4. Meals: Setting ISF_weights in /Preferences V.18.1

Warning regarding importance of proper profile ISFs.

Starters on autoISF FCL who are coming from using HCL with **dynamic**ISF must be aware of the following: It is absolutely essential to build your FCL on properly set **profile** ISFs (likely a circadian pattern over 24 hrs). It may not apply to you, but many dynamicISF users did never bother to determine their ISFs that would maximize their HCL performance, but employ dynamicISF so to speak for going „dynamically“ through a wide range of possible ISFs, until eventually hitting a sweet spot, and the whole thing works better than before with their profile ISF (often only one, e.g. coming from Autotune).
The following is important to understand, as it also leads straight into the core idea behind FCL with autoISF, too: It is a good idea to establish a well-running hybrid closed loop with set (non-dynamic) **ISF (as in profile each hour of the day).** That ISF must be **aggressive enough** that it gets you down from a high around 200 mg/dl to target. That is roughly also the way you experimentally determined it (I hope).
Using that value also at lower bg, on the way up (after meal start), is very positive, as it is probably **stronger** than you would use, if you had just that (lower) bg to correct. autoISF will also do just that, but in a much more pronounced and elegant way.
On the way down from peak to glucose target, a somewhat too strong ISF will not hurt because much of the time your loop (well supplied with insulin before, „on the way up“) is zero temping or at least has only a small gap to correct from predicted bg to target bg.
You have no business to be much above 200 mg/dl where an even stronger ISF may or may not help. It sure does not help at an occlusion which is about the only reason to see super high values as a looper.
Pegging ISF strength to bg level therefore does not make much sense for loopers. You will use the autoISF toolbox to get strongest ISF **at low** but beginning-to-rise bg,

> Disclaimer: There are very much refined versions of dynamicISF that can have beneficial applications. But going to autoISF FCL, you absolutely must anchor on the proper profile_ISF (which in times of illness etc. you can temp. change via profile switch, also when using autoISF).

Warning not to simply copy settings from others

When setting **your** parameters, **don't use any given numerical example**, but data from **your** *successful* Hybrid Closed Loop!

Most *examples in this paper* are from an adult diabetic (Lyumjev, G6) whose insulin sensitivity can be characterized as follows: approximately 37 U TDD, thereof 13 U profile basal, at about 200g daily carbs from mainly lunch and dinner; no couch snacks or sweet drinks. The user also participates in multiple instances of daily moderate activity such as dog walking, biking and gardening. In Hybrid Closed Loop, a typical meal bolus was 8 U that was sometimes reduced such as when activity followed the meal.

We are still working on a section 4.6 in which you find a profile helper for some rough orientation and plausibility cross-checking.

Warning. Importance to starting from a well-performing Hybrid Closed Loop

A **satisfying performance in Hybrid Closed Loop** mode. is a pre-requisite. Expect to reproduce about the same %TIR also in your FCL, but with less daily interaction, once established.

Note that this refers to prior use of „vanilla“ software, without fancy „dynamic add-ons“ (such as: Autotune determined factors, dynamicISF etc). that probably will not be compatible with autoISF use, and may have introduced bias into the profile settings you bring with you into FCL now.

Part of satisfying HCL performance will have to be, that you currently **master your meal management.** This is a pre-requisite **to be able to forget it** … - because the initial tuning we now turn to demands that you analyze your prior best practice, in an attempt to „teach“ your FCL , find appropriate settings etc.

This is the main subject of this section 4 (finding settings for automatic meal management) and sections 5-6 (finding settings for highly automatic management also of other potential disturbances).

4.1 Getting started

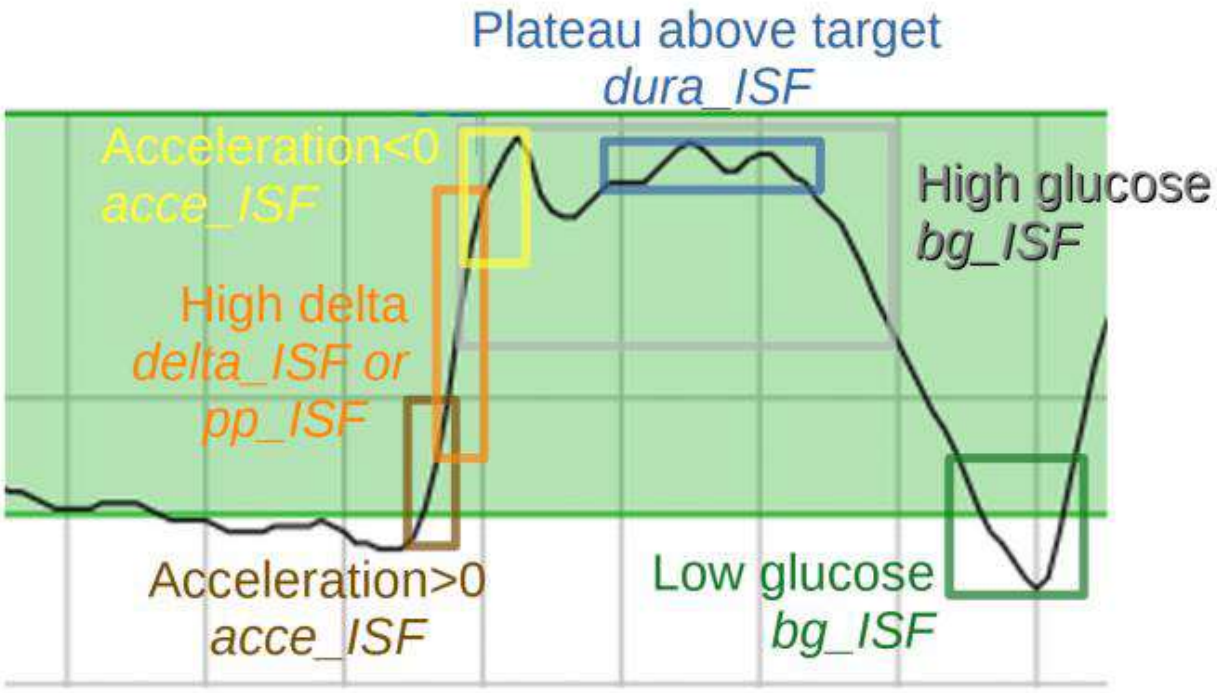
Make sure you have studied the preceding sections 1-3 on the general pre-requisites for FCL and on the workings of autoISF. Notably make sure you have set your iobTH (section 2.4 and *(if already launched* 4.6*)*)

In the early test phase, it is recommended to:

- switch FCL and autoISF ON only during daytime hours of a meal, *e.g. 11-18h*, for fully automatic "full closed loop" management of lunches
- take typical but not extreme lunches. Omit sweet drinks, or drink only slowly
- not use the activity monitor (see section 6.6), unless it is already well calibrated. If you consistently use an EatingSoonTT at meal start, this shuts activity monitor automatically off.

It is then essentially a matter of your UAM Full Closed Loop recognizing a meal start from the glucose trend, and ramping up iob.

When setting up your autoISF Full Closed Loop, you must set several ISF_weight parameters in AAPS Preferences/OpenAPS SMB/autoISF settings. They relate to different stages of the typical glucose curve after starting a meal:



Make sure you studied the related flowcharts in section 3. before you proceed.

Warning: **Any bolus you „sneak in“ will severely distort the glucose curve and can render your tuning of weights** (see below) **useless, if not dangerous**.

- Your FCL ideally runs without an insulin button at the bottom of the AAPS home screen.
- Issuing a bolus should kick you out of the FCL mode, back into Hybrid Closed Loop. *We are working on improving the User Interface (see e.g.* section 5.2*) that would facilitate and secure the bi-directional transitions*.
- How proper autoISF settings would differ for your meal management, depending on no-bolus, bolus like in HCL, or very small pre-bolus (Meal Announcement), is not well investigated at this point.

  Therefore it is best to remain consistent = use autoISF strictly for no-bolus FCL, and if you want to bolus for a meal, switch ISF adaptation to glucose behavior (autoISF) temporarily off.
- Maybe we are too cautious here, and in fact the autoISF adaptation to glucose behavior is tolerant enough of disturbances by user boli. Please report your findings in case you collect data of “mixed use” (FCL / Meal Announcement / HCL use with meal bolus). (A n=1 finding,

and guide how to evaluate, is reported here: https://github.com/ga-zelle/autoISF/blob/A3.2.0.2_ai3.0/To%20prebolus%20or%20not%20to%20prebolus.pdf).

Once we have a body of data, based also on users who moved from HCL with autoISF to FCL, we may need to re-define what the bi-directional transitions FCL < - > HCL in detail shall mean, and whether or not this has implications for needing different autoISF settings in /preferences for FCL and for HCL.

Please note that with autoISF 3.0 you are in an early-dev. environment, where the user interface is not optimized for safety of users who stray away from intended ways to use.

4.2 bgAccel_ISF_weight

When looping without carb inputs and without giving a bolus ourselves, the first crucial setting is to set the **bgAccel_ISF_weight** so that large SMBs are requested immediately when the loop detects an acceleration in your BG starting to rise.

Ideally within 15-20 minutes, after acceleration detection, as much iob is automatically supplied as we would have given with our bolus.
If you double the **bgAccel_ISF_weight**, the ISF strengthening is doubled.

Rule of thumb: Two of the first three SMBs each should be about ¼ (max 1/3) the size of a previous meal bolus in your HCL „career“.

Going over 1/3 can be problematic if your diet contains occasional low carb (or only snacking), and generally of course if your **CGM quality** is sometimes unreliable, and might produce an artefact that could be mistaken for a meal start. Be vigilant about this topic!

For hands-off FCL, your settings have to fit the whole **range of <u>your</u> meals**. In extreme cases you will have to balance too high running iob with additional carbs, and in the opposite case, you will have to reckon with temporarily exceeding the glucose target range and losses of the achieved %TIR for this day.

If your meals vary strongly, there are two avenues to ease your initial tuning job, or to optimize loop performance:

- Automations allow you to differentiate. For instance it is possible to apply different iobTH_percent and/or different bgAccel_ISF_weights for meals in different time windows or geo locations (details see sections 3.4 and 5.1)

In case you use autoISF 3.0 on the iAPS platform for i-phones, you need to use a third party automation software (see case study 4.X <- *call for an iAPS user to share his !!*)

- *(if already included)* autoISF 3.0 provides the option to pre-program settings for **4 different meal type clusters** (details see in section 5.2.3.1 (4) and 6.3).

In search of appropriate settings for your (one or up to 4) meal cluster(s), you must keep (real-time) track of the SMB tab when tuning. This can be impractical. You probably will end up making a lot of screenshots (quickly in the crucial minutes where the SMBs were given,or when you thought they should be given), for later analysis.

The superior method is to just copy logfiles every couple of hours from your phone/internal memory/AAPS/logs (all zip files there), and analyze them at your convenience later, using the emulator (see section 10). Some emulator-based analysis is also possible within AAPS on your phone (section-11).

Already when tuning the **bgAccel_ISF_weight** it can become evident that safety restrictions must be widened further (as discussed in section 2).

Especially if your profile basal rate is very small, the **smb_delivery_ratio** and/or the **smb_max_range_extention** "must" often be increased further.
Furthermore, the **smb_delivery_ratio** provides more leeway to increase the aggressiveness (e.g. 0.6 -> 0.72 results in another +20%).

In any case, it is worth the effort to tune the **bgAccel_ISF_weight** in such a way that high glucose increases are already nipped in the bud, so to speak.

This also facilitates the tuning task for the subsequent phases of the meal, because there is then largely zero-temping, as well known from HCL-times after YOUR administered bolus. Also, the lower and shorter lasting the glucose peak, the lesser the hypo danger from the activity tail of SMBs given when glucose was „stuck“ high.

Default bgAccel_ISF_weight is set to zero in autoISF. **To start**, I would try 0.05 or **max 0.1**, and keep trying in max 0.05 steps. Soon move to 0.02 steps (which still means 10-20% change). From my (very limited) overview, many use around 0.2, but possibly higher if their hourly basal rate is 0.1U or lower. (*Consult* section 4.6 *when available*). Do not be tempted to rush this setting by using large jumps in adjustments.

Ideally, one should set the bgAccel_ISF_weight so that for meals that are in the **lower** range of the "fast **carb load**" of your cluster, the necessary insulin supply is already approximately provided with 3 SMBs. The glucose curve, at such meals, begins to flatten early in this SMB phase, so a deceleration follows very soon.

4.3 pp_ISF_weight

With **high Carb** meals, or meals that come with a sweet drink, the acceleration phase will last longer, and BG will rise further, which will require a higher insulin supply.
Between acceleration and deceleration there is also a more or less linear further increase of insulin need.
Our autoISF should now "fight" this with the help of the post-prandial ISF, set via **pp_ISF_weight,** after we have set a halfway suitable bgAccel_ISF_weight.

Select **pp_ISF_postprandial all day = ON**

> In full closed loop mode, this parameter is preferred over deltaISF ((.. and highly beneficial for managing meals with gastroparesis)).

Tune your **pp_ISF_weight** after you have set a halfway suitable bgAccel_ISF_weight. You should check meals in the upper spectrum of your g carb, carefully starting with a weight of 0.01.

Normally the SMBs triggered by bgAccel_ISF_weight and pp_ISF_weight should be sufficient to reach and slightly exceed the **iobTH** (see section 2.4) so all the other autoISF parameters are relatively unimportant for now.

> A reason why this can work at all, also for quite a variety of meals, lies in the fact that there is an hourly carb absorption limit of about 30g/h (reference: Dana Lewis: https://github.com/danamlewis/artificialpancreasbook/blob/master/8.-tips-and-tricks-for-real-life-with-an-aps.md#heres-the-detailed-explanation-of-what-we-learned). So while meals might wildly vary in composition and size, what is digested and needs insulin in the first <90 minutes (when FCL tries to catch up with insulin need and differs strongly from HCL, and bgAccel_ISF and pp_ISF play the leading role) will be relatively close (…for meals with similar initial glucose acceleration and rises, anyways, and *that* is information the loop does have).

Depending on the type of meal and "aggressiveness" of your bgAccel_ISF_weight and pp_ISF_weight tuning, the iob will already be so high that in the phase of decelerated glucose rise towards the peak (the "last part of the rise") that no insulinReq is seen by the loop.

Therefore the **bgBrake_ISF_weight** is often unimportant.

Warning: Occasionally consult the SMB tab to see how your settings really work.

A setting that is actually too aggressive might be masked. **Tuning only works if** the effects of the settings being tuned are **not unintentionally limited by other** (e.g.„safety“) **settings**.

Also, always look at two or three different meals before deciding whether a tuning "fits" („good enough“ for each of them)

Case Study 4.1 (Pizza Meal) contains, towards the end, an example how you can go about tuning the _weights for various _ISF factors of autoISF.

4.4 bgBrake_ISF_weight

At a low carb meal, or an attempt at doing a weight reduction diet, the glucose goes up only sluggishly and iobTH should not be reached at all.

Acceleration and the phase of strong glucose rise are quickly over in these cases, and there is mainly a decelerating bulge of insulin action that projects over the next few hours.

Now the importance of the **bgBrake_ISF_weight** comes in. In full loop, the bgBrake_ISF_weight is often only about half as large as the bgAccel_ISF_weight (but that would also depend on your personal diet pattern and eating/digestion speed). Also here, one should approach the tuning gradually, increasing the weight from small values.

Case Study 4.2 shows a user example of a low carb meal managed in FCL by autoISF.

What is very helpful for us in any case is that the loop calculates the situation every 5 minutes, and corrects it.
However, if there is too much insulin in the system, the loop can only correct to a very limited extent, namely only to the extent that it can set basal to zero.
Therefore, the core problem is that the Full Closed Loop must build up iob very quickly, but not too much, in the initial phase of a meal.

But high BG values (out of range, >180 mg/dl) can not always be avoided..

Note regarding acceleration happening in late part of dropping glucose:

In version 2.2.8.2 there was a potential deficiency in situations where glucose was falling and the glucose acceleration was already positive. That meant a minimum glucose level can be extrapolated. If that happens to be less than target and expected in less than 15 minutes then there should be no strengthening of ISF as it would lower glucose even more. Therefore bgBrake_ISF_weight is used now instead of bgAccel_ISF_weight. But those situations were rare and less critical than might be expected at first sight. The reason is that in most cases the predictions ended up even below their threshold meaning SMB were disabled.

4.5 High Glucose Values and dura_ISF_weight

With large or high fat/protein meals, a 2nd hill of glucose will form, or a long high plateau.
For such situations there is in autoISF the modulation of ISF depending on BG level or duration of plateau formation.

High BG values and a plateaus in BG values are tuned using the **dura_ISF_weight** and associated parameters. This feature is also very useful in Hybrid Closed Loop. It elegantly manages, fully automatically, temporary fatty acid resistance. Please refer to other papers for details (for instance, section „Late stage of meals“ of: https://www.facebook.com/download/649096606100188/MealMgt.Basics_09Dec21.pdf).

Since in Full Loop we "turn up" our loop to give the maximum SMB size we can at the beginning of a rise, it is crucial to resist the temptation to continue with a particularly strong ISF in the meal phase with the highest glucose values .
This is a reason why in Full Loop we do not make much use of the **bg_ISF** component of autoISF. Near glucose peak, zero-temping usually prevails anyway, so the settings we try may not be used by the loop.
This also means that too aggressive settings might not come into play most of the time. However, some other time they might come into play, and produce a hypo 1-2 hours later.

Therefore, **study carefully the SMB tab to see what the selected weights would do, if there were no zero-temping at the time**. Also, try a completely different meal to see how your settings work there.

> The UAM Full Closed Loop doesn't have any clues as to how many grams of carbs will be absorbed late. Not knowing when your steady-state max carb absorption phase (the earlier mentioned 30g/h), and even sometimes a brief episode of insulin resistance to fats, end, the FCL will struggle to provide desired amounts of insulin, facing potential hypo danger later because of the DIA of the insulin in use.
>
> Actually, the UAM Full Closed Loop is not completely clueless regarding how carb absorption will go on. It will work with a prediction of further carb absorption building on the

carb deviation (=hypothesis of how much got absorbed in the past 5 minute segments), and phase out more carb decay in the course of the next 1 to max 3 hours. For more detail see https://openaps.readthedocs.io/en/latest/docs/While%20You%20Wait%20For%20Gear/Understand-determine-basal.html#understanding-the-basic-logic-written-version (or study your SMB tab info).

This UAM prediction about further carb absorption can be worse, but can also be better than a prediction based on the user‘s „e-Carb“ input in Hybrid Closed Loop.

In any case, and even when having perfect knowledge about how exactly the carbs fade out in the next hours, there would still be a principal problem for the loop: Heavy insulin „fire“ against highs will not work immediately (depending on the insulin’s time-to-peak), and notably it comes with a significant hypo danger (from the „tail“ of insulin activity.)
A big bolus, or also a series of boli, will rarely work exactly for several hours matching the absorption of carbs (from what, how much and and how fast the user ate).

Once your BG sits high, neither you, nor a hybrid closed loop with all the carb info, nor your FCL can work wonders.Resist the temptation to elevate the **dura_ISF**_weight very high.
Also, the author is sceptical about using the **bg_ISF** (at least be careful, use small weight, or shut-off).
Highs will take time to resolve. Interestingly, an after-dinner walk can work wonders sometimes.

As often, the best solution is to be pro-active:

The earlier large SMBs come (driven by bgAccel_ISF and pp_ISF), the less high the overall increase in BG will be, and (provided you set a proper iobTH) **the lesser the risk will be for a hypo after the meal**.
Therefore, put most of your tuning effort into determining suitable weights for bgAccel_ and for pp_ISF and iobTH.

Your FCL cockpit *(when fully developed and launched)* will give you easy access to tweek 2 of these 3 essential parameters (see section 5.2.3.1 /TT dialogue field), providing you an opportunity for more research on the fly, so to speak.

The experience of the author is that it is possible to tune the above mentioned weights for very different meals in such a way that the glucose almost always remains acceptably in range.

However, if you need **differentiated settings** for different meals or meal time clusters, you can

- either use Automations to address this (see section 5.1.3)
- or pre-program 4 different clusters in /preferences, and call them up within a second from the AAPS home screen (*when cockpit functions made fully available,* see section 6.4.3)

4.6 Profile helper

xls based tool is still under development / chapter will follow later