

Manual Completo de Uso de VFLUX 2 en MATLAB

◆ 1. Qué es VFLUX 2 y para qué sirve

VFLUX 2 es una herramienta desarrollada en MATLAB para estimar **flujos verticales de agua subterránea** (infiltración o exfiltración) a partir de **registros de temperatura** tomados en diferentes profundidades.

Se usa comúnmente en ríos, lagunas o humedales donde se instalan sensores (termocuplas o termistores) a distintas profundidades en el lecho.

El principio es simple:

- El calor se propaga hacia abajo desde la superficie de manera diaria (onda térmica diurna).
- Si el agua se infiltra, la señal térmica se atenúa más rápido y se retrasa.
- Si el agua exfiltra, la onda térmica se adelanta o amplifica.

VFLUX analiza esas diferencias entre sensores y estima **la velocidad y dirección del flujo de agua**.

◆ 2. Instalación paso a paso

■ Paso 1: Descargar los archivos necesarios

1. Descarga **VFLUX 2** desde el repositorio oficial (normalmente viene en formato ZIP).

Link:

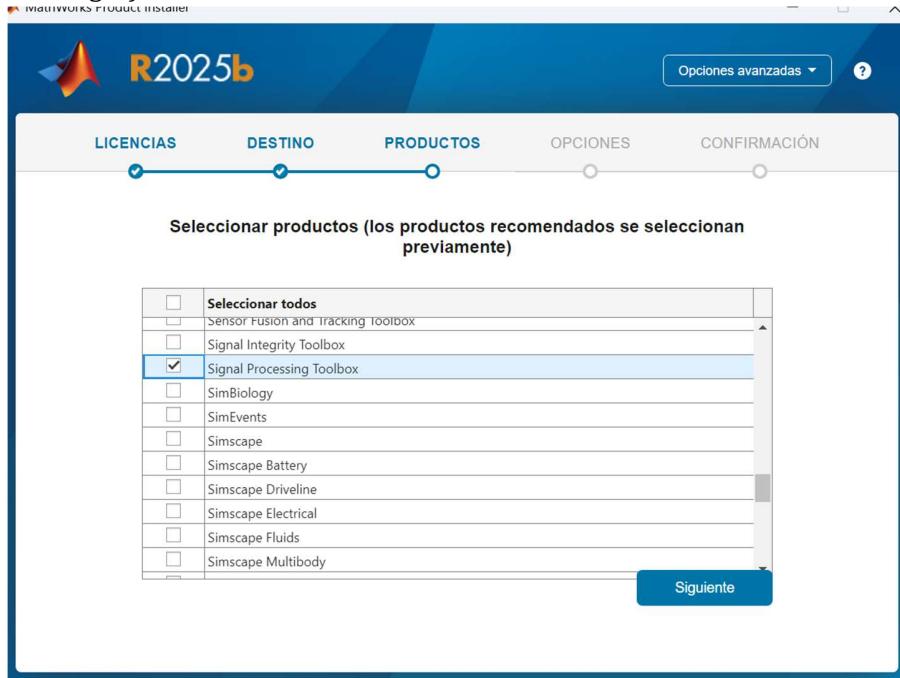
<https://www.hydroshare.org/resource/4df337867d314620bd87b27c6732e6fe/>

2. Descarga también **Captain Toolbox** desde el MATLAB File Exchange (necesario para la función arspec, que analiza la señal térmica).

Link: <https://la.mathworks.com/matlabcentral/fileexchange/173965-captain-toolbox>

3. Verifica que MATLAB tenga instalado el **Signal Processing Toolbox** (usa el comando ver en MATLAB y busca "Signal Processing Toolbox" en la lista al

descargar).



■ Paso 2: Colocar las carpetas

Guarda las carpetas vflux2 y captain en un lugar fácil, por ejemplo:

G:\Unidades compartidas\03.2 Proyectos MLP3\MLP-OT010_MC y MN Tranque Quillayes\03.WIP\02.Hidrología

■ Paso 3: Agregar al "path" de MATLAB

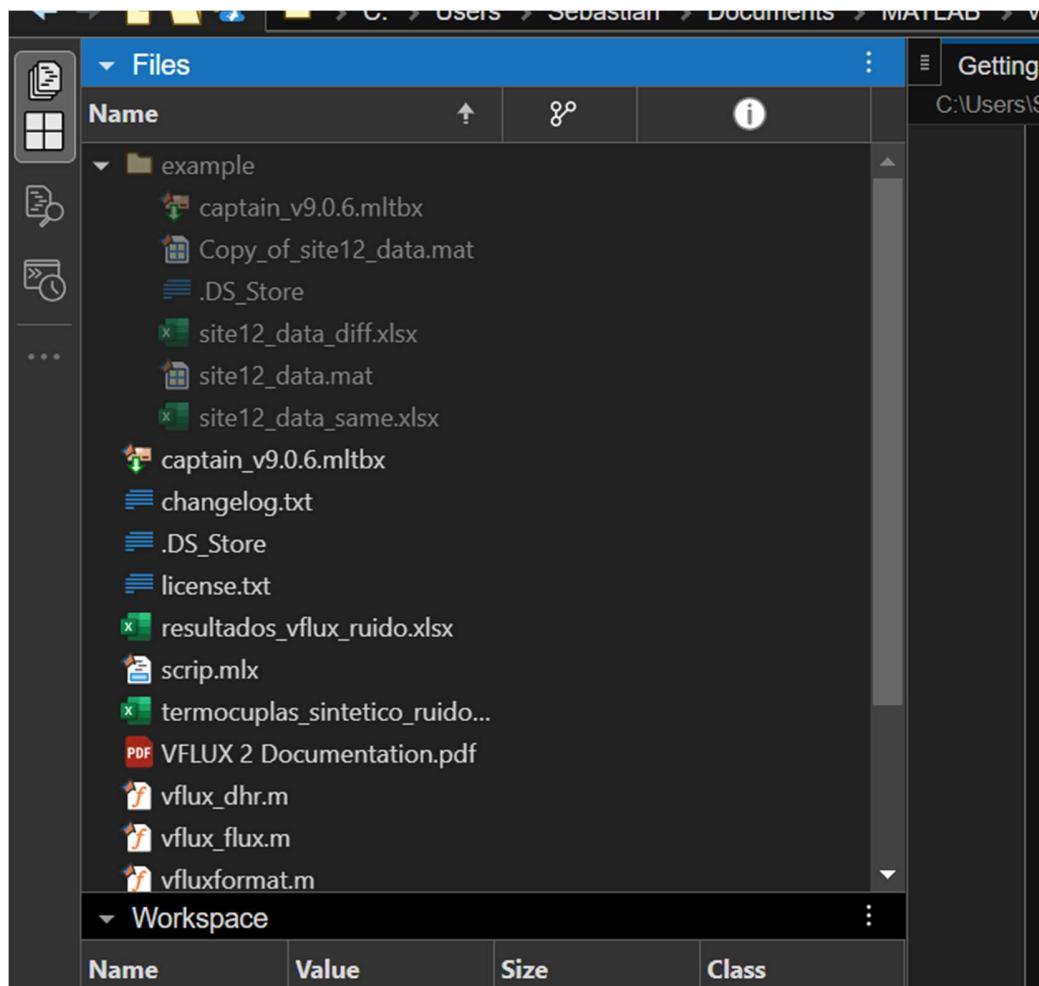
Abre MATLAB y escribe:

% === Agregar carpetas de VFLUX y Captain Toolbox ===
en el Command Window

A screenshot of the MATLAB Command Window. The window title is 'Command Window'. The text in the window shows the user entering the command 'addpath('G:\Unidades compartidas\03.2 Proyectos MLP3\MLP-OT010_MC y MN Tranque Quillayes\03.WIP\02.Hidrología\vflux2')'. The command is partially cut off at the end. The window has a dark theme with a blue header bar.

addpath('C:\Users\Sebastián\Desktop\Tutorial')

Por otro lado, el archivo de “Captain” hay que directamente arrastrarlo a la pestaña files de matlab y luego hacer doble click arriba del icono



% === Guardar el path para que se cargue automáticamente al iniciar MATLAB ===

Savepath

Para saber si esta está correcto se debe escribir: pwd en el coomand window y debe mostrar:

pwd

<img alt="Screenshot of the MATLAB Command Window showing the output of the 'pwd' command. The window displays the following text:<pre>Command Window
 Invalid use of operator.
>> addpath('C:\Users\Sebastián\Desktop\Tutoriales')
>> pwd
ans =
 'C:\Users\Sebastián\Documents\MATLAB\vflux2'
>>

◆ 3. Preparar tus datos

Tus datos deben estar en un **archivo Excel (.xlsx)** con las fechas y temperaturas registradas por cada sensor. En la Carpeta Vflux2

Cada sensor tiene su **columna de fecha y de temperatura**.

Ejemplo de estructura:

fecha1	temp1	fecha2	temp2	fecha3	temp3
01/01/2025 00:00	20.5	01/01/2025 00:02	19.9	01/01/2025 00:04	19.1
01/01/2025 00:30	21.0	01/01/2025 00:32	20.4	01/01/2025 00:34	19.6

💡 Nota: si tus sensores comenzaron a medir con unos minutos de diferencia (por ejemplo, 2 minutos de desfase), eso está bien. VFLUX puede manejarlo automáticamente.

Se deben ordenar del ibutton más superficial a la más profunda.

Guarda el archivo, por ejemplo, como:

termocuplas_Numerotermocupla_Fecha.xlsx

◆ 4. Leer y preparar los datos en MATLAB

En esta sección hay que copiar y pegar cada línea y pegarla en el command window de matlab, luego apretar enter

Copia este bloque en MATLAB y ejecútalo:

clc; clear; close all;

```
% === Leer archivo ===
```

```
archivo ="ruta excel"
```

```
data = readtable(archivo);
```

```
% === Convertir fechas ===
t1 = datenum(data.fecha1);
t2 = datenum(data.fecha2);
t3 = datenum(data.fecha3);

% === Temperaturas ===
temp1 = data.temp1;
temp2 = data.temp2;
temp3 = data.temp3;

% === Profundidades de los sensores (en metros) ===
vdata = vfluxformat(t1, temp1, 0.05, t2, temp2, 0.10, t3, temp3, 0.40);

% Sensor a 5 cm... % Sensor a 10 cm% Sensor a 20 cm...
Stos valores de medida es la profundidad del ibutton desde el lecho del rio en metros. Estos
valores dependen de cada ibutton y termocupla
```

👉 Este bloque convierte tus datos en un formato que VFLUX puede entender, corrigiendo el desfase temporal entre sensores.

◆ 5. Ejecutar el cálculo del flujo

Ahora corre el modelo:

```
results = vflux(vdata, 0, [1 2], 1, 0.3, 0.001, 0.0045, 0.5, 1, 'unattended');
```

Cada parámetro significa:

Parámetro	Qué hace	Valor recomendado
rfactor	Factor de reducción del muestreo (0 = automático)	0
windows	Pares de sensores a analizar	[1 2]

Parámetro	Qué hace	Valor recomendado
Pf	Filtro espectral	1
n	Número de armónicos	0.3
beta	Dispersividad térmica (m)	0.001
Kcal	Conductividad térmica del sedimento	0.0045
Cscal	Calor específico del sedimento	0.5
Cwcal	Calor específico del agua	1

◆ 6. Entendiendo el parámetro windows

El parámetro windows define **qué sensores se comparan entre sí**:

- [1] → compara sensores adyacentes (superior con intermedio, intermedio con inferior).
- [2] → compara el superior con el inferior.
- [1 2] → hace ambos cálculos y te da más detalle.

👉 Si solo tienes **3 sensores**, windows = [2] te da un flujo más **estable**. Si estás probando con datos sintéticos o limpios, usa [1 2] para comparar resultados entre capas.

(Ver Figura 1: esquema de pares de sensores)

◆ 7. Ver y guardar los resultados

Después de correr el modelo, los resultados quedan guardados en una variable llamada results.

Los principales campos son:

Campo	Qué representa	Unidad
results.fluxm	Flujo (m/s) según método McCallum	m/s
results.fluxha	Flujo por método Hatch (amplitud)	m/s

Campo	Qué representa	Unidad
results.fluxhp	Flujo por método Hatch (fase)	m/s
results.fluxk	Flujo según Keery	m/s
results.fluxl	Flujo según Luce	m/s
results.dtime	Tiempo en días MATLAB	-

◆ 8. Exportar los resultados a Excel

```
fecha_real = datetime(results.dtime, 'ConvertFrom', 'datenum');

flujo_m_s = results.fluxm;

flujo_mm_d = flujo_m_s * 1000 * 86400;

tabla = table(fecha_real, flujo_m_s, flujo_mm_d);

writetable(tabla, 'resultados_vflux_ruido.xlsx');
```

Esto crea un Excel con tres columnas:

- Fecha y hora
- Flujo en m/s
- Flujo en mm/día

(Ver Figura 2: ejemplo de tabla de resultados)

◆ 9. Graficar y comparar métodos

Copia y ejecuta este bloque:

```
figure('Name','Comparación entre métodos VFLUX');

subplot(5,1,1); plot(fecha_real, results.fluxm, 'b'); title('McCallum');

subplot(5,1,2); plot(fecha_real, results.fluxha, 'r'); title('Hatch (Amplitud)');

subplot(5,1,3); plot(fecha_real, results.fluxhp, 'm'); title('Hatch (Fase)');
```

```

subplot(5,1,4); plot(fecha_real, results.fluxk, 'g'); title('Keery');

subplot(5,1,5); plot(fecha_real, results.fluxl, 'c'); title('Luce');

xlabel('Tiempo'); ylabel('Flujo (m/s)');

```

(Ver Figura 3: comparación de métodos de flujo)

◆ 10. Cómo interpretar los resultados

Método	Qué usa	Características	Recomendación
McCallum	Amplitud + fase	Más estable, menos sensible al ruido	<input checked="" type="checkbox"/> Método principal
Hatch-Amp	Amplitud	Rápido, pero más sensible al ruido	Comparativo
Hatch-Fase	Fase	Menos estable con ruido	Diagnóstico
Keery	Difusividad corregida	Estima mejor flujos lentos	Complementario
Luce	Empírico	Más simple, útil como referencia	Secundario

◆ 11. Interpretación física del flujo

Signo del flujo	Significado físico	Ejemplo
Negativo (-)	Infiltración — el agua baja desde la superficie al acuífero	Río infiltrando agua al subsuelo
Positivo (+)	Exfiltración — el agua sube desde el acuífero hacia la superficie	Descarga subterránea hacia el río

(Ver Figura 4: interpretación de flujos positivos y negativos)

◆ 12. Consejos finales

- Asegúrate de que las profundidades sean precisas y en metros.

- Los sensores deben tener un muestreo constante (p. ej. cada 30 minutos).
- Si hay ruido excesivo, usa windows = [2] y revisa las curvas de temperatura.
- Para validaciones, compara el flujo medio diario con condiciones hidrológicas observadas (nivel de agua, lluvias, etc.).

1. rfactor — Factor de reducción del muestreo

Qué hace

VFLUX necesita analizar el ciclo térmico diario (la variación de temperatura cada 24 horas). Si tus datos tienen un paso muy corto (por ejemplo, cada 1, 5 o 10 minutos), tendrás **demasiadas muestras por día**, lo que puede:

- hacer más lento el cálculo,
- y aumentar el ruido.

El parámetro **rfactor** controla **cuántas muestras por día se usarán en el análisis**.

Cómo funciona

Cuando defines rfactor = 0, VFLUX **calcula automáticamente** un factor para dejar alrededor de **12 muestras por día** (una cada 2 horas aprox.).

Esto es suficiente para representar la forma de la onda térmica sin sobrecargar el modelo.

Si defines un número distinto, por ejemplo rfactor = 12, MATLAB tomará **1 de cada 12 puntos** del registro.

Recomendaciones

Tipo de datos	Paso temporal original	Valor sugerido de rfactor
Datos limpios cada 30 min	48 muestras/día	0 (automático)
Datos cada 10 min o menos	144 muestras/día	12
Datos ruidosos o incompletos	Variable	0 (deja que VFLUX lo ajuste)

Efecto visual (ver Figura 11)

- rfactor = 0 → Señal completa, alta resolución.

- rfactor = 12 → Señal suavizada, menos ruido.
 - rfactor muy alto (>20) → Pérdida de detalle del ciclo diario.
-

2. windows — Pares de sensores a analizar

Qué hace

Define **qué combinaciones de sensores** se usan para calcular el flujo térmico entre ellos. Se expresa en **intervalos** entre sensores, no en metros.

Por ejemplo, si tienes sensores a 5, 10 y 20 cm:

Valor de windows	Pares analizados	Qué representa
[1]	(5–10 cm) y (10–20 cm)	Flujos locales entre sensores adyacentes
[2]	(5–20 cm)	Flujo promedio global
[1 2]	Todos los anteriores	Comparación entre flujo local y total

Recomendaciones

Tipo de análisis	Valor sugerido
Datos sintéticos o experimentales [1 2] para comparar	
Datos de campo con ruido	[2] para estabilidad
Más de 4 sensores	1:n-1 según resolución vertical deseada

Consejos prácticos

- Valores más pequeños (como [1]) dan más **resolución vertical**, pero pueden generar **flujos erráticos** si el gradiente térmico es bajo.
- Valores más grandes (como [2]) suavizan y representan mejor el **flujo promedio**.

(Ver Figura 12: esquema de pares de sensores para diferentes valores de windows)

3. Pf — Filtro espectral

Qué hace

El parámetro **Pf** controla **qué parte del espectro de frecuencias de la señal térmica se conserva** durante el análisis.

En otras palabras, determina cuánto del “ruido” se elimina antes de calcular el flujo.

Las señales térmicas reales pueden tener:

- una onda diaria (que es la que nos interesa),
- variaciones lentas (por clima o sombra),
- y picos rápidos (ruido o errores de sensor).

El filtro espectral elimina estas últimas componentes indeseadas.

Cómo funciona

- Si $Pf = 1$: se usa todo el rango de frecuencias → sin filtro.
 - Si $Pf = 0.8$: se eliminan frecuencias muy altas → señal más suave.
 - Si $Pf = 0.5$: se conservan solo las componentes más lentas → mayor filtrado.
-

Recomendaciones

Tipo de señal	Características	Valor recomendado de Pf
Datos sintéticos o muy limpios	Sin ruido o error de medición	1.0
Datos reales con pequeñas fluctuaciones	Termocuplas bien calibradas	0.8
Datos muy ruidosos o con fallos	Señal inestable	0.5–0.7

Efecto visual (ver Figura 13)

- $Pf = 1$: mantiene la señal original (más detalle, más ruido).

- $P_f = 0.8$: filtra el ruido manteniendo la forma.
 - $P_f = 0.5$: señal muy suavizada (puede perder amplitud).
-

4. n — Número de armónicos

Qué hace

El parámetro **n** define cuántos **componentes de frecuencia** (armónicos) usa el análisis de regresión armónica (arspec) para ajustar la señal térmica.

Cada “armónico” representa una oscilación periódica:

- el primer armónico → ciclo diario (24 horas),
- el segundo → semidiario (12 horas),
- el tercero → 8 horas, etc.

El valor $n = 0.3$ significa que VFLUX ajustará principalmente **la componente diurna**, ignorando variaciones de menor período.

Interpretación simple

- n no es un número de ciclos, sino un **factor relativo**.
 - Un n pequeño (<0.5) prioriza la señal diaria.
 - Un n mayor incluye más frecuencias, aumentando el detalle pero también el ruido.
-

Recomendaciones

Tipo de análisis	Valor sugerido de n
------------------	----------------------------

Datos con buena resolución temporal 0.3 (valor por defecto)

Datos muy suaves o promedios diarios 0.2

Datos con variaciones rápidas 0.5 o más

(Ver Figura 14: ejemplo del efecto de incluir más armónicos en el ajuste de la señal térmica)

Resumen general

Parámetro	Qué controla	Efecto principal	Valor recomendado
rfactor	Frecuencia de muestreo	Reduce datos y ruido	0 (automático)
windows	Pares de sensores	Define resolución vertical [1 2]	
Pf	Filtrado espectral	Suaviza la señal	1
n	Complejidad armónica	Ajusta el detalle temporal	0.3

Interpretación combinada:

- rfactor y Pf afectan **la suavidad y el ruido** de la señal.
- windows define **dónde se mide** el flujo (entre qué sensores).
- n controla **qué parte de la señal térmica** se usa para estimar el flujo.

Cuando se usan juntos con los valores sugeridos (rfactor=0, windows=[1 2], Pf=1, n=0.3), el resultado es estable y físicamente representativo para la mayoría de los casos reales.