m)==0
            cont(1)=cont(1)+1; %Cuenta el número de días que son ceros en el
hidrograma
        end
    end
end

```

```

        sum(1)=sum(1)+grafic(n,1); %Va sumando los valores de la columna decimal
del archivo de entrada
        grafic(n,2)=adl(m,2); %Guarda los valores de la columna flujo base1 en
una matriz
        if adl(m,2)==0
            cont(2)=cont(2)+1; %Cuenta días que son ceros en el archivo de
salida en la columna de flujo base1
        end
        sum(2)=sum(2)+grafic(n,2);
        grafic(n,3)=adl(m,3);
        if adl(m,3)==0
            cont(3)=cont(3)+1;
        end
        sum(3)=sum(3)+grafic(n,3);
        grafic(n,4)=adl(m,4);
        if adl(m,4)==0
            cont(4)=cont(4)+1;
        end
        sum(4)=sum(4)+grafic(n,4);
        %fecha(n)=str2num(fech{m});
        fecha(n)=n;
        n=n+1;
    end
end
handles.val=val;
%Verifica que el período ingresado por el usuario esté dentro del rango de
%los datos sino despliega un mensaje de error
if can==0 || isnan(str2double(a))==1
    msgbox('Para el año que ingresó no hay coincidencia en los
datos','Revisar','Error')
else
handles.grafic=grafic(:,:);
% Grafica el hidrograma y los flujos bases
hold(handles.hidrograma);
handles.p1=plot(handles.hidrograma,fecha(:),grafic(:,1),'r','LineWidth',2,'Marker
Size',4);
handles.p2=plot(handles.hidrograma,fecha(:),grafic(:,2),'g','LineWidth',2,'Marker
Size',4);
handles.p3=plot(handles.hidrograma,fecha(:),grafic(:,3),'k','LineWidth',2,'Marker
Size',4);
handles.p4=plot(handles.hidrograma,fecha(:),grafic(:,4),'b','LineWidth',2,'Marker
Size',4);
legend(handles.hidrograma,'Hidrograma','Paso 1','Paso 2','Paso 3')
% Calcula los volúmenes de flujo base
volesc=sum(1)*((fecha-cont(1))*86400);
volflu1=sum(2)*((fecha-cont(2))*86400);
volflu2=sum(3)*((fecha-cont(3))*86400);
volflu3=sum(4)*((fecha-cont(4))*86400);
porflu1=num2str((volflu1/volesc)*100);
porflu2=num2str((volflu2/volesc)*100);
porflu3=num2str((volflu3/volesc)*100);
%Imprime los resultados
set(handles.edit2,'Visible','on','string',porflu1);
set(handles.edit3,'Visible','on','string',porflu2);
set(handles.edit4,'Visible','on','string',porflu3);
set(handles.text4,'Visible','on');
set(handles.text5,'Visible','on');
set(handles.text6,'Visible','on');
end
guidata(hObject, handles);

```

```

function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit1 as text
%        str2double(get(hObject,'String')) returns contents of edit1 as a double
handle.input = get(hObject, 'String');

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes during object creation, after setting all properties.
function hidrograma_CreateFcn(hObject, eventdata, handles)
% hObject    handle to hidrograma (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: place code in OpeningFcn to populate hidrograma

function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit3 as text
%        str2double(get(hObject,'String')) returns contents of edit3 as a double

% --- Executes during object creation, after setting all properties.

```

```

function edit3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit4 as text
%         str2double(get(hObject,'String')) returns contents of edit4 as a double

% --- Executes during object creation, after setting all properties.
function edit4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hace invisible las cajas que contienen texto
set(handles.edit2,'Visible','off');
set(handles.edit3,'Visible','off');
set(handles.edit4,'Visible','off');
set(handles.text4,'Visible','off');
set(handles.text5,'Visible','off');
set(handles.text6,'Visible','off');
set(handles.edit1,'string','');
set(handles.edit5,'string','');
set(handles.edit6,'Visible','off');
set(handles.text9,'Visible','off');
set(handles.edit9,'Visible','off');
set(handles.text19,'Visible','off');
set(handles.edit10,'Visible','off');
set(handles.text20,'Visible','off');
% Borra todas las gráficas
hold(handles.hidrograma,'off')

```

```

delete(handles.p1)
delete(handles.p2)
delete(handles.p3)
delete(handles.p4)
% Hace invisible las leyendas de las gráficas.
legend(handles.hidrograma,'off')
% Borra la recta
delete(handles.rect)
a=1;
% Borra los tramos
while a<=length(handles.m)
delete(handles.m{a})
a=a+1;
end
% Borra variables para no afectar la graficacion
clear handles.grafic
clear fech
clear val
clear f
clear ad
clear b1
clear b2
clear b3
% guidata(hObject, handles);

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% Se asegura que se haya ejecutado el botón Graficar hidrograma antes de
% que se ejecute este botón
t = str2double(get(handles.edit2,'String'));
xx = get(handles.edit5,'String');
entr=str2double(xx);
if isnan(t)==1 && entr>3
    msgbox('Debe ejecutar primero el botón Graficar Hidrograma','Revisar','Error')
else
%Extrae las posiciones donde comienzan y terminan los tramos descendentes a
%través de puntos de inflexión, y se asegura que cada tramo se descende
%gracias al cálculo de la pendiente (negativa)
hold(handles.mrc,'off')
a=1;
limt=length(handles.prov)-3;
d=1;
while a<=limt
    b=a+1;
    c=b+1;
    if handles.val(handles.prov(a))>handles.val(handles.prov(b)) &&
handles.val(handles.prov(b))<handles.val(handles.prov(c))
        pos(d)=handles.prov(b);
        d=d+1;
    else
        if handles.val(handles.prov(a))<handles.val(handles.prov(b)) &&
handles.val(handles.prov(b))>handles.val(handles.prov(c))
            pos(d)=handles.prov(b);
            d=d+1;
        end
    end
    a=a+1;
end
q=1;
w=1;

```

```

contr=0;
% Evalua si los tramos son mayores a los requerimientos que pide el usuario
% y si los tramos son descendentes
while q<length(pos)
    r=q+1;
    if (handles.val(pos(q))-handles.val(pos(r)))>=0 && (pos(r)-pos(q))>entr
        desc(w)=pos(q);
        desc(i(w)=pos(r);
        w=w+1;
    else
        contr=contr+1;
    end
    q=q+1;
end
if (contr+1)==length(pos)
    msgbox('No existen tramos de ese tamaño, ingrese un valor
menor', 'Revisar', 'Error')
else
    % Almacena cada tramo en vectores
    qq=1;
    m=1;
    flag=0;
    axes(handles.mrc)
    nn=1;
    hh=1;
    pend=0;
    prep=0;
    entro=0;
    if flag==0
        jj=1;
        while jj<=length(descs)
            cont=descs(jj);
            n=1;
            % Guarda cada tramo en dos vectores
            while (cont<=desci(jj))
                y(n)=handles.val(cont);
                x(n)=nn;
                n=n+1;
                nn=nn+1;
                cont=cont+1;
            end
            ss=1;
            tam=length(y)-3;
            % Reduce el tramo a través de puntos de inflexión
            while ss<=tam
                dd=ss+1;
                ee=dd+1;
                if ((y(ss)-y(ee))/2)<(y(ss)-y(dd)) && ((y(dd)-y(ee))<((y(ss)-
y(ee))/2)
                    postram=ee;
                    ss=tam;
                else
                    postram=1;
                end
                ss=ss+1;
            end
            tt=1;
            while postram<=length(y)
                vecgray(tt)=log(y(postram));
                vecgrax(tt)=hh;

```

```

        postram=postram+1;
        tt=tt+1;
        hh=hh+1;
    end
    % Toma los tramos de recesión más representativos y la fecha de
    % inicio y fin
    lentramos(jj)=length(vecgray);
    if length(vecgray)>9
        prep=(vecgray(length(vecgray))-vecgray(1))/(length(vecgray));
        lenrec=length(vecgray);
        fecha=handles.fech{descs(jj)};
        entro=1;
    end

    % Guarda las posiciones de los tramos para después graficarlos
    pend=pend+vecgray(length(vecgray))-vecgray(length(vecgray)-1);
    ptoy(jj)=vecgray(length(vecgray));
    pttox(jj)=vecgray(length(vecgray));
    tramos{jj,1}=vecgray;
    tramos{jj,2}=vecgray;
    clear x
    clear y
    clear vecgray
    clear vecgray
    jj=jj+1;
end
% Calcula la pendiente
if entro==1
    pend=prep;
else
    pend=pend/length(descs);
end
if (jj-1)>1
    cptox=pttox;
    cptoxy=ptoxy;
    gg=1;
    while (gg<=length(ptoxy))
        jj=gg+1;
        while jj<=length(ptoxy)
            if cptoxy(gg)>cptoxy(jj)
                auxy=cptoxy(jj);
                auxx=cptox(jj);
                cptoxy(jj)=cptoxy(gg);
                cptox(jj)=cptox(gg);
                cptoxy(gg)=auxy;
                cptox(gg)=auxx;
            end
            jj=jj+1;
        end
        gg=gg+1;
    end
end
% Calcula el punto b, para poder graficar la recta
b=cptoxy(1)-(pend*cptox(1));
mm=1;
% Calcula la diferencia entre los puntos finales de cada tramo y de la
% recta, y luego los almacena
while mm<=length(ptoxy)
    py(mm)=pend*cptox(mm)+b;
    dif(mm)=py(mm)-cptoxy(mm);
    mm=mm+1;
end

```

```

end
% Grafica la recta que pasa por todos los puntos inferiores
handles.rect=plot(ptox(:),py(:),'r');
hold on
% Artilugio para dejar fijos los ejes
limx(1)=0;
limx(2)=0;
limx(3)=-4;
limy(1)=4;
limy(2)=-4;
limy(3)=0;
plot(limx(:),limy(:),'k.','LineWidth',0.1)
grid on
tramos{1,1}=0;
tramos{1,2}=0;
rr=1;
% Grafica cada uno de lo tramos sobre la recta agregando la diferencia
% calculada anteriormente para que los tramos pasen por la recta
while rr<=length(tramos)
    limtra=length(tramos{rr,2});
    vecaux=tramos{rr,2};
    kk=1;
    while kk<=limtra
        vecaux(kk)=vecaux(kk)+dif(rr);
        kk=kk+1;
    end
    tramos{rr,2}=vecaux;
    handles.m{rr}=plot(tramos{rr,1},tramos{rr,2},'b. ');
    rr=rr+1;
end
else
    vecauxx=tramos{1,1};
    vecauxy=tramos{1,2};
    pend=(vecauxy(length(vecauxy))-vecauxy(1))/length(vecauxy);
    grafic(1,1)=vecauxx(1);
    grafic(2,1)=vecauxx(length(vecauxx));
    grafic(1,2)=vecauxy(1);
    grafic(2,2)=vecauxy(length(vecauxx));
    handles.rect=plot(grafic(:,1),grafic(:,2),'r');
    hold on
    grid on
    handles.m{1}=plot(vecauxx,vecauxy,'b. ');
end

% Muestra los resultados obtenidos
set(handles.text19,'Visible','on');
set(handles.text20,'Visible','on');
set(handles.text9,'Visible','on');
set(handles.edit9,'Visible','on');
if entro==1
set(handles.edit9,'string',lenrec);
set(handles.edit10,'string',fecha);
else
    tem=sort(lentramos);
    dias=tem(length(tem));
    fecha=handles.fech{descs(length(tem))};
    set(handles.edit9,'string',dias);
    set(handles.edit10,'string',fecha);
end
set(handles.text10,'Visible','on');

```

```

set(handles.edit10,'Visible','on');
set(handles.edit6,'Visible','on');
set(handles.edit6,'string',pend);
end
end
end

guidata(hObject, handles);

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% zoom on

% --- Executes during object creation, after setting all properties.
function edit5_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit6_Callback(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit6_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% -----
functionopers_Callback(hObject, eventdata, handles)
% hObject    handle to opers (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% -----
function Oper1_Callback(hObject, eventdata, handles)
% Muestra los Cuadros Primer pantalla
set(handles.text18,'Visible','off');
set(handles.diasrec,'Visible','on');

```

```

set(handles.text16,'Visible','on');
set(handles.diasr,'Visible','on');
set(handles.text17,'Visible','on');
set(handles.edit7,'Visible','on');
set(handles.edit8,'Visible','on');
set(handles.grageneral,'Visible','on');
set(handles.hidrogeneral,'Visible','on');
set(handles.pushbutton7,'Visible','on');
set(handles.mrcgeneral,'Visible','on');
set(handles.mrcgral,'Visible','on');
set(handles.graftot,'Visible','on');

% Borrar Segunda pantalla
set(handles.uipanel1,'Visible','off');
set(handles.hidrograma,'Visible','off');
% set(handles.hidropeque,'Visible','off');
set(handles.mrc,'Visible','off');
set(handles.text13,'Visible','off');
set(handles.text2,'Visible','off');
set(handles.text14,'Visible','off');
set(handles.text11,'Visible','off');
set(handles.text15,'Visible','off');
set(handles.text3,'Visible','off');
set(handles.text4,'Visible','off');
set(handles.text5,'Visible','off');
set(handles.text6,'Visible','off');
set(handles.text9,'Visible','off');
set(handles.edit2,'Visible','off');
set(handles.edit3,'Visible','off');
set(handles.edit4,'Visible','off');
set(handles.edit6,'Visible','off');
set(handles.text10,'Visible','off');
set(handles.edit10,'Visible','off');

% -----
function Oper2_Callback(hObject, eventdata, handles)
% hObject    handle to Oper2 (see GCBO)

% Muestra los Cuadros segunda pantalla
set(handles.text18,'Visible','off');
set(handles.graftot,'Visible','off');
set(handles.uipanel1,'Visible','on');
set(handles.hidrograma,'Visible','on');
set(handles.mrc,'Visible','on');
set(handles.text13,'Visible','on');
set(handles.text2,'Visible','on');
set(handles.text14,'Visible','on');
set(handles.text11,'Visible','on');
set(handles.text15,'Visible','on');
set(handles.text3,'Visible','on');
% Borrar primera pantalla
set(handles.fechas,'Visible','off');
set(handles.fechatram,'Visible','off');
set(handles.text16,'Visible','off');
set(handles.text17,'Visible','off');
set(handles.edit7,'Visible','off');
set(handles.edit8,'Visible','off');
set(handles.grageneral,'Visible','off');
set(handles.mrcgeneral,'Visible','off');
set(handles.hidrogeneral,'Visible','off');

```

```

set(handles.pushbutton7,'Visible','off');
set(handles.diasr,'Visible','off');
set(handles.diasrec,'Visible','off');
set(handles.mrcgral,'Visible','off');
set(handles.edit16,'Visible','off');
set(handles.edit17,'Visible','off');
set(handles.edit18,'Visible','off');
% set(handles.text26,'Visible','off');
% set(handles.text27,'Visible','off');
% set(handles.text28,'Visible','off');
set(handles.expyteor,'Visible','off');
set(handles.edit19,'Visible','off','string','');
set(handles.text29,'Visible','on');

function edit7_Callback(hObject, eventdata, handles)
% hObject    handle to edit7 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit7_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit7 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in grageneral.
function grageneral_Callback(hObject, eventdata, handles)
% Extracción de datos
clc
% Verifica que el archivo caudales.out exista de lo contrario lo genera
if fopen('caudales.out')<0
    system('bflow.exe') %Ejecuta
end
% Extrae los datos del archivo caudales.prn y los guarda en dos vectores
% columna
[fech,val]=textread('caudales.prn','%s %f',-1);
handles.val=val;
handles.fech1=fech;
% Extrae los datos del archivo caudales.out y los guarda en vectores

```

```

% columna
[f,ad,s,b1,b2,b3,b4]=textread ('caudales.out','%s %f %f %f %f %f
%f','headerlines',2);
ad1(1,1)=0;
ad1(1,2)=0;
ad1(1,3)=0;
ad1(1,4)=0;
%Organiza el vector en las columnas correctas
for i=1:length(fech)
    if length(f{i})==6
        ad1(i+1,1)=s(i);
        ad1(i+1,2)=b1(i);
        ad1(i+1,3)=b2(i);
        ad1(i+1,4)=b3(i);
    else
    if length(f{i})==8
        ad1(i+1,1)=ad(i);
        ad1(i+1,2)=s(i);
        ad1(i+1,3)=b1(i);
        ad1(i+1,4)=b2(i);
    else
        if ad(i)>0 && ad(i)<=9
            ad1(i+1,1)=b1(i);
            ad1(i+1,2)=b2(i);
            ad1(i+1,3)=b3(i);
            ad1(i+1,4)=b4(i);
        else
            ad1(i+1,1)=s(i);
            ad1(i+1,2)=b1(i);
            ad1(i+1,3)=b2(i);
            ad1(i+1,4)=b3(i);
        end
    end
end
end
handles.prov1=0;
pru=0;
%Extraer datos de cada caja de texto donde ingresa el usuario el periodo
t1 = str2num(get(handles.edit7,'String'));
t2 = str2num(get(handles.edit8,'String'));
if t1<t2
    b1=get(handles.edit7,'String');
    b2=get(handles.edit8,'String');
    j=t1;
    q=1;
    u=1;
% Guarda en un vector el rango de fechas que ingresó el usuario y luego
% obtiene las posiciones de esas fechas para guardarlas en un vector.
while j<=t2
    periodo{u}=num2str(j);
    for m=1:length(fech)
        temp=strncmp(fech{m}, periodo{u}, 4);
        if temp==1
            pru(q)=temp;
            handles.prov1(q)=m;
            q=q+1;
        end
    end
    u=u+1;
    j=j+1;
end

```

```

    end
% Obtiene las posiciones de los datos de acuerdo al periodo establecido
can=0;
n=1;
cont=[0 0 0 0];
sum=[0 0 0 0];
m=1;
% Realiza el mismo procedimiento descrito anteriormente para el otro botón
% Grafic hidrog pero solo para las fechas que el usuario ingresó
while m<=length(fech)
    if m==handles.prov1(n) && n<length(handles.prov1)
        can=can+1;
        grafic(n,1)=val(m);
        if val(m)==0
            cont(1)=cont(1)+1;
        end
        sum(1)=sum(1)+grafic(n,1);
        grafic(n,2)=adl(m,2);
        if adl(m,2)==0
            cont(2)=cont(2)+1;
        end
        sum(2)=sum(2)+grafic(n,2);
        grafic(n,3)=adl(m,3);
        if adl(m,3)==0
            cont(3)=cont(3)+1;
        end
        sum(3)=sum(3)+grafic(n,3);
        grafic(n,4)=adl(m,4);
        if adl(m,4)==0
            cont(4)=cont(4)+1;
        end
        sum(4)=sum(4)+grafic(n,4);
        fecha(n)=n;
        n=n+1;
        m=m+1;
    else
        m=m+1;
    end
end
% Extrae información del archivo baseflow.dat y lo almacena
[fil,x1,x2,x3,x4,pen,x5]=textread('baseflow.dat','%s %f %f %f %f %f
%f','headerlines',3);
handles.pen=pen;
% Calcula los volúmenes de flujo base
volesc=sum(1)*((fecha-cont(1))*86400);
volflu1=sum(2)*((fecha-cont(2))*86400);
volflu2=sum(3)*((fecha-cont(3))*86400);
volflu3=sum(4)*((fecha-cont(4))*86400);
porflu1=num2str(roundn((volflu1/volesc),-2));
porflu2=num2str(roundn((volflu2/volesc),-2));
porflu3=num2str(roundn((volflu3/volesc),-2));
% Imprime los resultados en los cuadros correspondientes
if length(val)==length(handles.prov1)
    set(handles.edit11,'Visible','on','string',x1);
    set(handles.edit12,'Visible','on','string',x2);
    set(handles.edit13,'Visible','on','string',x3);
else
    set(handles.edit11,'Visible','on','string',porflu1);
    set(handles.edit12,'Visible','on','string',porflu2);
end

```

```

        set(handles.edit13,'Visible','on','string',porflu3);
    %     set(handles.edit19,'Visible','on','string',(-1*pen));
end
% set(handles.edit16,'Visible','on','string',x1);
% set(handles.edit17,'Visible','on','string',x2);
% set(handles.edit18,'Visible','on','string',x3);

set(handles.expyteor,'Visible','on');
set(handles.flujo1,'Visible','on');
set(handles.flujo2,'Visible','on');
set(handles.flujo3,'Visible','on');
% set(handles.text25,'Visible','on');
% set(handles.text26,'Visible','on');
% set(handles.text27,'Visible','on');
% set(handles.text28,'Visible','on');
handles.mrcgen=grafic;
% val=roundn(val,0);
handles.vall=log(val);
% Grafica el hidrograma y los flujos base
hold(handles.hidrogeneral);
handles.h1=plot(handles.hidrogeneral,fecha(:),grafic(:,1),'r','LineWidth',2,'MarkerSize',4);
handles.h2=plot(handles.hidrogeneral,fecha(:),grafic(:,2),'g','LineWidth',2,'MarkerSize',4);
handles.h3=plot(handles.hidrogeneral,fecha(:),grafic(:,3),'k','LineWidth',2,'MarkerSize',4);
handles.h4=plot(handles.hidrogeneral,fecha(:),grafic(:,4),'b','LineWidth',2,'MarkerSize',4);
legend(handles.hidrogeneral,'Hidrograma','Paso 1','Paso 2','Paso 3')
% Le permite al usuario obtener las posiciones de los puntos de los tramos
% más largos, dando él mismo clic sobre las secciones correspondientes.
grid on
    but = 'r';
    while but=='r'
        but=1;
        n = 0;
        xy = [];
        while but == 1
            [xi,yi,but] = ginput(1); %obtiene la coordenadas del punto
                if but==1           %but=1 significa que oprimo el boton
                    izquierdo
                        n = n+1;           %contador de numero de puntos
                        xy(:,n) = [xi;yi];%copia los puntos en el vector xy
                    end
                end % termina si oprime otra tecla diferente al click izquierdo
            end
        else
            msgbox('Para el año que ingresó no hay coincidencia en los
datos','Revisar','Error')
        end
        z=1;
        k=1;
        % Calcula las fechas de los tramos y los muestra en un listbox
        if mod(length(xy),2)==0
        while z<=length(xy)
            posfechaini=handles.prov1(roundn(xy(1,z),0));
            posfechafin=handles.prov1(roundn(xy(1,z+1),0));
            fechaini=fech{posfechaini};
            fechafin=fech{posfechafin};
            %     dif=num2str(posfechafin-posfechaini);

```

```

        vectfecha{k}=strcat(fechaini,'-',fechafin);
        k=k+1;
        z=z+2;
    end
    % Imrpime resultados
    set(handles.fechas,'Visible','on');
    set(handles.fechatram,'Visible','on');
    set(handles.fechas,'String',vectfecha);
    else
        msgbox('Los tramos tienen un punto de inicio y de fin','Revisar','Error')
    end
    clear x1
    clear x2
    clear x3
    clear fecha
    clear grafic
    clear fech
    clear val
    clear periodo
    clear t1
    clear t2
    % handles.prov1
    guidata(hObject, handles);

function edit9_Callback(hObject, eventdata, handles)
% hObject    handle to edit9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit9_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit9 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)

%Oculta cajas de texto
handles.prov1=0;
set(handles.edit11,'Visible','off');
set(handles.edit12,'Visible','off');
set(handles.edit13,'Visible','off');
set(handles.flujo1,'Visible','off');
set(handles.flujo2,'Visible','off');
set(handles.flujo3,'Visible','off');
set(handles.expyteor,'Visible','off');
set(handles.edit19,'Visible','off','string','');
set(handles.text29,'Visible','on');
set(handles.edit15,'Visible','off');
set(handles.edit16,'Visible','off');

```

```

set(handles.edit17,'Visible','off');
set(handles.edit18,'Visible','off');
set(handles.text25,'Visible','off');
set(handles.text26,'Visible','off');
set(handles.text27,'Visible','off');
set(handles.text28,'Visible','off');
set(handles.fechastram,'Visible','off');
set(handles.diasr,'string','');
% Borra todas las gráficas
delete(handles.h1)
delete(handles.h2)
delete(handles.h3)
delete(handles.h4)
delete(handles.rect)
legend(handles.mrcgeneral,'off')
legend(handles.hidrogeneral,'off')
set(handles.edit7,'string','');
set(handles.edit8,'string','');
hold(handles.hidrogeneral,'off')
for i=1:length(handles.m)
    delete(handles.m{i})
end
% for i=1:length(handles.m1)
%     delete(handles.m1{i})
% end
% clear

function edit10_Callback(hObject, eventdata, handles)
% hObject    handle to edit10 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit10_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit10 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit5_Callback(hObject, eventdata, handles)
% hObject    handle to edit5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

function edit8_Callback(hObject, eventdata, handles)
% hObject    handle to edit8 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

function edit11_Callback(hObject, eventdata, handles)

```

```

% hObject    handle to edit11 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit11_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit11 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit12_Callback(hObject, eventdata, handles)
% hObject    handle to edit12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit12_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit12 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit13_Callback(hObject, eventdata, handles)
% hObject    handle to edit13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function edit13_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit13 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in mrcgral.
function mrcgral_Callback(hObject, eventdata, handles)

```

```

% Hace los mismo que el botón para la extracción de los tramos
t = str2double(get(handles.edit11,'String'));
xx = get(handles.diasr,'String');
entr=str2double(xx);
if isnan(t)==1
    msgbox('Debe ejecutar primero el botón Graficar Hidrograma','Revisar','Error')
else
%Extrae las posiciones donde comienzan y terminan los tramos descendentes
hold(handles.mrc,'off')
a=1;
limt=length(handles.prov1)-3;
d=1;
while a<=limt
    b=a+1;
    c=b+1;
    if handles.vall(handles.prov1(a))>handles.vall(handles.prov1(b)) &&
handles.vall(handles.prov1(b))<handles.vall(handles.prov1(c))
        pos(d)=handles.prov1(b);
        d=d+1;
    else
        if handles.vall(handles.prov1(a))<handles.vall(handles.prov1(b)) &&
handles.vall(handles.prov1(b))>handles.vall(handles.prov1(c))
            pos(d)=handles.prov1(b);
            d=d+1;
        end
    end
    a=a+1;
end
q=1;
w=1;
contr=0;
% Evalua si los tramos son mayores a los requerimientos que pide el usuario
while q<length(pos)
    r=q+1;
    if (handles.vall(pos(q))-handles.vall(pos(r)))>=0 && (pos(r)-pos(q))>entr
        desc(w)=pos(q);
        desc(w)=pos(r);
        w=w+1;
    else
        contr=contr+1;
    end
    q=q+1;
end
if (contr+1)==length(pos)
    msgbox('No existen tramos de ese tamaño, ingrese un valor
menor','Revisar','Error')
else
qq=1;
m=1;
flag=0;
axes(handles.mrcgeneral)
nn=1;
hh=1;
pend=0;
prep=0;
entro=0;
if flag==0
    jj=1;
    while jj<=length(descs)
        cont=descs(jj);

```

```

n=1;
while (cont<=desci(jj))
    y(n)=handles.vall(cont);
    x(n)=nn;
    n=n+1;
    nn=nn+1;
    cont=cont+1;
end
ss=1;
tam=length(y)-3;
% Reduce el tramo
while ss<=tam
    dd=ss+1;
    ee=dd+1;
    if ((y(ss)-y(ee))/2)<(y(ss)-y(dd)) && ((y(dd)-y(ee))<((y(ss)-
y(ee))/2)
        postram=ee;
        ss=tam;
    else
        postram=1;
    end
    end
    ss=ss+1;
end
tt=1;
while postram<=length(y)
%   vecgray(tt)=log(y(postram));
    vecgray(tt)=y(postram);
    vecgrax(tt)=hh;
    postram=postram+1;
    tt=tt+1;
    hh=hh+1;
end
%   lenramos(jj)=length(vecgray);
if length(vecgray)>7
    prep=(vecgray(length(vecgray))-vecgray(1))/(length(vecgray));
end
hold on
grid on
%   Guarda las posiciones de los tramos para después graficarlos
pend=pend+vecgray(length(vecgray))-vecgray(length(vecgray)-1);
ptoy(jj)=vecgray(length(vecgray));
ptox(jj)=vecgrax(length(vecgrax));
tramos{jj,1}=vecgrax;
tramos{jj,2}=vecgray;
clear x
clear y
clear vecgrax
clear vecgray
jj=jj+1;
end
%   Calcula la pendiente
if entro==1
    pend=prep;
else
    pend=pend/length(descs);
end
if (jj-1)>1
    cptox=ptox;
    cptoy=ptoy;
    gg=1;

```

```

while (gg<=length(ptoy))
    jj=gg+1;
    while jj<=length(ptoy)
        if cptoym(gg)>cptoym(jj)
            auxy=cptoym(jj);
            auxx=cptomx(jj);
            cptoym(jj)=cptoym(gg);
            cptomx(jj)=cptomx(gg);
            cptoym(gg)=auxy;
            cptomx(gg)=auxx;
        end
        jj=jj+1;
    end
    gg=gg+1;
end
set(handles.text29,'Visible','on');
if length(handles.val)==length(handles.prov1)
    set(handles.edit19,'Visible','on','string',handles.pen*-1);
else
    set(handles.edit19,'Visible','on','string',pend);
end
b=cptoym(1)-(pend*cptomx(1));
mm=1;
while mm<=length(ptoy)
    py(mm)=pend*cptomx(mm)+b;
    dif(mm)=py(mm)-ptoy(mm);
    mm=mm+1;
end
handles.rect=plot(cptomx(:),py(:),'r');
hold on

% Artilugio para dejar fijos los ejes
limx(1)=0;
limx(2)=0;
limx(3)=-3;
limy(1)=4;
limy(2)=-3;
limy(3)=0;
plot(limx(:),limy(:),'k.','LineWidth',0.1)
grid on
tramos{1,1}=0;
tramos{1,2}=0;
rr=1;
% Grafica
while rr<=length(tramos)
    limtra=length(tramos{rr,2});
    vecaux=tramos{rr,2};
    kk=1;
    while kk<=limtra
        vecaux(kk)=vecaux(kk)+dif(rr);
        kk=kk+1;
    end
    tramos{rr,2}=vecaux;
% Para dar color a los tramos
if mod(rr,2)==0
    handles.m{rr}=plot(tramos{rr,1},tramos{rr,2},'b. ');
else
    handles.m{rr}=plot(tramos{rr,1},tramos{rr,2},'g. ');
end
rr=rr+1;

```

```

end
else
    vecauxx=tramos{1,1};
    vecauxy=tramos{1,2};
    pend=(vecauxy(length(vecauxy))-vecauxy(1))/length(vecauxy);
    grafic(1,1)=vecauxx(1);
    grafic(2,1)=vecauxx(length(vecauxx));
    grafic(1,2)=vecauxy(1);
    grafic(2,2)=vecauxy(length(vecauxx));
    handles.rect=plot(grafic(:,1),grafic(:,2),'r');
    hold on
    grid on
    handles.m{1}=plot(vecauxx,vecauxy,'b.');
```

```

end
end
end
end
clc
guidata(hObject, handles);

function diasr_Callback(hObject, eventdata, handles)
% hObject    handle to diasr (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes during object creation, after setting all properties.
function diasr_CreateFcn(hObject, eventdata, handles)
% hObject    handle to diasr (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit15_Callback(hObject, eventdata, handles)
% hObject    handle to edit15 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit15 as text
%       str2double(get(hObject,'String')) returns contents of edit15 as a double

% --- Executes during object creation, after setting all properties.
function edit15_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit15 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
```

```

end

function edit16_Callback(hObject, eventdata, handles)
% hObject    handle to edit16 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit16 as text
%        str2double(get(hObject,'String')) returns contents of edit16 as a double

% --- Executes during object creation, after setting all properties.
function edit16_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit16 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit17_Callback(hObject, eventdata, handles)
% hObject    handle to edit17 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit17 as text
%        str2double(get(hObject,'String')) returns contents of edit17 as a double

% --- Executes during object creation, after setting all properties.
function edit17_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit17 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit18_Callback(hObject, eventdata, handles)
% hObject    handle to edit18 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit18 as text
%        str2double(get(hObject,'String')) returns contents of edit18 as a double

```

```

% --- Executes during object creation, after setting all properties.
function edit18_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit18 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit19_Callback(hObject, eventdata, handles)
% hObject    handle to edit19 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit19 as text
%         str2double(get(hObject,'String')) returns contents of edit19 as a double

% --- Executes during object creation, after setting all properties.
function edit19_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit19 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in fechas.
function fechas_Callback(hObject, eventdata, handles)
% hObject    handle to fechas (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns fechas contents as
cell array
%         contents{get(hObject,'Value')} returns selected item from fechas

% --- Executes during object creation, after setting all properties.
function fechas_CreateFcn(hObject, eventdata, handles)
% hObject    handle to fechas (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```