



iMakeCalibrationFormula 1.00

Infusion Systems Ltd.
2033 Vendome avenue
Montreal, QC, H4A 3M4
Canada

Tel: (514) 484-5850

Fax: (514) 484-5852

Email: info@infusionsystems.com

<http://www.infusionsystems.com>



2014-12-07 © Infusion Systems, Ltd.

+

Customize your own perfect interface
Performer-activated lighting and sound cues
Interactive trade show displays
music synchronization for dance
greater accessibility via alternate controllers

Interac

[type text]

INTRODUCTION.....	2
PRE-REQUISITES.....	2
ON A MAC.....	2
ON WINDOWS.....	2
INSTRUCTIONS.....	3
CALIBRATION DATA SECTION.....	4
DATA FIT SECTION.....	5

INTRODUCTION

This document describes the elements and the use of iMakeCalibrationFormula, a Matlab standalone design to initially generate calibration formulas. This program reads in calibration data: sensor value vs. physical value being measured (a.k.a. measurand) using the sensor. It displays the data points, and allows processing the input data in order to identify separate zones of interest. Once the zones of interest in the calibration data are identified, it is possible to fit each zone with a polynomial, a sum of gaussians or based on the multiplicative inverse function.¹ At the end of the process, you can save the calibration formulas for each zone and they can be used to calibrate sensor data. This is useful for sensors such as the ReachClose.² In the following, the pre-requisites to run iMakeCalibrationFormula are detailed. Then the basic elements of the GUI are presented and the parameters are explained.

PRE-REQUISITES

You need to install version R2014a of Matlab Compiler Runtime (MCR), available at: <http://www.mathworks.com/products/compiler/mcr/>

ON A MAC

Once the MCR is downloaded, you should unzip the MCRInstaller file. Then, in the MCRInstaller folder, double-click on InstallForMacOSX and follow the instructions.

ON WINDOWS

Make sure to download the 64-bit version of the MCR installer.

Once the MCR is downloaded, simply double-click on the downloaded zip file, then follow the instructions.

¹ http://en.wikipedia.org/wiki/Multiplicative_inverse

² Available at http://infusionsystems.com/catalog/product_info.php/products_id/136

INSTRUCTIONS

Figure 1 depicts the user interface for iMakeCalibrationFormula. It is basically separated into two parts: the **Calibration data** section and the **Data fit** section. Each section is presented in details in what follows.

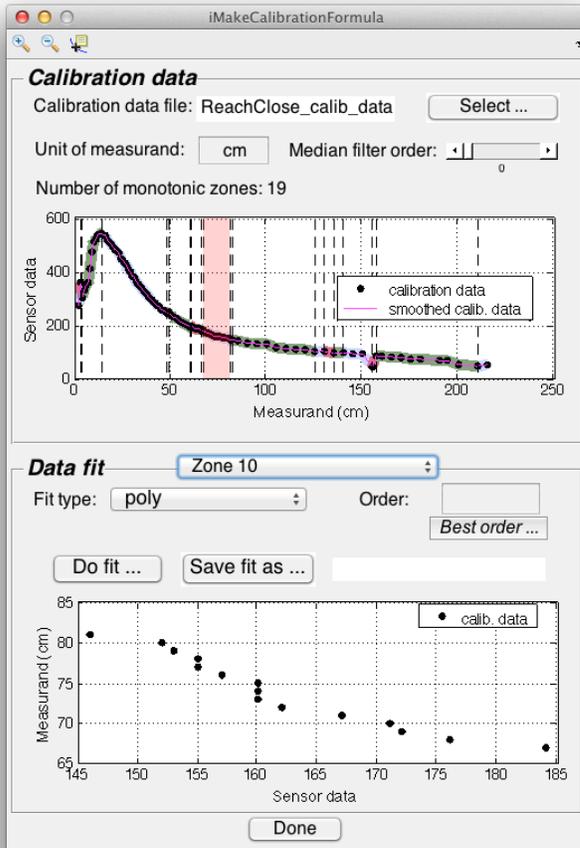


Figure 1: iMakeCalibrationFormula in action on data gathered using the ReachClose sensor.

CALIBRATION DATA SECTION

You can load the calibration data file you want to generate calibration formulas from by clicking on the **Select ...** button (see Figure 1). The name of the calibration file being loaded will then appear (in Figure 1, the file was called ReachClose_calib_data.txt). You can see the full path of the calibration data file by hovering over the name of the file.

You can gather calibration data using an I-CubeX provided patch such as ReachClose 2.00.³ You may get the data using any other method, as long as you save it into a simple .txt file and that it is organized as follows: rows of two columns, separated by a space, where the left column represents the physical value being measured and the right column is the ADC value of the sensor; each row is separated by a carriage return.

When loading a calibration data file, a plot of the sensor data values vs. the measurand values is generated. Each calibration data point is represented by a black dot. You can specify the unit of the measurand in the **Unit of measurand** text box. This information will then be saved in the calibration formula file.

On the plot of Figure 1 one can also see different sections of the calibration data that have been highlighted in various colours and are delimited by vertical dashed lines. The calibration data between two such dashed lines represent a zone of interest for the calibration.

More precisely, such a zone is referred to as a **monotonic zone**⁴ in the rest of this document. These zones are identified automatically based on a smoothed version of the calibration data (the pink line in Figure 1). In simple terms, a monotonic zone should be a range of measurand values for which an increase in the measurand value leads to either an increase of sensor data values or a decrease of sensor data values; not both an increase and a decrease. This is a way to handle calibration response such as the one displayed in Figure 1, where the same sensor data value corresponds to two different distances.

It is clear from Figure 1 that there should only be 2 main monotonic

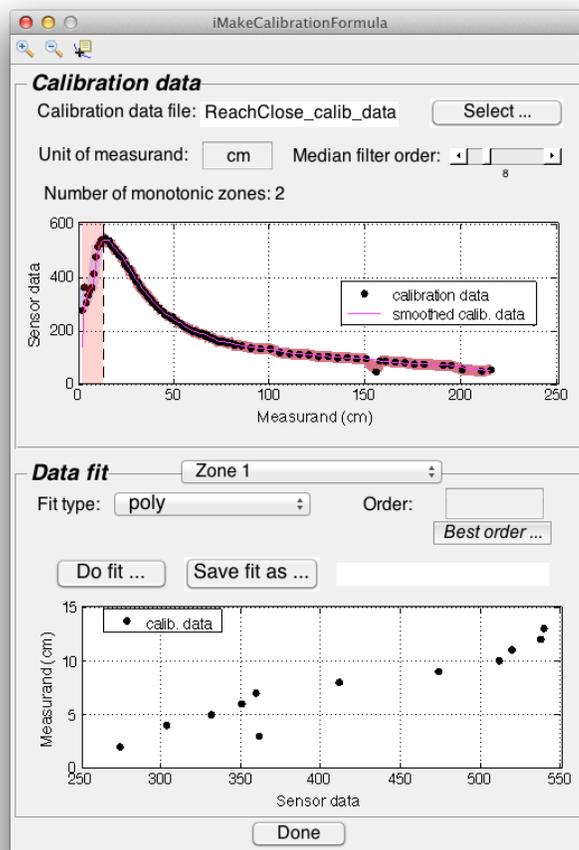


Figure 2: effect of a change in median filter order on the number of monotonic zones detected

³ in the Software Support line of the Technical specification table

http://infusionssystems.com/catalog/product_info.php/products_id/136

⁴ http://en.wikipedia.org/wiki/Monotonic_function for a definition of a monotonic function

zones in the data: one for distances between 0 and about 15cm (where an increase in distance is linked to an increase in sensor data value) and one for distances above 15cm (where an increase in distance leads to diminishing sensor values).

We see that 19 different monotonic zones have been found. This can be adjusted using the **Median filter order** slider.⁵ By default the median filtering order is set to 0 (as in Figure 1) and this does not always lead to good results: instead of 2 monotonic zones, 19 were found. Setting the median filter order to 8, for that example, leads to the appropriate detection of the monotonic zones as can be seen in **Error! Reference source not found.**

DATA FIT SECTION

This section is where the user can choose a model to fit the calibration data for a given monotonic zone.

A typical workflow would look as follows:

1. Select a monotonic zone using the drop-down menu (see menu displaying “Zone 1” in Figure 2).

Then, the measurand vs. sensor data value plot is updated with values corresponding to that zone. Note that, compared to the plot in the Calibration section, the axis have been inverted. This was done intentionally as this is how the fitted model is computed: indeed, a measurand value is computed based on the available sensor data in that zone.

2. Use the **Fit type** drop down menu to select which type of fit you want to use on the data. You can choose between:
 - a. **poly**:⁶ a polynomial for which you can specify the order in the **Order** text box (from 1 to 9). By pressing the **Best Order ...** button the order that minimizes the root mean square error of the fit is selected (the resulting fit is not always what you want but this can be a good start).
 - b. **gauss**:⁷ a sum of gaussian functions, you determine the number of gaussians used using the **Order** text box. You can also use the **Best Order ...** button for this type of fit.
 - c. **m=1/(a*s+b)**: a model where **m**, the measurand value, is obtained from **s**, the sensor value using the following formula: $m = \frac{1}{as+b}$. The parameters a and b are then estimated.
3. Once you have selected a model and set its parameters, you can perform the fit by pressing the **Do fit ...** button (see Figure 2). The fit is then plotted as a red curve in the plot. By visual inspection, you can then determine how good the fit is to your data. An example obtained for Zone 1 of the ReachClose data modeled by a polynomial of order 3 can be found in Figure 3.

⁵ <http://www.mathworks.com/help/signal/ref/medfilt1.html>

⁶ <http://www.mathworks.com/help/curvefit/list-of-library-models-for-curve-and-surface-fitting.html#btbcvn1> under “Polynomial Names and Equations”

⁷ <http://www.mathworks.com/help/curvefit/list-of-library-models-for-curve-and-surface-fitting.html#btbcvn1> under “Gaussian Model Names and Equations”

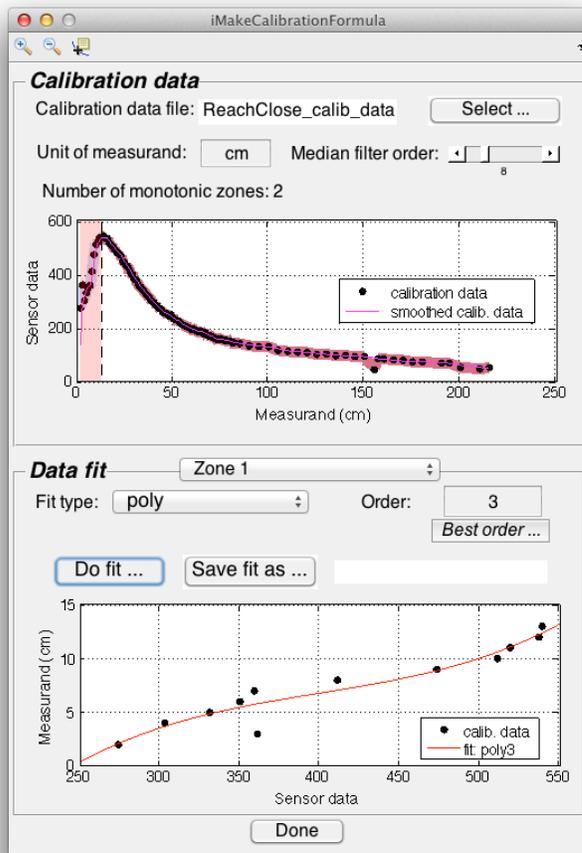


Figure 3: Example of a fit using a polynomial of order 3

4. If you are satisfied with the fit you can save it by pressing on the **Save fit as ...** button. The resulting file is a text file of the form:

```
"measurand_unit","formula","sensor_in_min","sensor_in_max","cf_","fit_type_str",
"cm","0.000000824605740*$i1*$i1*$i1+(-0.000988804418840)*
$i1*$i1+0.419427707000689*$i1+(-55.539934286367576)","274","539","poly3",
```

This file can then be used in Max using the **iCalibrate** external.