Measurement uncertainty: Improvements in the algorithm

Applications in imc FAMOS

Improvements in the algorithm

This document describes how you can improve your evaluation algorithm. A temperature measurement over time exists (see Fig. 1). The measuring duration was approx. 50 sec and a temperature rise from approx. 24 °C to approx. 36 °C is recognisable. You now wish to determine the rise time and the measurement uncertainty occurring in the analysis.

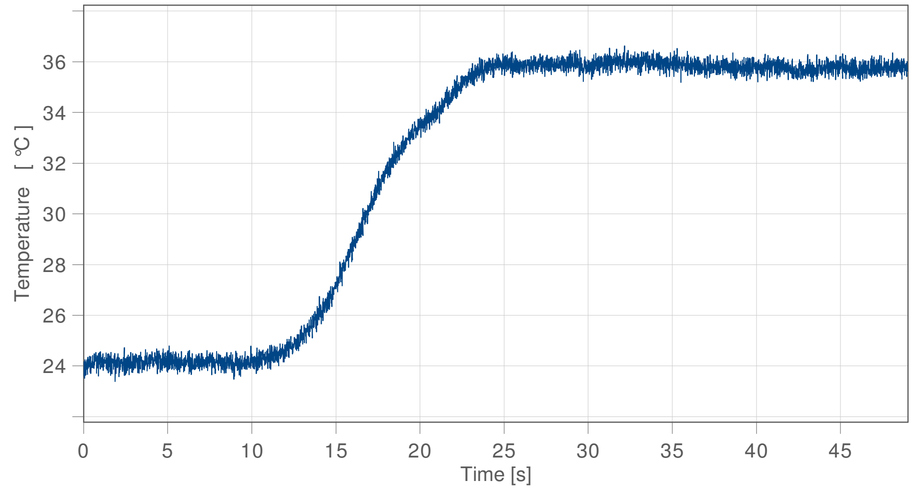

Figure 1

The rise time can be calculated in imc FAMOS with the following algorithm:

```
_Temperature = Temperature
L1 = mean(cut(_Temperature,5,10))
L2 = mean(cut(_Temperature,30,35))
RiseTime = pos(_Temperature, L1+(L2-L1)*0.9) - pos(_Temperature, L1+(L2-L1)*0.1)
```

The result for the rise time is:

RiseTime = 8.6s

Explanation of the code:

Calculation of L1 = 24.1 °C: The mean value of the temperature values between the 5th and 10th second of the measurement is determined.

Calculation of L2 = 36.0°C: The mean value of the temperature values between the 30th and 35th second is determined.

Calculation of the rise time: The rise time is calculated. The temperature start and end value is thereby multiplied by a factor (0.1 or 0.9)

But how precise is the result? The measurement uncertainty is estimated from the visible noise band. Fig. 2 shows the noise band of the range from seconds 6 to 8 (the width of the noise band is constant over the duration of the measurement). The peaks lie between 23.7 °C and 24.5 °C of the temperature measurement. From the available data, the assumption of 0.25 °C for the measurement uncertainty appears quite reasonable.

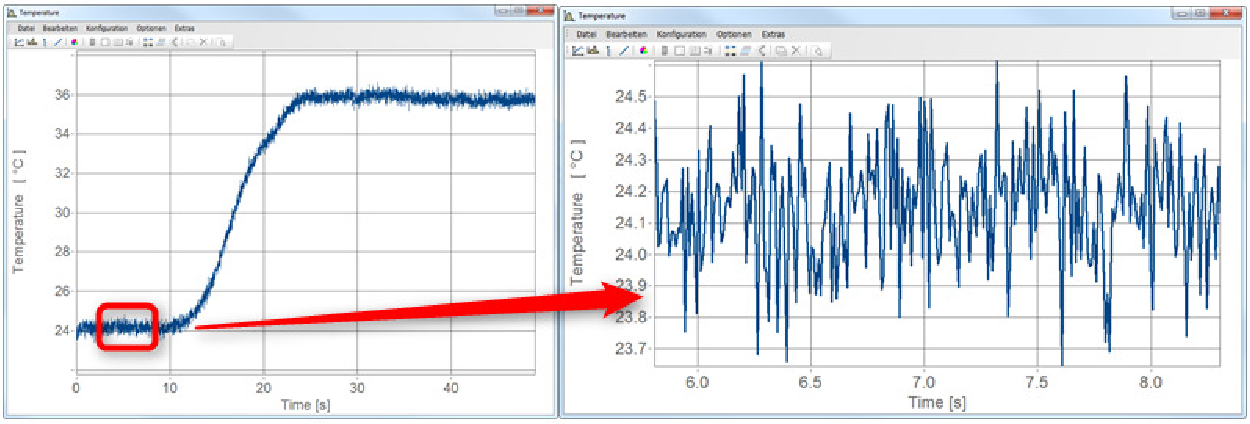

Figure 2

The measurement uncertainty of the rise time can be determined with a slightly expanded algorithm:

```
UncertaintySet( Temperature, "Uncertainty", 0.25)
UNCERTAINTY_LOOP 1000
_Temperature = UncertaintyModify (Temperature)
L1 = mean(cut(_Temperature,5,10))
L2 = mean(cut(_Temperature,30,35))
RiseTime = pos(_Temperature, L1+(L2-L1)*0.9) - pos(_Temperature, L1+(L2-L1)*0.1)
UncertaintyCalc (RiseTime )
End
uc = UncertaintyGet( RiseTime, "Uncertainty")
```

The result for the measurement uncertainty of the rise time is

uc = 2.6s.

Explanation of the code:

UncertaintySet(): the estimated measurement uncertainty (0.25) is assigned to the measurement data.

Uncertainty_Loop: 1000 Monte Carlo experiments are carried out in which the input data are slightly varied each time (addition of noise) using UncertaintyModify(). Like the first sequence, a mean value is calculated for the lower limit value and for the upper limit value. After that the rise time and the uncertainty of the rise time (uc) are calculated.

With a value of 8.6 s, a measurement uncertainty of 2.6 s means a relative measurement uncertainty of 30%. It should be noted that the correct interpretation is: "if we were to repeat the measurement several times, the calculated value of the rise time would be scattered by 2.6 s!". The result is not satisfactory in this form.

What is the cause of the large measurement uncertainty? Is the measurement uncertainty of the input data now too large? Or is the algorithm poor? Both are true! However, the measurement uncertainty of the input signal can only be changed with difficultly: under certain circumstances the measurement is no longer repeatable. Even if it was, that would be very expensive (better experimental setup, high-quality amplifier, etc.). Conversely, you can improve the algorithm with much less effort. This is accomplished by incorporating smoothing into the algorithm:

```
UncertaintySet( Temperature, "Uncertainty", 0.25)
UNCERTAINTY_LOOP 100 1
    _Temperature = UncertaintyModify (Temperature)
    _Temperature = smo(_Temperature,1)
    L1 = mean(cut(_Temperature,5,10))
    L2 = mean(cut(_Temperature,30,35))
    RiseTime = pos(_Temperature, L1+(L2-L1)*0.9) - pos(_Temperature, L1+(L2-L1)*0.1)
    UncertaintyCalc (RiseTime )
End
uc = UncertaintyGet( RiseTime, "Uncertainty")
```

The measurement uncertainty is now 0.07 s which, with a value of 8.6 s, leads to a relative measurement uncertainty of 0.8%. That is sufficient for the task.

The example shows how easy it is with imc FAMOS to estimate the measurement uncertainty of the results. With the determination of the measurement uncertainty of the analysis results, changes to the algorithm can now be quantitatively assessed, allowing a robust design and optimisation of the algorithm.

Additional information:

imc Test & Measurement GmbH
Voltastr. 5
13355 Berlin, Germany

Telephone:	+49 (0)30-46 7090-0
Fax:	+49 (0)30-46 31 576
E-mail:	hotline@imc-tm.de
Internet:	http://www.imc-tm.com

imc Test & Measurement GmbH is a manufacturer and solution provider of productive test and measurement systems. imc implements metrological solutions for research, development, service and production. imc has particular expertise in the design and production of turnkey electric motor test benches. Precisely outfitted sensor and telemetry systems complement our customer applications.

Our customers from the fields of automotive engineering, mechanical engineering, railway, aerospace and energy use imc measurement devices, software solutions and test stands to validate prototypes, optimize products, monitor processes and gain insights from measurement data. As a solution provider, imc offers their customers an attractive and comprehensive range of services. These include project consulting, contracted measurements, data evaluation, specialist deployment, customer-specific software development and system integration. imc consistently pursues its claim of providing services for "productive testing".

If you would like to find out more specific information about imc products or services in your particular location, or if you are interested in becoming an imc distributor yourself, please go to our website where you will find both a world-wide distributor list and more details about becoming an imc distributor yourself:

http://www.imc-tm.com/our-partners/