

# Übungen zu imc FAMOS II – Digital Kurs

- Block 1 -

Doc. Rev.: 1.2- 27.08.2025



Gezielter Wissenstransfer – höhere Produktivität

# Übung A

## Übungsziel:

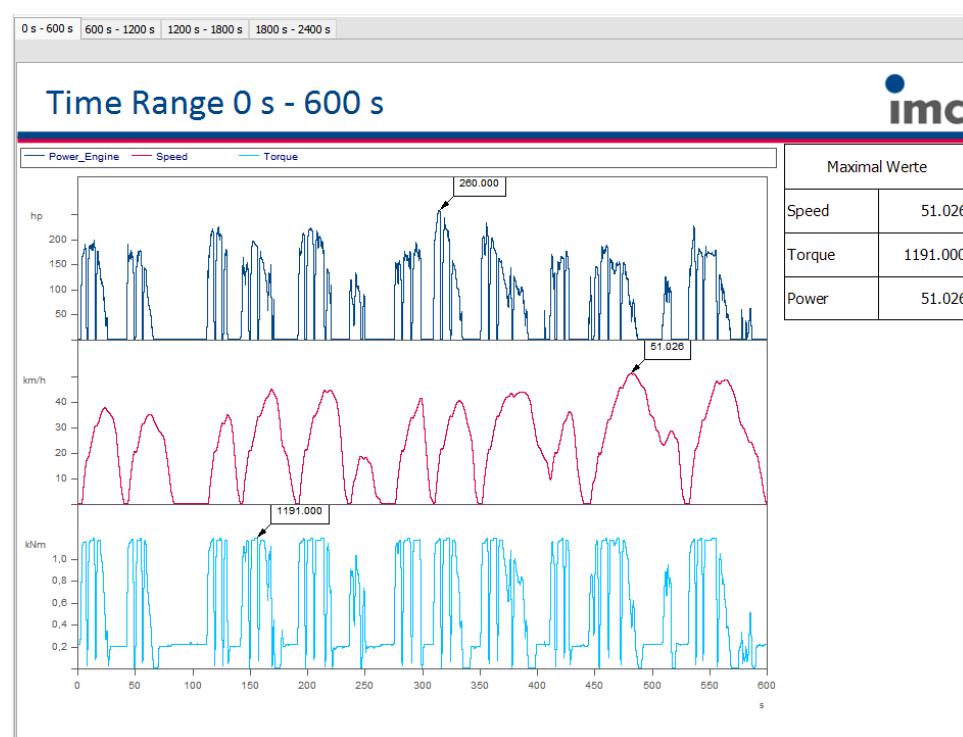
Ziel ist die Entwicklung einer Sequenz, welche einen längeren Datensatz in mehrere Abschnitte gleicher Länge unterteilt und jeweils eine eigene Panelseite für jeden Abschnitt erzeugt. Auf den Panelseiten sollen dann der jeweilige Abschnitt sowie einige ausgewertete Parameter der einzelnen Abschnitte gezeigt werden.

Als Techniken kommen die Funktionsgruppen zur Nutzung des Kurvenfensters und das Panel zum Einsatz. Darüber hinaus werden Befehle zum Umgang mit Zeiten, Ausschnitten und Schleifen vorgestellt bzw. vertieft.

## Aufgabenstellung:

- Erzeugen Sie mit den Messkanälen aus dem Beispieldatensatz **bustrip.dat** mithilfe der Panelvorlage **Ueb1.panel** mehrere Panelseiten, die fortlaufend je 600 Sekunden der Messungen im Kurvenfenster anzeigen (0-600 s, 600-1200 s, ... usw.).
- Die Messdaten sollen in den Kurvenfenstern der Panelseiten übereinander angeordnet werden.
- Beschriften Sie die Panelseiten (Tab-Beschriftung) sowie die Überschrift-Textfelder in den Panelseiten mit diesen Bereichsgrenzen.
- Berechnen Sie die Maximalwerte für jeden Bereich und jeden Kanal und tragen diese in die jeweiligen Tabellen sowie als Marker in die jeweiligen Kurvenfenster ein (Ergebnisse immer mit 3 Nachkommastellen).
- Erstellen Sie einen Report, indem Sie die Panelseiten als PDF exportieren.

## Ergebnis:



Das Ergebnispanel soll wie in der nebenstehenden Grafik aussehen.

Zusätzlich erhalten Sie ein PDF als Report mit analogem Inhalt.

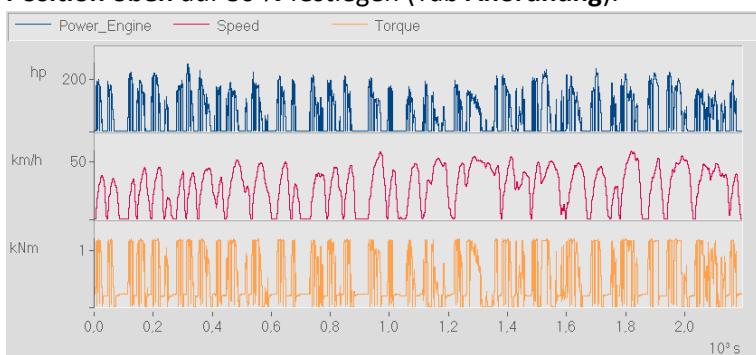
## Übungsschritte:

### Vorbereitung

- Laden Sie zunächst das Panel **Ueb1.panel**, welches in den heruntergeladenen Beispieldateien enthalten ist. Anschließend laden Sie den Datensatz **bustrip.dat**, der sich ebenfalls in den heruntergeladenen Beispieldateien befindet. Nach dem Laden sollten die 3 Kanäle **Speed**, **Torque** und **Power\_Engine** in der Variablenliste vorhanden sein.

### Erstellung der Kurvenfenster-Konfiguration

- Visualisieren Sie die Messdaten in einem freifliegenden Kurvenfenster (y-Achsen übereinander).
- Um später die eingefügten Marker gut sehen zu können, erzeugen Sie noch etwas Platz über den jeweiligen Kurven, indem Sie in der Kurvenfenster-Konfiguration für alle y-Achsen die **Position oben auf 80 %** festlegen (Tab **Anordnung**).



- Speichern Sie diese Kurvenfenster-Konfiguration als **bustrip.csv** im vorgeschlagenen Standardverzeichnis ab und schließen das Kurvenfenster. Das eingebettete Kurvenfenster im Panel soll später mit dieser Konfiguration geladen werden.
- Hinweis: Die Datei **bustrip.csv** können Sie auch in den heruntergeladenen Beispieldateien finden.

### Erstellung der Schleife

- Erstellen Sie eine neue Sequenz durch Klick auf das zugehörige Icon im Eingabefenster und speichern Sie diese ab.
- Mithilfe einer **While**-Schleife sollen nun nach und nach die Panelseiten erzeugt werden. Dazu werden zunächst die zeitlichen Positionen des linken und rechten Randes des ersten Zeitfensters mit zwei Hilfsvariablen definiert. Pro Seite soll jeweils ein 600 Sekunden-Abschnitt der **bustrip**-Messungen anzeigen:

```
Range0 = 0; Linker Rand des 600-s-Zeitfensters
Range1 = 600; Rechter Rand des 600-s-Zeitfensters
```

- Die **While**-Schleife soll solange laufen, bis der linke Rand **Range0** kleiner oder gleich der Zeitlänge eines Kanals (hier bspw. **Speed**) ist. Am Ende jedes Schleifendurchlaufs werden beide Ränder für den nächsten Schleifendurchlauf um 600 Sekunden inkrementiert:

```
; solange Zeitfenster < Kanaldauer
While Range0 <= XDel?(Speed)*(Leng?(Speed)-1)
    ; weiterer Code
    Range0 = Range0 + 600 ; 600 Sekunden weiterrücken für
    Range1 = Range1 + 600 ; nächsten Schleifendurchgang
End
```

### Erstellung des Schleifeninhalts

- Innerhalb der Schleife wird nachfolgend der Funktionsteil der Schleife entwickelt. Achten Sie darauf, dass die Updates von **Range0** und **Range1** immer als letztes am Schleifenende verbleiben.
- Die einzelnen Panelseiten werden erzeugt, indem bei jedem Schleifendurchgang die erste Panelseite kopiert wird. Für die Namengebung der Panelseiten soll das Schema „0 s – 600 s“ (als Beispiel für den ersten Block) verwendet werden.

```
Paneltext = TForm(Range0, "f1.0") + " s - " + TForm(Range1, "f1.0") + " s"
PnInsertPage("", 1, Paneltext, 0)
```

- Lesen Sie den Namen des eingebetteten Kurvenfensters in seinen Widget-Eigenschaften ab und laden Sie die zuvor erstellte Kurvenfenster Konfiguration

```
CwLoadCCV("Curve1", "bustrip.ccv")
```

- Als nächstes sollen die Bereichsgrenzen für die x-Achse gesetzt werden. Dazu selektieren Sie zuerst das Kurvenfenster und dann die enthaltene X-Achse:

```
CwSelectWindow("Curve1")
CwSelectByIndex("x-axis", 1)
```

- Anschließend setzen Sie die Bereichsgrenzen der X-Achse mit Hilfe der Range Variablen. Zuvor muss allerdings die Bereichswahl immer auf „fest“ gesetzt werden, damit diese auch übernommen werden:

```
CwAxisSet("range", 4); immer erst Range auf "feste Grenzen" setzen
CwAxisSet("min", Range0)
CwAxisSet("max", Range1)
```

- Um die Maximalwerte für jeden Abschnitt berechnen zu können, müssen jeweils die anzuzeigenden Abschnitte aus den Daten herausgeschnitten werden. Anschließend lassen sich die Maxima der Ausschnitte bestimmen:

```
CutSpeed = Cut(Speed, Range0, Range1)
CutTorque = Cut(Torque, Range0, Range1)
CutPower = Cut(Power_Engine, Range0, Range1)
MaxSpeed = Max(CutSpeed)
MaxTorque = Max(CutTorque)
MaxPower = Max(CutPower)
```

- Setzen Sie die berechneten Abschnittsmaxima in die Tabelle auf dem Panel. Finden Sie dazu wieder zunächst den Namen über die Widget Eigenschaften heraus:

```
PnTableSetCell("Tab1", 2, 2, TForm(MaxSpeed, "f1.3"))
PnTableSetCell("Tab1", 2, 3, TForm(MaxTorque, "f1.3"))
PnTableSetCell("Tab1", 2, 4, TForm(MaxPower, "f1.3"))
```

- Die Überschrift der Panelseite wird durch ein Label-Widget definiert. Setzen Sie die Überschrift, sodass diese den jeweiligen Time Range beinhaltet:

```
PnSetText("LB_Start", "Time Range " + Paneltext)
```

- Als nächstes sollen jeweils Marker an die Maximalpositionen aller Kurven gesetzt werden. Da in der Kurvenfenster-Konfiguration mehr als 1 Koordinatensystem vorhanden ist, muss zuerst die

Kurvenlinie des betreffenden Kanals per Sequenz selektiert werden. Beginnen Sie mit dem Kanal **Speed**:

*CwSelectByChannel("Line", Speed); Selektiert Linie des Kanals*

- Erstellen Sie anschließend einen Marker und ordnen ihn der selektierten Linie zu:

*CwNewElement("marker")*

*CwMarkerSet("Line.selected", 1); Anordnung Marker an selektierte Linie*

- Abschließend muss der Marker korrekt positioniert werden, Koordinatenangaben sollen dabei als physikalische Einheit (und nicht als % der Achsenlänge) gedeutet werden:

*CwMarkerSet("x.type", 1); Physikalische Einheiten werden genommen*

*CwMarkerSet("y.type", 1); s.o.*

*CwMarkerSet("x", Pos(CutSpeed, MaxSpeed)); Position X (Max vom Abschnitt)*

*CwMarkerSet("y", MaxSpeed); Position Y*

*CwMarkerSet("text", TForm(MaxSpeed, "f1.3")); Markertext*

- Wiederholen Sie alle Schritte zur Marker-Erstellung für die Kanäle **Torque** und **Power\_Engine**. Achten Sie darauf die entsprechenden Kanäle vorher zu selektieren.
- Nachdem die Schleife vollständig abgearbeitet wurde, soll noch die erste Seite, welche als Vorlage diente, entfernt werden. Exportieren Sie anschließend das Panel als PDF:

*PnRemovePage(1); Leere Dialogseite rauschmeißen*

*PnExportPDF("C:\temp\Report.pdf", 0, 0)*

- Speichern Sie die erstellte Sequenz ab.

# Übung B

## Übungsziel:

Ziel der Übung ist es eine Auswertung eines Datensatzes, bei der bestimmte Kennwerte bestimmt werden, auf einen ganzen Satz an Datensätzen parallel anzuwenden. Als Techniken kommen dabei das Schneiden von Datensätzen, das Arbeiten mit Gruppen und Schleifen sowie die Generierung von Datensätzen mit fest vorgegebenen Werten zum Einsatz.

## Übungsbeschreibung:

Bestimmen Sie aus dem Beispieldatensatz **LoadTestSample.dat** für die Variablen **load1..4** die Mittelwerte der einzelnen Plateaus.

Zur Durchführung der Übung gehen Sie folgendermaßen vor:

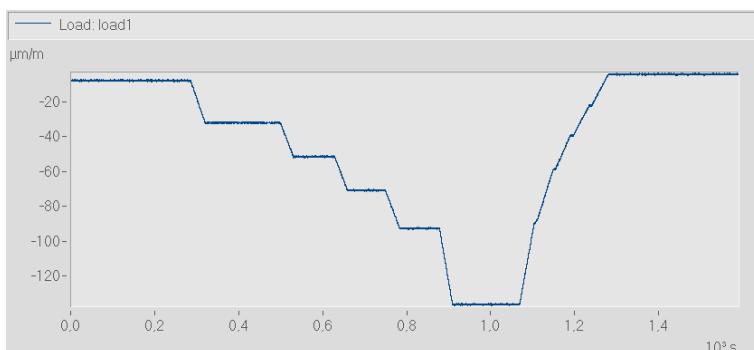
- Bestimmen Sie mithilfe der Messcursoren ungefähr die zeitlichen Mittelpunkte der Plateaus und erstellen mit diesen Zeiten einen Kanal mit dieser Zahlenabfolge.
- Schneiden Sie sich mittels einer Schleife erst einmal probeweise nur für den ersten der 4 Datensätze an allen ermittelten Zeiten jeweils ein 20-Sekunden langes Datenstück heraus.
- Bestimmen Sie die Mittelwerte dieser Stücke jeweils in einem neuen Datensatz, der den jeweiligen verwendeten zeitlichen Mittelpunkt im Namen enthält.
- Um die Berechnung gleichzeitig für alle **load1..4** durchzuführen, gruppieren Sie anschließend diese Variablen und führen die erstellte Sequenz stattdessen mit der Gruppe selbst durch.

## Ergebnis:

Sie erhalten für jedes der 7 Plateaus aus den Datensätzen **load1..4** jeweils eine neue Gruppe, die die Mittelwerte aller 4 Datensätze zu den jeweiligen Plateaus enthalten. Die Namen der Gruppen beinhalten dabei jeweils die mittleren Zeitpunkte der Plateaus.

## Übungsschritte:

- Laden Sie den heruntergeladenen Datensatz **LoadTestSample.dat** über den FAMOS-Browser in die Variablenliste. Anschließend sollten die Variablen **Force1..2**, **load1..4**, **displacement1..3** in der Variablenliste erscheinen.
- Visualisieren Sie den Datensatz **load1**. In diesem sind insgesamt 7 Plateaus mit annähernd gleicher Last zu sehen, die über eine jeweils unterschiedliche Dauer anliegen.



- Bestimmen Sie mit Hilfe der Messcursoren die ungefähren zeitlichen Mittelpunkte der Plateaus und notieren Sie diese. Erstellen Sie einen **Time\_of\_Interest** Kanal, welcher die notierten Zeiten enthält und vergeben diesem die y-Einheit „s“:

```
Time_of_Interest = [140, 410, 580, 700, 830, 990, 1440]
SetUnit(Time_of_Interest, "s", 1)
```

- Erstellen wir zunächst die Schleifenstruktur über die Anzahl der Plateaus:

```
FOR i = 1 TO Leng?(Time_of_Interest)
; Code der Schleife
END
```

- Zu Beginn jedes Schleifendurchlaufs soll jeweils für das i-te Plateau die Start- und Endzeit des herauszuschneidenden Datenstücks bestimmt werden. Da die Länge der herausgeschnittenen Daten jeweils 20-Sekunden betragen soll, lassen sich die gesuchten Zeitpunkte wie folgt definieren:

```
x1 = Time_of_Interest[i] - 10 ; Startzeit
x2 = Time_of_Interest[i] + 10 ; Endzeit
```

- Schneiden Sie das entsprechende Datenstück heraus:

```
CutLoad = Cut_Load1, x1, x2)
```

- Erstellen Sie eine neue Textvariable, die sich aus dem Namen **MeanLoad\_** sowie den jeweiligen Einträgen aus dem **Time\_of\_Interest** Datensatz zusammensetzt.

```
TxMeanLoad = "MeanLoad_" + TForm(Time_of_Interest[i], "")
```

- Verwenden Sie die erstellte Textvariable, um eine neue Variable mit dem Namen des Textes zu erzeugen, die den Mittelwert aus dem ausgeschnittenen Datenstück enthält.

```
<TxMeanLoad> = Mean(CutLoad)
```

- Speichern Sie die Sequenz und führen Sie diese als Testlauf vollständig aus. Sie sollten anschließend 7 Variablen mit den entsprechenden Zeitpunkten im Namen in der Variablenliste haben, die die jeweiligen Mittelwerte der Plateaus angeben.

### Anwendung auf eine Gruppe

- Die Berechnungen der Sequenz sollen im Folgenden für alle **load**-Variablen gleichzeitig durchgeführt werden. Dazu markieren Sie in der Variablenliste **load1** bis **load4** und wählen im Kontextmenü (Rechtsklick) **Zusammenfassen**, um die selektierten Variablen in einer Gruppe zusammenzufassen. Als Gruppennamen geben Sie **load** (ohne Ziffer) ein.
- Um die erstellte Sequenz auf die Gruppe anzuwenden, muss in dieser die Variable **load1** durch den Gruppennamen **load** ersetzt werden.
- Führen Sie die Sequenz aus. Sie erhalten als Ergebnis jeweils eine neue Gruppe für jeden Zeitpunkt des **Time\_of\_Interest** Datensatzes mit den Mittelwerten der einzelnen ausgeschnittenen Plateaus aus **load1..4**.
- Speichern Sie die Sequenz ab.

Die Gesamtsequenz sieht nun folgendermaßen aus:

```
Time_Of_Interest = [140, 410, 580, 700, 830, 990, 1440]
SetUnit(Time_Of_Interest, "s", 1)

FOR i = 1 TO Leng?(Time_Of_Interest)
    x1 = Time_Of_Interest[i] - 10 's'
    x2 = Time_Of_Interest[i] + 10 's'
    CutLoad = Cut(Load, x1, x2)
    TxMeanLoad = "MeanLoad_" + TForm(Time_Of_Interest[i], "")
    <TxMeanLoad> = Mean(CutLoad)
END
```

# Übung C

## Übungsziel:

Diese Übung ist eine Erweiterung zu Übung B. Auch hier ist das Ziel der Übung eine Auswertung eines Datensatzes, bei der bestimmte Kennwerte bestimmt werden, auf einen ganzen Satz an Datensätzen parallel anzuwenden. Die Kennwerte werden in diesem Fall nicht einfach ausgegeben, sondern direkt für weiterführende Auswertungen verwendet. Als Techniken kommen dabei das Verbinden von Datensätzen, das Arbeiten mit XY-Datensätzen sowie das Sortieren von Werten zum Einsatz.

## Übungsbeschreibung:

Erstellen Sie aus **LoadTestSample.dat** die Kennlinien **displacement1** über **Force1** sowie **displacement2** über **Force2** aus den Mittelwerten ihrer 7 Plateaus und lassen sich beide Kennlinien im selben Koordinatensystem und auf der gleichen Achse anzeigen. Arbeiten Sie in den Berechnungen von Beginn an mit Gruppen statt mit Einzelkanälen.

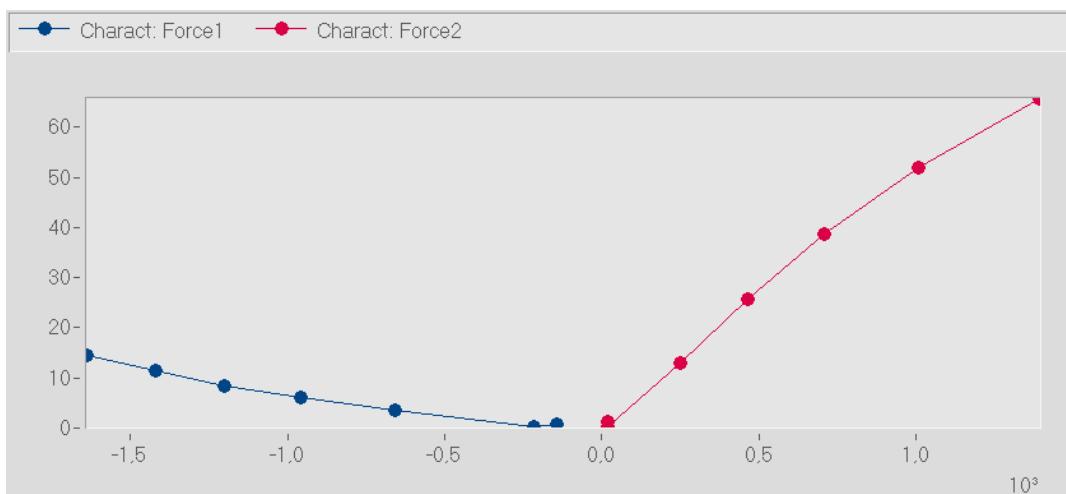
Hinweis: Die Positionen aller Plateaus in **LoadSampleData.dat** stimmen zeitlich überein.

Zur Durchführung der Übung gehen Sie folgendermaßen vor:

- Übernehmen Sie Ihren Quelltext aus Übung B.
- Gruppieren Sie in der Variablenliste **Force1..2** zu **Force** und **displacement1..2** zu **Displacement**.
- Bestimmen Sie analog zu **CutLoad** aus Übung B die Stücke **CutDisp** und **CutForce**.
- Erstellen Sie zwei Verläufe mit allen Mittelwerten aus **Displacement** und aus **Force**.
- Nach der Schleife verbinden Sie die 2 Verläufe zu einem neuen XY-Datensatz **Displacement** über **Force** und lassen sich die Kennlinien im Kurvenfenster anzeigen.
- Sortieren Sie die XY-Datensätze nach den X-Werten der Kennlinien.

## Ergebnis:

Das Ergebnis dieser Übung liefert jeweils eine berechnete Kennlinie für die Datensätze **Displacement** über **Force**, die im nachfolgenden Kurvenfenster gezeigt werden:



## Übungsschritte:

- Die Berechnungen aus Übung B sollen ebenfalls auf die Kanäle **Force1..2** und **displacement1..2** angewendet werden. Anstelle der Mittelwerte der einzelnen Plateaus in separaten Datensätzen einfach auszugeben, sollen in diesem Fall jedoch die einzelnen Plateau-Mittelwerte verwendet werden, um eine Displacement-Force-Kennlinie zu erstellen.
- Kopieren Sie den Inhalt der Sequenz aus Übung B in eine neue Sequenz.
- Gruppieren Sie die Kanäle **Force1..2** in einer Gruppe **Force** sowie **displacement1..2** in einer Gruppe **Displacement**.
- Ergänzen Sie den Schleifeninhalt der Sequenz um entsprechende Kommandos, um Teilstücke für **Displacement** und **Force** analog zu den **load** Teilstücken zu erstellen:

*CutDisp = **Cut**(Displacement, x1, x2)  
 CutForce = **Cut**(Force, x1, x2)*

- Berechnen Sie wieder die Mittelwerte für jedes Teilstück. Anstatt diese jedoch in neue Datensätze zu schreiben, fügen Sie die berechneten Werte einem Datensatz **DispTrd** und **ForceTrd** hinzu, sodass diese am Ende der Schleife alle Mittelwerte der Plateaus enthalten:

*DispTrd = **Join**(DispTrd, **Mean**(CutDisp))  
 ForceTrd = **Join**(ForceTrd, **Mean**(CutForce))*

- Da zu Schleifenbeginn **DispTrd** und **ForceTrd** noch nicht existieren, initialisieren Sie diese vor der Schleife als leere Datensätze:

*DispTrd = **empty**  
 ForceTrd = **empty***

- Nach der Schleife sollen die die Kennlinien erzeugt werden. Fassen Sie dazu die Gruppen **DispTrd** und **ForceTrd** zu einem XY-Datensatz namens **Charact** zusammen:

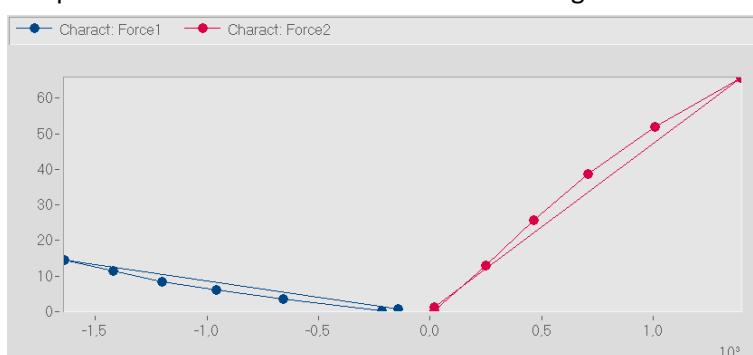
*Charact = **XYof**(ForceTrd, DispTrd)*

**DispTrd** ist hier als Y-Koordinate und **ForceTrd** als X-Koordinate festgelegt. Auch hier erkennt FAMOS automatisch, dass es sich bei **ForceTrd** und **DispTrd** um Gruppen handelt und bildet automatisch eine neue Gruppe **Charact**.

### Achtung!

*FAMOS erzeugt hier keine Warnung, obwohl nicht die gleiche Anzahl an Datensätze in den Gruppen vorhanden sind. Somit fällt der dritte Kanal **displacement3** hier weg, ohne dass der Nutzer davon etwas mitbekommt.*

- Vor der **Charact**-Gruppe in der Variablenliste ist das Hysterese-Symbol  zu sehen, da die X-Komponenten nicht monoton steigend sind. Sie können dies nachvollziehen, indem Sie beispielsweise den Plateau-Verlauf der Messung **Force1** anschauen. Entsprechend sieht die

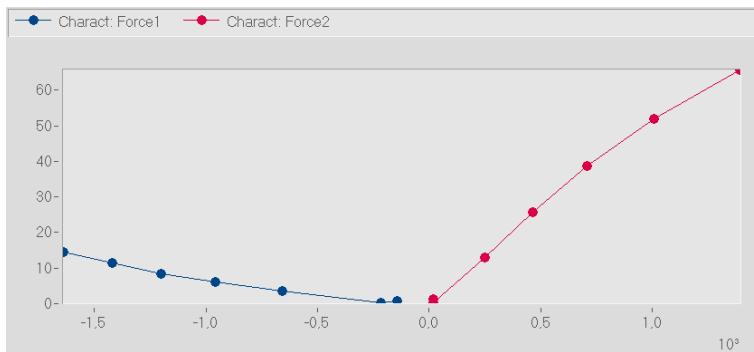


Kennlinie aus, wenn Sie **Charact** visualisieren (1 Koordinatensystem, 1 Achse, Linien mit Punkten):

- Um die Kennlinie wie gewohnt darzustellen, sortieren Sie die Werte der Kennliniengruppe aufsteigend nach den X-Werten:

*Charact = Sort(Charact, 7)*

Bei Ausführung dieses Kommandos ändern sich nun die Variablen **Symbol** zu **Symbol** und Sie erhalten für die Kennlinie **Force** über **Displacement** folgenden Kennlinienverlauf:



- Speichern Sie die Sequenz ab.

Der vollständige Code sieht folgendermaßen aus:

```

Time_Of_Interest = [140, 410, 580, 700, 830, 990, 1440]
SetUnit(Time_Of_Interest, "s", 1)
DispTrd = empty
ForceTrd = empty
FOR i = 1 TO Leng?(Time_Of_Interest)
    x1 = Time_Of_Interest[i] - 10 's'
    x2 = Time_Of_Interest[i] + 10 's'
    CutLoad = Cut(Load, x1, x2)
    TxMeanLoad = "MeanLoad_" + TForm(Time_Of_Interest[i], "")
    <TxMeanLoad> = Mean(CutLoad)
    CutDisp = Cut(Displacement, x1, x2)
    CutForce = Cut(Force, x1, x2)
    DispTrd = Join(DispTrd, Mean(CutDisp))
    ForceTrd = Join(ForceTrd, Mean(CutForce))
END

Charact = XYof(ForceTrd, DispTrd)
Charact = Sort(Charact, 7)

```