

Übungen zu imc FAMOS I – Digital Kurs

- Block 4 -

Doc. Rev.: 1.1- 25.08.2025



Gezielter Wissenstransfer – höhere Produktivität

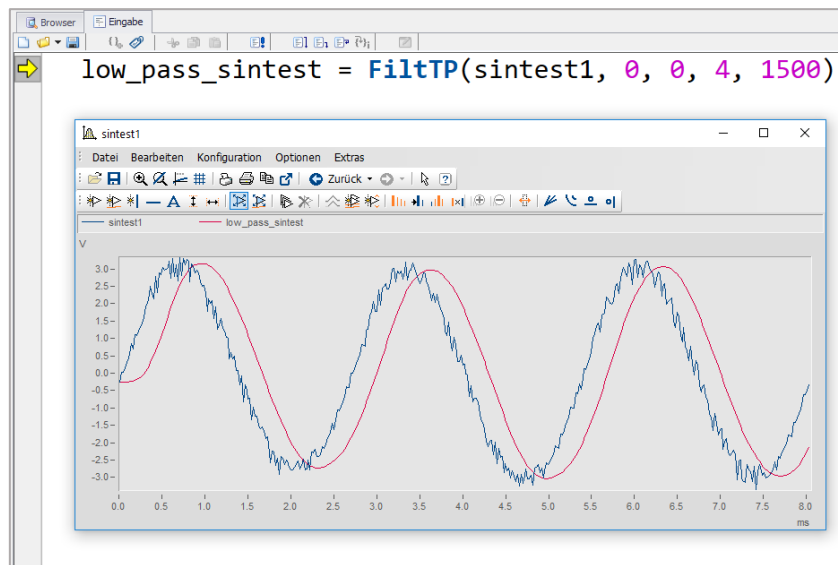
Übung A

Übungsziel:

Anwendung einer Funktion aus der Funktionsbibliothek auf einen Datensatz am Beispiel des Tiefpassfilters.

Ergebnis:

Im folgenden Bild ist die auszuführende Zeile im Eingabe-Bereich zu sehen. Das frei fliegenden Kurvenfenster zeigt den Ursprungsdatensatz sowie den gefilterten Datensatz.



Übungsschritte:

- Zum Ausführen von Funktionen müssen diese mit ihrer korrekten Syntax im Eingabefenster eingegeben werden. Geben Sie die Zeile:
`low_pass_sintest = FiltLP(sintest1, 0, 0, 4, 1500)`
 direkt mit der Tastatur im Eingabefenster ein und führen Sie diese mit der Tastenkombination „Strg+Return“ aus. Alternativ navigieren Sie in der Funktionsbibliothek zur Gruppe „Bearbeitung“ und ziehen den Kanal „sintest1“ mittels Drag & Drop auf die Funktion „Tiefpass-Filter“ in der Untergruppe „Filter“.



- Daraufhin öffnet sich der Eingabeassistent, mit dessen Hilfe Sie die Parameter der Funktion wie gewünscht konfigurieren können. Anschließend kann durch Klick auf „Kopieren“ die entsprechend konfigurierte Zeile in das Eingabefenster kopiert und ausgeführt werden. Die Ausführung kann zudem direkt im Eingabeassistenten durch Klick auf „Ausführen“ durchgeführt werden.

Zur Vertiefung:

- Erzeugen Sie einen weiteren Datensatz, indem Sie die Funktion „Smo5()“, welche ebenfalls das Rauschen unterdrückt, zweimal auf den Kanal „sintest1“ anwenden.

$gla_sintest1 = \text{Smo5}(\text{Smo5}(\text{sintest1}))$

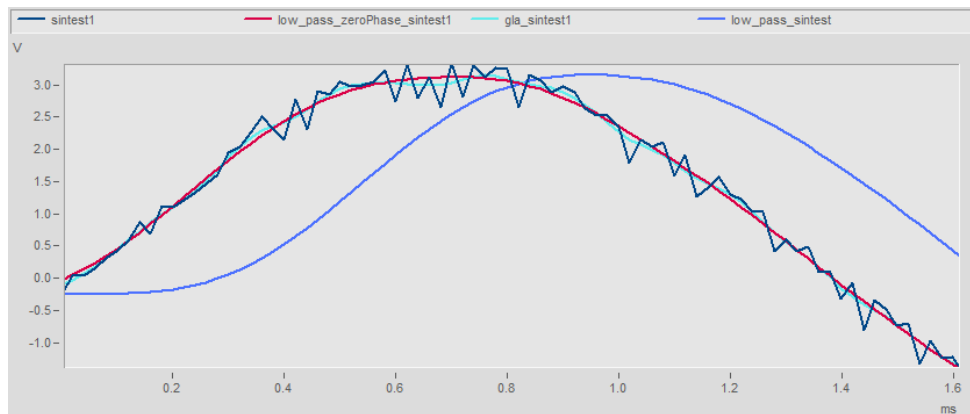
- Vergleichen Sie diesen mit dem Ergebnis des Tiefpass-Filters, indem Sie alle Datensätze in einem gemeinsamen Kurvenfenster anzeigen.

- Falls Sie über eine FAMOS Lizenz der Edition „Professional“ oder höher verfügen, stehen Ihnen zusätzlich ZeroPhase Filter zur Verfügung. Erstellen Sie einen weiteren gefilterten Datensatz unter Verwendung des ZeroPhase Tiefpass-Filters.

$low_pass_zeroPhase_sintest1 = \text{FiltLPZ}(\text{sintest1}, 0, 0, 4, 1500)$

- Vergleichen Sie anschließend auch dieses Ergebnis mit den vorherigen (bezüglich der Phase, Rest-Rauschen usw.)

Das Ergebnis sollte etwa wie folgt aussehen:



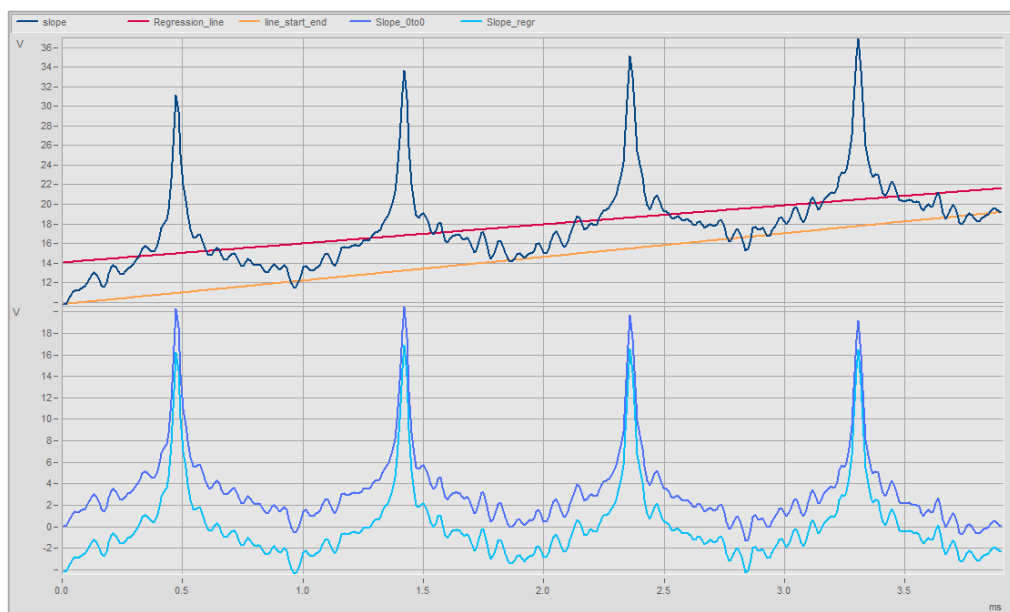
Übung B

Übungsziel:

An einem Datensatz soll ein linearer Trend eines Datensatzes korrigiert werden. Dieses Ziel wird in der Übung auf zwei verschiedene Arten realisiert. Einmal kommt die Regressionsgerade des Datensatzes zur Anwendung, im zweiten Fall werden einfach die Start- und Endpunkte zur Bestimmung einer Geraden zur Trendkorrektur verwendet, indem diese jeweils vom Originaldatensatz abgezogen werden.

Ergebnis:

Das folgende Kurvenfenster zeigt den Ausgangsdatsatz, die erstellten Hilfsgeraden sowie die Ergebnisse nach der Trendkorrektur.



Übungsschritte:

Die folgenden Zeilen erzeugen die oben dargestellten Datensätze:

1. Methode: Verwendung der Regressionsgeraden

- Laden Sie den Datensatz „slope“ aus den Beispieldatensätzen.
- Erstellen Sie mit Hilfe der Funktion „Regr()“ eine Regressionsgerade von „slope“.
- Um die Regressionsgerade zur Trendkorrektur zu verwenden, muss diese die gleiche Samplerate haben wie der Originaldatensatz. Verwenden Sie die Funktion „RSamp()“, um die Samplerate entsprechend anzupassen.
- Berechnen Sie den korrigierten Datensatz, indem Sie die Regressionsgerade vom Originaldatensatz subtrahieren.

Lösungshinweis: Der entstandene Code im Eingabefenster sollte in etwa wie folgt aussehen:

```
Regression_Line = Regr(slope)
Regression_Line = RSamp(Regression_Line , slope)
Slope_regr = slope - Regression_Line
```

2. Methode: Verwendung der Start- und Endpunkte des Originaldatensatzes.

- Laden Sie den Datensatz „slope“ aus den Beispieldatensätzen.
- Erstellen Sie einen neuen Hilfsdatensatz, der den ersten sowie den letzten y-Wert des Datensatzes „slope“ enthält. Tipp: Auf den Wert eines Datensatzes an einer bestimmten Indexposition kann direkt mit „daten[index]“ zugegriffen werden.
- Erstellen Sie einen neuen Hilfsdatensatz, der den ersten sowie den letzten x-Wert des Datensatzes „slope“ enthält. Tipp: Für die Bestimmung der x-Werte kann der Offset des Datensatzes verwendet werden.
- Unter Verwendung der beiden Hilfsdatensätze erstellen Sie nun einen neuen Datensatz, der die Korrekturgerade repräsentiert. Verwenden Sie dazu die Funktion „XYof()“.
- Um den Korrekturdatensatz vom Datensatz „slope“ subtrahieren zu können, müssen beide Datensätze die gleiche Abtastrate haben. Führen Sie daher mit der Funktion „RSamp()“ ein Resampling der Korrekturgeraden durch, indem Sie den Datensatz „slope“ als Referenz verwenden.
- Führen Sie die Trendkorrektur durch, indem Sie den Korrekturdatensatz vom Datensatz „slope“ subtrahieren.
-

Lösungshinweis: Der entstandene Code im Eingabefenster sollte in etwa wie folgt aussehen:

```
y_werte = [slope[1], slope[Leng?(slope)]]
x_werte = [(xOff?(slope), xOff?(slope) + (Leng?(slope)-
1)*xDel?(slope)]
line_start_end = XYof(x_werte, y_werte)
line_start_end = RSamp(line_start_end , slope )
Slope_0to0 = slope - line_start_end
```

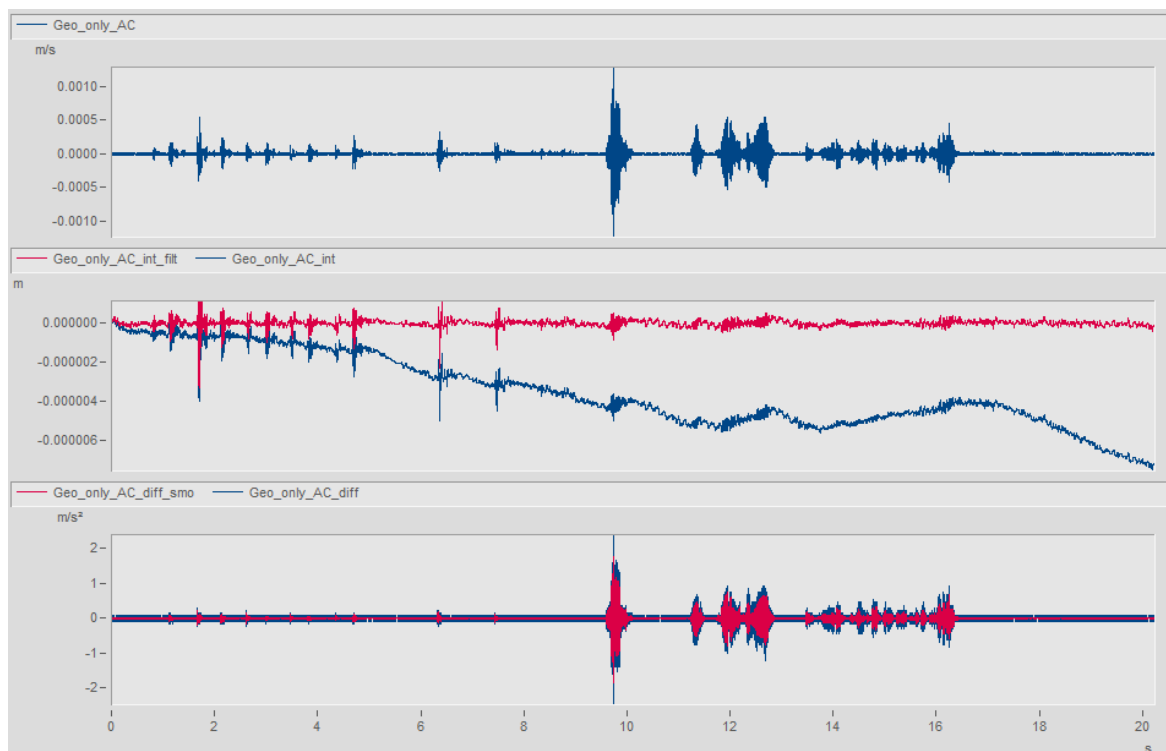
Übung C

Übungsziel:

Diese Übung behandelt wie man häufige Probleme bei der Integration und Ableitung erkennen und beheben kann. Zur Veranschaulichung wird dabei aus einer gemessenen Schwinggeschwindigkeit die Schwingbeschleunigung sowie der Schwingweg berechnet.

Ergebnis:

Das folgende Kurvenfenster zeigt den Originaldatensatz sowie die integrierten und differenzierten Signale vor und nach ihrer Korrektur.



Übungsschritte:

- Laden Sie den Datensatz „Geo_only_AC“ aus den Beispieldatensätzen und zeigen Sie diesen in einem Kurvenfenster an. Das Signal wurde mit einem Geophon aufgezeichnet, welches ein ähnliches Verhalten wie ein IEPE-Sensor für die Beschleunigungsmessung (piezoelektrischer Sensor) zeigt. Diese Sensoren erfassen keine statischen oder sehr langsamen Änderungen

Integration

- Integrieren Sie den Datensatz mit Hilfe der Funktion „Int()“ aus der Gruppe „Grundfunktionen / Integration Differentiation“ und zeigen Sie das Ergebnis ebenfalls im Kurvenfenster an.

$\text{Geo_only_AC_int} = \text{int}(\text{Geo_only_AC})$

- ➔ Ein auf dem Datensatz vorhandener Offset wird durch die Integration aufsummiert und erzeugt so eine Drift in der Integration

- Um das fälschliche Aufsummieren des Offsets zu unterdrücken, wenden Sie entweder vor oder nach der Integration einen Hochpassfilter mit sehr niedriger Grenzfrequenz (z.B. 1 Hz) an.

$Geo_only_AC_int_filt = FiltHpZ(Geo_only_AC_int, 0, 0, 2, 1)$

- Fügen Sie den gefilterten Datensatz ebenfalls zum Kurvenfenster hinzu. Variieren Sie die Grenzfrequenz des Hochpassfilters und achten Sie auf die Veränderungen im Ergebnisdatensatz.

Differentiation (Ableitung)

- Differenzieren Sie den Datensatz „Geo_only_AC“ mit der Funktion „Diff()“ und fügen Sie das Ergebnis erneut dem Kurvenfenster hinzu

$Geo_only_AC_diff = diff(Geo_only_AC)$

➔ Ein auf dem Datensatz vorhandenes Rauschen wird durch die Differenzierung verstärkt

- Um die Verstärkung des Rauschens zu unterdrücken, wenden Sie entweder vor oder nach der Differentiation einen der Glättungsfilter (Smo3(), Smo5() oder Smo()) an.

$Geo_only_AC_diff_smo = Smo(Geo_only_AC_diff, 0.002)$

- Fügen Sie das geglättete Ergebnis zum Kurvenfenster hinzu. Variieren Sie die Glättungsfunktionen bzw. das Glättungsintervall und beobachten Sie die Rauschunterdrückung im Ergebnisdatensatz.

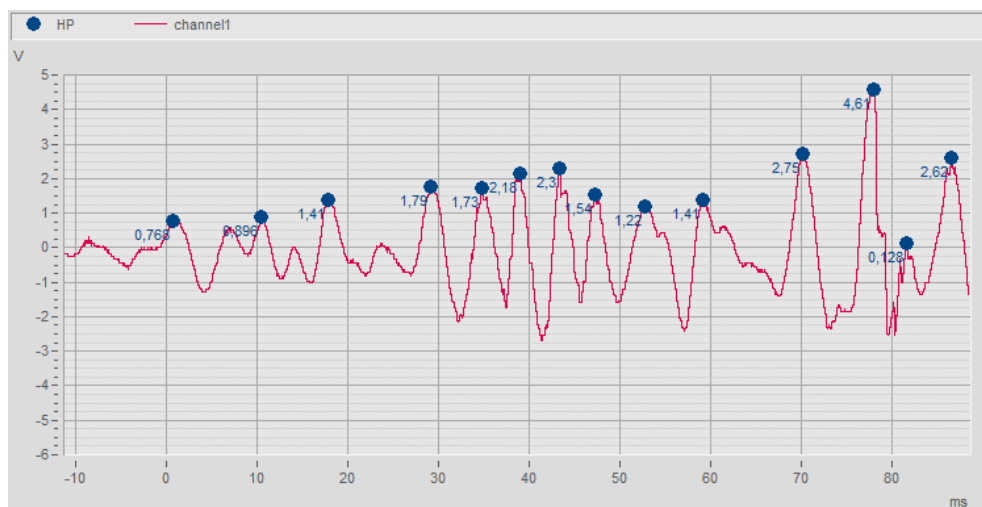
Übung D

Übungsziel:

Von einem Datensatz sollen alle relativen Hochpunkte mit einer gewissen Mindestamplitude zwischen den Extrema bestimmt werden. Im verwendeten Beispiel sollen nur Oszillationen mit einer Mindestamplitude von 1V beachtet werden.

Ergebnis:

Sie erhalten folgendes Ergebnis. Es werden alle Hochpunkte berechnet, die nach einem Tal mit der Mindestamplitude von 1V folgen.



Übungsschritte:

- Navigieren Sie im Datenbrowser zum Ordner „EXPERIM“ in den Beispieldatensätzen und laden Sie den Kanal „CHANNEL1.DAT“ aus dem Messungsordner „0001“. Stellen Sie den geladenen Kanal anschließend in einem Kurvenfenster dar.
- Um alle Oszillationen zu eliminieren, die kleiner als die gewünschte Mindestamplitude sind, filtern Sie den Datensatz mit einem Hysteresefilter (Funktion Hyst()) in einem Hilfsdatensatz.

`help = hyst(channel1, 1)`

`hp = FindExtrema(channel1, "Local.Max")`

- Öffnen Sie die Eigenschaften des Kurvenfensters und wechseln Sie zum Thema „Linien“. Passen Sie die Einstellungen für den Datensatz der Hochpunkte so an, dass diese nur als Punkte mit entsprechendem Label gezeigt werden.

Versionen vor FAMOS 2025:

- Bestimmen Sie die x-Werte aller Hochpunkte mit der xMaxi() Funktion.
- Die zugehörigen y-Werte können mit der Funktion Value() bestimmt werden.

`x_Koord = xMaxi(help, -1e35)`

`y_Koord = Value(channel1, x_Koord)`

- Erstellen Sie aus den x- und y-Werten einen neuen xy-Datensatz, der alle gewünschten Hochpunkte beinhaltet. Fügen Sie diesen Datensatz anschließend dem Kurvenfenster hinzu.

`HP = xyOF(x_Koord, y_Koord)`