Release Notes Version 2.2 D R A F T

As a prerequisite please study the documents for the original autoISF (Github) and the draft for versions 2.0/2.1 (private communication).

Accelerated ISF

The term „accelerated ISF" refers to the main new feature, the adaptation of ISF based on the acceleration of the glucose level. The regular AAPS does not use the glucose acceleration apart may be from some rough estimate by comparing delta, short average delta and long average delta. But even if so such a number is not up to date as it uses data from up to 40 minutes from the past. The method used here first calculates a best fit of the recent glucose to a parabola. Once that formula is known it is an easy second step to calculate best fit delta and accelerations as first and second derivatives of glucose:

$$
\begin{aligned}
gluc(t) &= a_2*t^2 + a_1*t + a_0 \\
delta(t) &= 2*a_2*t + a_1 \\
acce(t) &= 2*a_2
\end{aligned}
$$

Here "t" is the time difference relative to the last measured glucose value and expressed in units of 5 minutes. This means that for the current situation t=0 and a_0 is the fitted glucose value and a_1 is the fitted delta. Further, *parabolic_fit_last_delta* is a_1-a_2, *parabolic_fit_next_delta* is a_1+a_2 .

This method is a well known trick in science to fit a theoretical curve or surface to measured data and then the downstream analyses become easy and fast by using these analytical formulae. Also, this fit has a smoothing effect.

This whole context reminds me of Newtons 2nd law, this time expressed inversely: if a body experiences an acceleration then there is a force acting on it. By analogy, if glucose accelerates there is some „sweet force" acting in the system driving glucose higher and we should counter it by more insulin, e.g. strengthening ISF. Likewise, if glucose decelerates there is some „antisweet force" driving glucose down and we should counter it by reducing insulin, e.g. weakening ISF. Not knowing anything about the metabolism behind accelerated glucose the easiest starting point is a linear change of sensitivity with acceleration:

acce_ISF = 1 + acce * weight

There are two independent implementations of that weight factor and both can be defined in Preferences → Absorption Settings → Advanced:

1. *bgAccel_ISF_weight* is used for positive acceleration. With 0 there is no change and the higher the number the more aggressive it will react. Start small, e.g. 0.02 and increase it gradually. Using the emulator I had figured that 0.15 might be a suitable starting value for me. I worked my way up to 0.17 but it is still early days … and YDMV.

2. *bgBrake_ISF_weight* is used for negative acceleration, i.e. deceleration. With 0 there is no change and the higher the number the stronger the „braking" effect will be. The main purpose is to slow down IOB build up thus reducing the risk of a hypo later on. On the other hand it will also slow down the decline in glucose levels and you need to find your own balance. Start small, e.g. 0.02 and increase it gradually. I worked my way down from 0.15 found in the emulator down to 0.13 but it is still early days … and YDMV.

I guess these two weights will be of similar size in the end. Because *acce_ISF* can be below 1.0 we now need a lower limit contained in the newly introduced *autoisf_min* which can be found in the Preferences right next to *autoisf_max.* A value of 1.0 would obviously shut off the effect of „braking". This lower limit will now also be active when glucose is low and *bg_ISF* would drive ISF sky high in version 2.0. In that respect version 2.2 behaves different from versions 2.0 and 2.1. One of the first things to do after starting version 2.2 is to define this minimum value because the default is 1.0 !

Combining the adaptations

So far in autoisf all the impacts from resistance, bg level, delta, postprandial meal absorption and now acceleration were independent of each other and the strongest one would be used. With the introduction of acceleration this principle was given up. The reason is that on many occasions the acceleration impact points in the opposite direction of where the other reasons want to go. The two scenarios are:

1. Acceleration is positive, but glucose is below target and bg_ISF wants to weaken ISF – this can happen if carbs were taken, bg rises but is still below target. Here half of the acce_ISF is used and multiplied with bg_ISF and they meat somewhere in the middle. In the future this fixed 50% contribution may be too simple and may need to be refined.

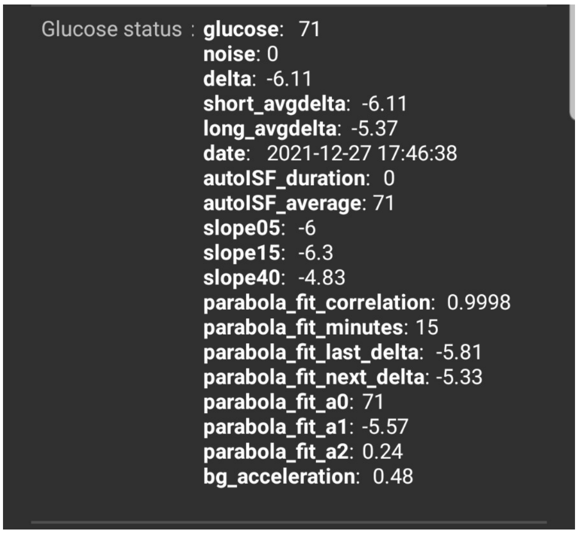
2. Acceleration is negative, glucose is above target and some autoISF functions want to strengthen ISF – in this case adding to IOB should be reduced by what the deceleration says. The resulting effect comes from multiplying acce_ISF with the strengthening ISF factor and we end up somewhere between the two. This can be below 1.0 (potentially save insulin) or above 1.0 (potentially more insulin). It is only potentially because in the end only the loop algorithm decides on the insulin required and just uses the adapted ISF.

How do you see what is going on and why?

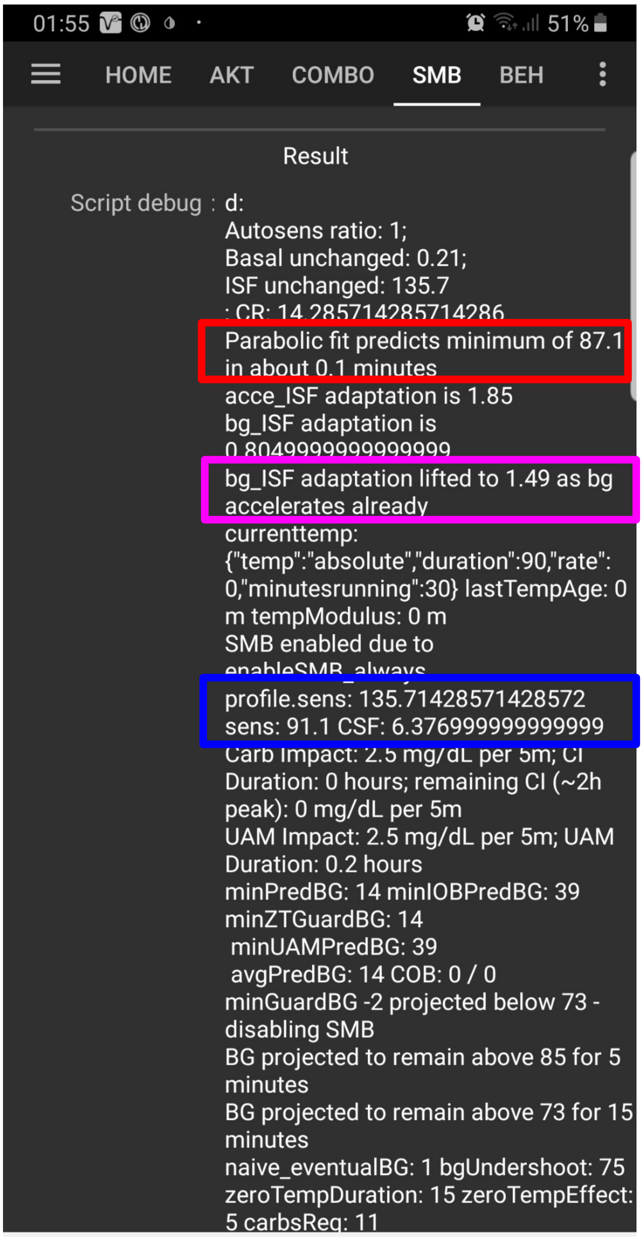
The SMB-tab shows much more information than before. In the end some of that may not really be required or significant and can be removed in the future.

1. Glucose status shows detailed results from the fit. It now also lists the coefficients of the fit. The time units are per 5min or (per 5min)^2, respectively. These are used further down in Results to calculate when and where the parabola had or will have its maximum or minimum.

   After the evaluation phase some of these may be removed.



2. The Results section goes through the decision sequence by listing all the potential contributions to the adapted ISF.
    - Echo of where and when the parabola has its minimum or maximum. The example here is the most useful of the possible combinations because it forecasts when and where the minimum might occur. Before, I did that maths in the top of my head. It can theoretically be consulted when extra carbs are required to judge whether they are really required. However, this has to be watched carefully and please let me know whether it works and is reliable. If not, it should not be listed at all.
    - Echo of the combined effect of acceleration and opposing other reason.
    - The resulting ISF is shown as usual.



Things to investigate

1. Why does it take so long to reach target again once glucose is below target? It looks like the acceleration impact is too strong in these cases.
2. What happens if meals are announced in the typical AAPS way? Will there be too much insulin?

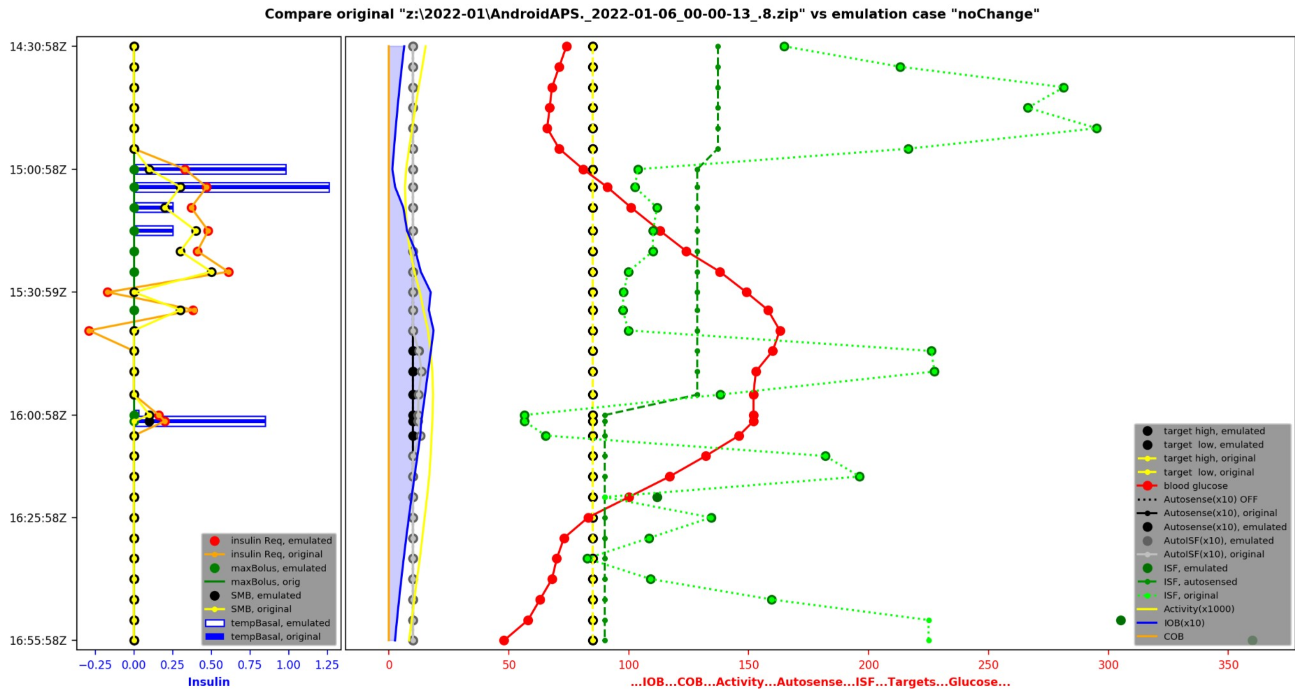
   I tried this on two occasions and there were no problems.
3. How should the combination with autosens be handled? I have not used it for the last 18 months so it slipped my attention.
4. How does it handle gastroparesis?
5. Do we still need the contributions from delta?
6. Read more on Tim Streets method and implementation of his „no bolus“.

Results

… show some statistical results ...

Example 1: A low in the afternoon of Jan.6th

At 14:13Z I had 2 slices of bread containing about 30-40g carbs – with UAM you stop weighing the food. What was the reason for getting low? To investigate this further I started with the emulation of the status quo, firstly to check that the emulator can handle the new capabilities of autoISF2.2 and second to gain more insight. To start here is the graphic output:



The emulator reproduces the original run, that is good news and the vdf-file is fit for a “no change scenario”. From15:00:58T through to 15:35:59Z the ISF was stronger than profile. With the exception at 15:30:590Z an SMB was released every time and another one at 16:00:58Z when the profile returned from 70% to 100%. For details I checked the tab-file with relevant columns extract6ed as shown below:

time		autoISF		orig parabola fit					ISFs							insulin Req		SMB	
UTC	bg	orig	emul	corr	dura	last-Δ	next-Δ	bg_accel	orig	prof	auto	high	rise	acce	emul	orig	emul	orig	emul
14:30:58Z	74	1	1						164,8	137,1	137,1	164,8	137,1	88,1	164,7	0	0	0	0
14:35:58Z	71	1	1	1	15	-2,8	-1,3	1,50	213,2	137,1	137,1	213,2	137,1	109,3	213,2	0	0	0	0
14:40:58Z	68	1	1	1	15	-2,8	-2,3	0,50	281,4	137,1	137,1	281,4	137,1	131,6	281,3	0	0	0	0
14:45:58Z	67	1	1	1	25	-1,3	-0,26	1,04	266,4	137,1	137,1	266,4	137,1	116,5	266,3	0	0	0	0
14:50:58Z	66	1	1	1	30	-0,55	0,4	0,95	295,1	137,1	137,1	295,1	137,1	126,9	295	0	0	0	0
14:55:58Z	71	1	1	0,99	45	2,91	4,62	1,71	216,6	137,1	137,1	216,6	137,1	106,1	216,5	0	0	0	0
15:00:58Z	81	1	1	1	15	10,2	15,7	5,50	103,8	128,6	128,6	103,8	128,6	66,4	103,8	0,33	0,33	0,1	0,1
15:05:19Z	91	1	1	1	15	11	13,5	2,50	102,5	128,6	128,6	102,5	128,6	90,2	102,6	0,47	0,47	0,3	0,3
15:10:20Z	101	1	1	1	15	10	10	0,00	111,8	128,6	128,6	128,6	111,8	128,6	111,8	0,37	0,37	0,2	0,2
15:15:58Z	113	1	1	1	15,1	11,43	12,31	0,88	110,1	128,6	128,6	120,2	110,1	111,8	110,1	0,48	0,48	0,4	0,4
15:20:58Z	124	1	1	1	25	11,49	11,95	0,46	110,1	128,6	128,6	110,2	110,1	119,2	110,1	0,41	0,41	0,3	0,3
15:25:59Z	138	1	1	1	25	13,05	13,93	0,88	99,8	128,6	128,6	99,8	107,7	111,8	99,8	0,61	0,61	0,5	0,5
15:30:59Z	149	1	1	0,99	40	9,96	9,65	-0,31	97,8	128,6	128,6	93,9	118	133,8	97,8	-0,17	-0,17	0	0
15:35:19Z	158	1	1	0,99	45	8,66	8,16	-0,50	97,5	128,6	128,6	91,1	115,6	137,5	97,5	0,39	0,38	0,3	0,3
15:40:19Z	163	1	1	0,99	45	6,98	6,25	-0,73	99,9	128,6	128,6	90,4	118,7	142,1	99,9	-0,29	-0,29	0	0
15:45:20Z	160	1,24	1,24	0,99	15	-2,2	-8,2	-6,00	226,1	128,6	128,6	90,5	127,5	321,4	226,1	0	0	0	0
15:50:20Z	153	1,36	1,36	0,99	20	-7,57	-13,29	-5,72	227,5	128,6	128,6	91	128,6	321,4	227,5	0	0	0	0
15:55:58Z	152	1,23	1,23	0,97	45	-5,41	-7,98	-2,57	138,2	128,6	128,6	92,1	128,6	193,1	138,2	0	0	0	0
16:00:58Z	152	1,22	1,22	0,97	15	1	4,5	3,50	56,4	90	90	65	90	56,4	56,4	0,16	0,16	0,1	0,1
16:02:27Z	152	1,22	1,22	0,97	15	1	4,5	3,50	56,4	90	90	65	90	56,4	56,4	0,2	0,2	0	0,1
16:05:58Z	146	1,32	1,32	0,92	25	-2,21	-1,75	0,46	65,4	90	90	65,3	90	83,5	65,3	0	0	0	0
16:10:58Z	132	1	1	1	15	-13,6	-20,6	-7,00	182	90	90	72,8	90	225	182	0	0	0	0
16:15:58Z	117	1	1	1	15	-16,4	-20,9	-4,50	196,3	90	90	81,4	90	216,9	196,3	0	0	0	0
16:20:59Z	100	1	1	1	15	-16,8	-18,3	-1,50	90	90	90	90	90	111,8	111,8	0	0	0	0
16:25:58Z	83	1	1	1	20	-17,51	-18,66	-1,15	134,3	90	90	134,3	90	97,2	134,3	0	0	0	0
16:30:58Z	73	1	1	1	15	-11,4	-7,9	3,50	108,5	90	90	108,5	90	56,4	108,5	0	0	0	0
16:35:58Z	70	1	1	1	15	-2,96	4,04	7,00	82,6	90	90	82,6	90	41,1	82,6	0	0	0	0
16:40:58Z	68	1	1	1	25	-0,47	4,2	4,67	109	90	90	109	90	50,1	109	0	0	0	0
16:45:58Z	63	1	1	0,99	35	-0,55	2,54	3,09	159,5	90	90	159,5	90	59	159,5	0	0	0	0
16:50:59Z	58	1	1	0,99	15	-5,6	-7,1	-1,50	225	90	90	305,1	90	99,7	305,1	0	0	0	0
16:55:58Z	48	1	1	1	20	-8,83	-11,11	-2,28	225	90	90	360	90	105,7	360	0	0	0	0

Legend: Green: single dominant effect Yellow: which two effects worked in combination

During the relevant time period the steep rise delivered the sole contribution three times totalling 0.9U and high glucose delivered the sole contribution once with 0.5U. From 15:00:58Z to 15:05:19Z glucose is still below target+10 and wants to weaken ISF but *bg _acceleration* is positive and jointly they release 0.3U. At 15:35:19Z glucose is high with a strengthening effect but *bg_acceleration* is negative which activates the *bgBrake_ISF_weight* and reduces that strengthening somewhat leading to another 0.3U. At 16:00:58Z bg_acceleration positive and dominates leading to another 0.1U.

Assembling this together in historical sequence gives these phases:

- 0.4U from low glucose and high *bg_acceleration*
- 0.9U from high delta, in this case high *pp_ISF* as can be extracted from the logfile
- 0.5U from high glucose
- 0.3U from high glucose but reduced by negative acceleration
- 0.1U from positive *bg_acceleration*

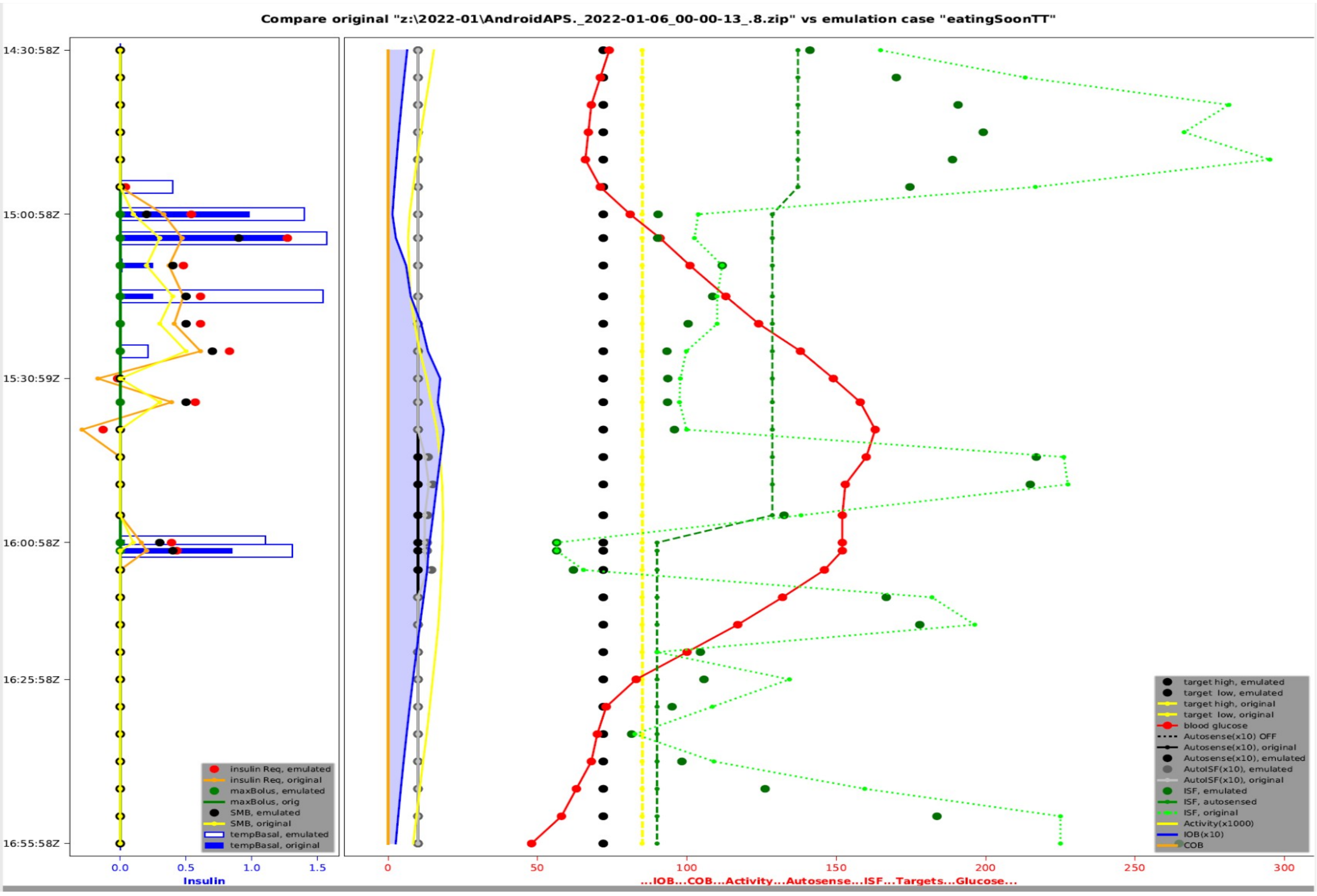
Keep in mind that the intention was to shift insulin delivery upfront. The IOB curve in the diagram above clearly shows the relatively slow build-up. In order to avoid the hypo the glucose should end up 20-30mg/dl higher which for a profile ISF of 90 means 0.3U less insulin. Alternative settings to be investigated are:

1. Increase initial insulin release

As long as glucose is below target initially the contribution from *acce_ISF* is capped and will not have significant impact without raising bgAccel_ISF to dangerously high levels. Therefore the obvious other approach is to use *Eating soon TT* like done by BerNie. The vdf-file is very simple:

```
profile          min_bg                72                        ### eating soon TT
profile          max_bg                72                        ### eating soon TT
```

The result looks like this:



Not surprisingly we clearly see more insulin required, basal and SMB. The extra amounts can be seen in the tab-file (extract):

time		insulin Req		-maxBolus-		---SMB---		--tmpBasal---	
UTC	bg	orig	emul	orig	emul	orig	emul	orig	emul
...									
14:55:58Z	71	0	0.04	0	0	0	0	0	0.4
15:00:58Z	81	0.33	0.54	0	0	0.1	0.2	0.98	1.4
15:05:19Z	91	0.47	1.27	0	0	0.3	0.9	1.26	1.57
15:10:20Z	101	0.37	0.48	0	0	0.2	0.4	0.25	0.01
15:15:58Z	113	0.48	0.61	0	0	0.4	0.5	0.25	1.54
15:20:58Z	124	0.41	0.61	0	0	0.3	0.5	0	0
15:25:59Z	138	0.61	0.83	0	0	0.5	0.7	0	0.21
15:30:59Z	149	-0.17	-0.02	0	0	0	0	0	0
15:35:19Z	158	0.39	0.57	0	0	0.3	0.5	0	0
15:40:19Z	163	-0.29	-0.13	0	0	0	0	0	0
15:45:20Z	160	0	0	0	0	0	0	0	0
15:50:20Z	153	0	0	0	0	0	0	0	0
15:55:58Z	152	0	0	0	0	0	0	0	0
16:00:58Z	152	0.16	0.39	0	0	0.1	0.3	0.03	1.1
16:02:27Z	152	0.2	0.43	0	0	0	0.4	0.85	1.31
...									
Minimum:	48					0	0		
Maximum:	163					0.5	0.9		
Totals:						2.2	4.4	0.27	0.52

The meal was taken at 14:13Z and selecting a TT of 60 minutes duration would have given 0.9U more insulin very early in the rise. From 15:15Z things would return to the behaviour in the original case. This would have had a significant impact on glucose levels, which the emulator cannot predict. So it is worth a try and check what happens towards the end of the absorption. However, before going ahead other meals must be analysed first to see whether they show similar behaviour.

2. Reduce insulin release during the peak and towards its end

As discussed before this makes no sense before the start of the absorption is under better control.

OK, so weit wäre ich gekommen, wenn nicht die 2 Korrekturen angefallen wären. Von der Eating Soon Variante hatte ich noch die älteren Auswertungen um sie einzubauen. Durch die Korrekturen ämndert sich die Basis, wobei die neue Basis, also das orig im Sinne des Emulators, nur als Ergebnis eines ersten Emulatorlaufs abgelesen werden kann. Im zweiten Durchlauf kommt dann die Emulation des Eating Soon TT und der Vergleich der beiden Emulationen sieht tabellarisch so aus:

time		insulin Req (emul)		SMB (demul)		tmpBasal (emul)	
UTC	bg	Neue Basis	EatingSoonTT	Neue Basis	EatingSoonTT	Neue Basis	EatingSoonTT
14:30:58Z	74	0	0	0	0	0	0
14:35:58Z	71	0	0	0	0	0	0
14:40:58Z	68	0	0	0	0	0	0
14:45:58Z	67	0	0	0	0	0	0
14:50:58Z	66	0	0	0	0	0	0
14:55:58Z	71	0	0,04	0	0	0	0,4
15:00:58Z	81	0,24	0,54	0,1	0,2	0,8	1,4
15:05:19Z	91	0,46	1,27	0,2	0,9	1,26	1,57
15:10:20Z	101	0,37	0,48	0,2	0,4	0,25	0,01
15:15:58Z	113	0,48	0,61	0,4	0,5	0,25	1,54
15:20:58Z	124	0,41	0,61	0,3	0,5	0	0
15:25:59Z	138	0,61	0,83	0,5	0,7	0	0,21
15:30:59Z	149	-0,17	-0,02	0	0	0	0
15:35:19Z	158	0,39	0,57	0,3	0,5	0	0
15:40:19Z	163	-0,28	-0,13	0	0	0	0
15:45:20Z	160	0	0	0	0	0	0
15:50:20Z	153	0	0	0	0	0	0
15:55:58Z	152	0	0	0	0	0	0
16:00:58Z	152	0	0,39	0	0,3	0	1,1
16:02:27Z	152	0	0,43	0	4	0	1,31
16:05:58Z	146	0	0	0	0	0	0
16:10:58Z	132	0	0	0	0	0	0
16:15:58Z	117	0	0	0	0	0	0
16:20:59Z	100	0	0	0	0	0	0
16:25:58Z	83	0	0	0	0	0	0
16:30:58Z	73	0	0	0	0	0	0
16:35:58Z	70	0	0	0	0	0	0
16:40:58Z	68	0	0	0	0	0	0
16:45:58Z	63	0	0	0	0	0	0
16:50:59Z	58	0	0	0	0	0	0
16:55:58Z	48	0	0	0	0	0	0

Wenn auch hier das TT 60 Minuten dauert, gäbe es sogar 1.0U mehr an SMB.