4. Meals: Setting ISF_weights in /Preferences

v.4.3

Please note that with autoISF 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. Good safety features exist, but these are only as good as the development-oriented user understands and implements them. This is not a medical product, refer to disclaimer in section 0

Available related case studies:

Case study 4.1: Pizza
Case study 4.3: Hands-off FCL on Xmas

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 what they had used as a profile ISF (often only one, e.g. coming from Autotune).

The following is important to understand, as it 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** (set in **profile** for 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. See https://github.com/bernie4375/HCL-Meal-Mgt.-ISF-and-IC-settings/blob/HCL-.-settings-main-repo-(pdf)/ISF%20determination_V.3.33.pdf).

- Using *that strong* value also *at lower bg,* (on the way “up” , after meal start), is very positive: We do *not* want to have a *softer* acting loop when at *lower* bg (which is what dynamicISF tends to do!). autoISF will, in contrast, temporarily sharpen your ISF when, at low bg, acceleration is detected..
- On the way down from peak, towards glucose target, a somewhat too strong ISF should 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 an experienced looper.

Pegging ISF strength to bg level therefore **does not make sense in FCL**. You will use the autoISF toolbox to get strongest ISF **at low** but beginning-to-rise bg,

> Note: There are very much refined versions of dynamicISF that can have beneficial applications, notably in HCL.

Going to autoISF FCL, you absolutely must anchor on the proper profile_ISF.

When using autoISF you can – as you did in the past, e.g. around exercise, or in times of illness – temporarily modify your profile ISFs, via a **%profile switch**. Also the other two top buttons, exercise and TT, can be used to adapt to changes in sensitivity/resistance. More about that in section 5.2.2.2 . But, first, spend a couple of days (if not weeks) to get your key autoISF related settings right, strictly on/for days with your normal insulin sensitivity. This is what this section 4 is about.

Warning: Do not copy settings from other FCL loopers

When setting your parameters, don't use any given numerical example (not even as “a starting point”). Instead, anchor on **data from your *successful* Hybrid Closed Loop!**

> Most *examples given 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 exercise 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.

After seeing some more inputs from a variety of users we might put together a profile helper for some rough orientation, and for plausibility cross-checking, in section 4.8

Warning. Importance of 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 may have introduced bias into the profile settings you bring with you into FCL now.

To reach a satisfying performance you must start from a hybrid closed loop in which you did **master your meal management well** using the oref(1) algo SMB+UAM**.**
This is a pre-requisite **to be able to forget it** … - because the initial tuning that we now turn to demands, that you analyze your **prior best practice as your blueprint** to find appropriate settings and „teach“ your FCL to come up with the necessary iob.

This is the main subject of this section 4 (finding settings for automatic meal management).

Section 5 will explore avenues to manage “disturbances”, i.e. time blocks or situations that might demand enhanced or reduced loop aggressiveness.
Section 6 will focus on the exercise mode, and the activity monitor.

Resist the temptation to make use of the tools presented in sections 5 and 6 too early.
On your **first** setting-up and tuning attempt, **it is strongly recommended that you not “play around” with all ultimately available features, but stick to the sequence of steps** to take.

Yes, “playing around” with the many extra buttons often will help find an improvement. But you likely create an instable FCL that, already at fairly standard situations, uses up some of your FCL’s principal capacity to correct for disturbances. This limits what will be left to manage extreme situations.

Also, **once you created a maze of little errors and counter-strategies/counter-errors**, **it will be nearly impossible to find your way** out of this mess, **towards better settings**, at any later point of time.
AutoISF comes with very many extra parameters, and even when employing the emulator (sections 10 and 11) it is quite hard to analyze their interaction.

One principal reason why things are difficult to analyze is, that you really can only analyze one change, and that will put you on another bg curve. So, you can never see the full effect, along more than 10 minutes, that *any* change will ultimately result in.

PS: Section 11.4 describes the ultimate tool to investigate "what-if" regarding a setting change you may contemplate.

4.1 Getting started

Make sure you have studied the preceding sections 1and 2 on the general pre-requisites for FCL and section 3 on the principal workings of autoISF.

Caution: This entire e-book is about Full Closed Looping. **In case you intend to work with giving boli**, many suggestions made - notably in this section 4, and in section 2 – should **not** be followed. You would have to **do extra research**, on your own data, how your bolus changes things. (See also section 7, and discussion on pre-bolussing, ~2 pages down)

Make sure you have appropriately:

- **widened the SMB size restrictions** (section 2.1),
- **elevated** the max allowed ISF amplification with your set **autoISFmax** (section 2.2)
- **set** your **iobTH%** (refer to section 2.4 and if available 4.8)

In the early test phase, it is recommended to:

- Run the system as dummy, not connected to your body (or, on own risk, connect only as long as you watch closely)
- In AAPS preferences, switch your autoISF FCL (= **autoISF/"Enable adaptation of ISF to glucose behavior"**) ON only during daytime hours of a meal, *e.g. 11-18h*, for fully automatic "full closed loop" management *of lunches*.

  You can do this switching manually *at 11 h and 18 h every day*, *or* set up an Automation that does that (see section 3.4).
- Take typical but not extreme meals. Omit sweet drinks, or drink only slowly.
- Occasionally, watch the time-pattern of bg, iob (SMBs given), and insulin activity after meal start. Aside from serious "mathematical" attempts to tune settings based on data from the SMB tab (or the emulator, section 10), just watching the curves develop on your AAPS main

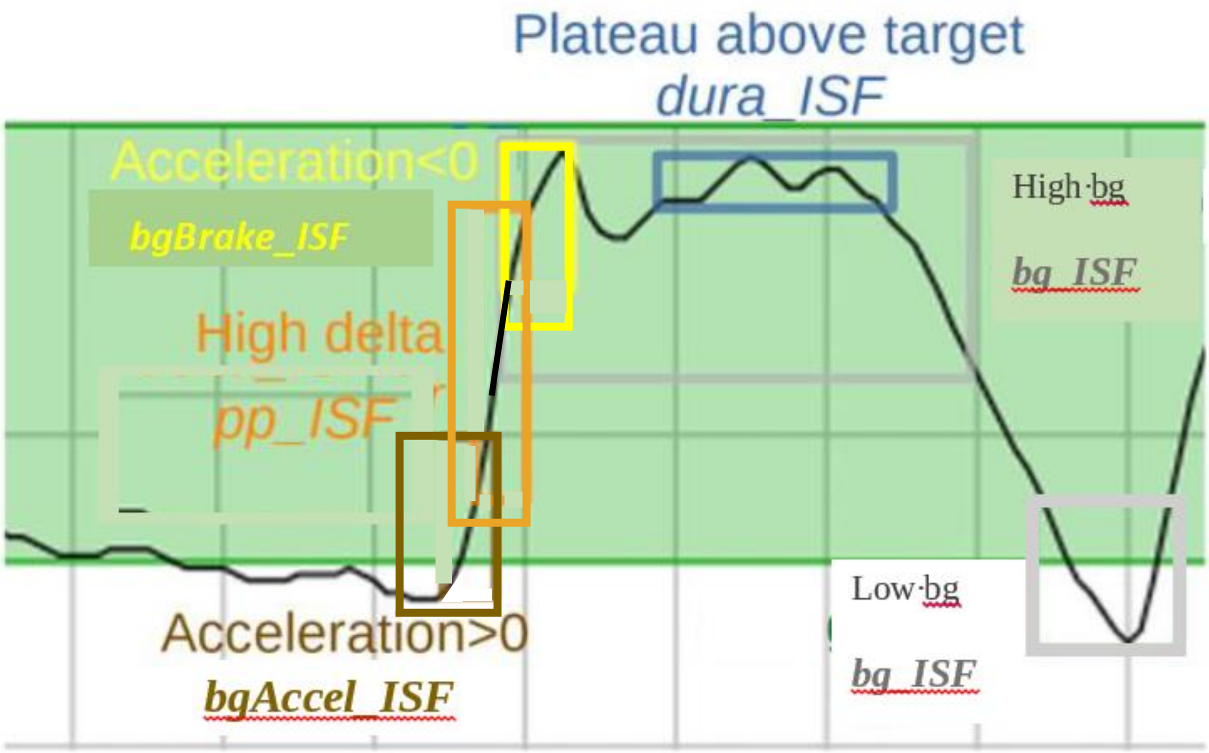
screen can, over time, give you “a feel” what settings, and eating behaviors, are benign or detrimental to good %TIR performance.

- It is wasted time to “optimize” settings based on 1 type of meal. You need a “good enough” compromise that works with your range of usual meals. See case study 8.2

- Do not use the Activity monitor (see section 6.6), unless it is already well calibrated.
  In case you use an EatingSoonTT at meal start, note that any active TT shuts activity monitor automatically off for a while.

The core challenge of your UAM Full Closed Loop is to recognize 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:



Note: ***bg_ISF*** is not used much in FCL, as it is rather late to act on high (or low) bg level that developed. But, feel free to experiment, e.g. in case you have indications, in your data, that in the past dynamicISF was useful to manage bg extremes in some situations.

The core advantage of using autoISF with oref(1) SMB+UAM (in FCL as well as in hybrid closed loop) is that it manages the glucose curve it sees developing, **no matter what the underlying reason** is.

42 potential factors were identified (see: https://github.com/bernie4375/HCL-Meal-Mgt.-ISF-and-IC-settings/blob/FCL-w/autoISF/42%20factors%20influence%20bg.pdf), so, no wonder, that loopers who meticulously input their carbs will often *not* see the expected result.

Reminder: autoISF has that advantage only if the pre-requisites (section 1) are given, notably a very fast insulin, and reliable CGM and insulin delivery (not leaking, and permanently Bluetooth connected).

Before you progress, make sure you studied the flowcharts in section 3 that describe how autoISF calculates the **effective**(ly used) **ISF**.

Consult sometimes your SMB tab, to see how the applied effective ISF (named **sens** there) is calculated. (Example given in section 5.4.5).

Warning: **Any bolus you „sneak in" will severely distort the glucose curve. That could** render your tuning of weights (see below) useless, and could **make your loop act in unpredictable ways (**potentially also dangerous, however, your set iobTH (section 2.4) should help here, too).
In case you feel tempted to use boli, be ready for some own extra research, and refer to section 7.

After doing the prep work as outlined in section 2 **you now get to calibrate your FCL to your normal meal spectrum** by initially **setting and tuning the various _ISF_weights**, that dynamically change with bg curve characteristics as sketched in the chart on the previous page.

Please stay away from extremes (regarding both, meals and exercise) **when you go through this** section 4. It is about getting a first *roughly right* set of settings, as a basis.

Researching your standard meal patterns, and finding settings for the various -ISF_weights is the core job in setting up your autoISF FCL.

Depending how varied your diet and general lifestyle are (and your expectation of %TIR you like to reach), this could be the main job at hand. However, there is much more you *could* do *later*, and that will be outlined in later sections 5 and 6.

4.2 Meal detection and managing the initial bg rise: bgAccel_ISF

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 blood glucose (bg) that is starting to rise.

Ideally **within about 20 minutes after acceleration detection, which would be the first up to 4 SMBs, as much iob should automatically be supplied as we would have given with our bolus in hybrid closed loop.**

Insert here. BLUEPRINT ANALYSIS   TOTAL IOB FOR MEAL HCL => FCL

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

> **Going over 1/3 would 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 each of your meal times, e.g. should suit (nearly) all your lunches that you tend to have.

> *Between* your daily mealtime slots, your circadian ISFs make a differentiation.

In extreme cases you will have to balance too high running iob with additional carbs (a late additional snack against going too low), and in the opposite case, you will have to reckon with temporarily exceeding the glucose target range, and losing some %TIR for this day.

If your meals vary very strongly, there are avenues to ease your initial tuning job, or to optimize overall resulting 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 on the iAPS/Open iAPS platform for i-phones, you may need to use a third party automation software, or "middleware" (! call for a case study 4.X)

- You can pre-program **custom buttons for special** meal (or snack) **types**, with different underlying FCL settings (see "cockpit", section 5.2.2.3)

- *Skip what is in **green writing***:
  = Drafted fragments or not implemented ideas. Please contribute, or wait for update with the missing info

  In an update, autoISF 3.x might provide the option to pre-program settings for 4 different meal type clusters, accessible from the TT button (presented in section 5.3.3.1 (4) and 6.3).

- You can **modulate FCL aggressiveness manually** making use of temporary switches of %profile and/or set for a a couple of minutes an odd (=>SMB off) glucose target (section 5.2.2.2)

Experimenting with the three above mentioned “avenues”, the author found:

- the third easiest to occasionally use, and the first one hardest.
- it worth investing some effort (also using the emulator a couple of times) to iterate through the typical meal spectrum a couple of times, for finding a “good enough” set of .._ISF_weights and other settings (like autoISFmax, iobTH% etc), *and **not do** much extra differentiation.* (More see in section 5).

It is certainly worth trying hard at finding a good set of ISF_weights for your meal spectrum, to keep interventions in daily life to a minimum.

In search of appropriate settings, 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 after a SMB *was* given, or when *you thought it should be* given), for later analysis.

The superior method is to just copy **logfiles**

... from autoISF 3.0.1 onwards, about every 2 weeks should suffice...

from your phone/internal memory/AAPS/logs (all zip files there), and analyze them at your convenience later, using the **emulator** (see section 10; used e.g. in last pages of case study 4.1). 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 (as discussed in section 2) must be **widened** further:

- Especially if your *profile basal* rate is very small, the **smb_max_range_extention** and/or the **autoISF_max** "must" often be increased further.
- Pay attention also to the **iobTH**% and, potentially, iobMAX
- Note that the smb_delivery_ratio “only” portions the insulinReq differently over the next 15 minutes (see also section 2.3), and therefore is not a prime tuning parameter.

In the end you should **not set these safety limits too tight,** so “nudging” aggressiveness by another 10 or 20% from your cockpit, later, will not bounce into restrictions.

On the other hand, setting **narrower** restrictions for max allowed SMB size can also become necessary:

- You don’t want your loop bounce, regardless of the carb load, “immediately” into your iobTH limit (and up to 30% above), which is not desirable if your meal spectrum is very varied
- Poorer CGM quality demands narrower restrictions, too, for safety reasons.
- If you use a 1-minute CGM (Libre 3) please observe section 1.4.2

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.

> Remember: In FCL, the first 3 or 4 SMBs should not be much delayed, and amount to similar iob like your "former boli in HCL".

Early strong iob also **eases 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.

bgAccel_ISF_weight is set default 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. From my (very limited) overview, many use around 0.2, and possibly even higher if their hourly basal rate is 0.1U or lower. (Consult section 4.8 when available). Do not be tempted to rush this setting by using large jumps in adjustments.

How changing the _weights influences the resulting calculated insulinRequired

To get a feel for how changing the _weights influences the resulting calculated insulinRequired, it is best to start cautiously and just do 10 to max 20% steps up, and watch out for the effects. Doing similar step sizes should yield about similar effects each time.

Example 1: Going from bgAccel_ISF_weight of 0.2 to 0.16 (20% less).

If your profile_ISF is 40 mg/dl/U and with bgAccel_ISF_weight = 0.20 you saw acce_ISF factor of 1.31, this would ((if the acce influence dominates and is used as effective ISF)) lead to the effectively used ISF of 40/1.31 = 30.53 mg/dl/U. For an intended correction by – 10 mg/dl the insulinRequired would calculate to 10 / 30.53 = 0.328 U.

Now, going with a 20% reduced bgAccel_ISF_weight of 0.16:

*acce_ISF = 1+ bgAccel_ISF_weight * internalFactor*

*before 1,31 = 1 + 0.20 * iF => 0.31 = 0.20 * iF => iF = 1,55*

*after ? = 1 + 0.16 * iF => ? = 1 + 0.16 * 1.55 = 1.25*

New effective ISF would be 40 / 1.25 = 32.05 mg/dl/U. For an intended correction by – 10 mg/dl the insulinRequired would calculate to 10 / 32.05 = 0.312 U, which is 4.9% less.

Example 2: Going from bgAccel_ISF_weight of 0.2 to 0.10 (50% less; or doubling in the other direction).

If your profile_ISF is 40 mg/dl/U and with bgAccel_ISF_weight = 0.20 you saw acce_ISF factor of 1.31, this would ((if the acce influence dominates and is used as effective ISF)) *lead to the effectively used ISF of 40/1.31 = 30.53 mg/dl/U. For an intended correction by – 10 mg/dl the insulinRequired would calculate to 10 / 30.53 = 0.328 U.*

Now, going with a 50% reduced bgAccel_ISF_weight of 0.10:

*acce_ISF = 1+ bgAccel_ISF_weight * internalFactor*

*before 1,31 = 1 + 0.20 * iF => 0.31 = 0.20 * iF => iF = 1,55*

*after ? = 1 + 0.10 * iF => ? = 1 + 0.10 * 1.55 = 1.155*

New effective ISF would be 40 / 1.155 = 34.63 mg/dl/U. For an intended correction by – 10 mg/dl the insulinRequired would calculate to 10 / 34.63 = 0.289 U, which is 12 % less (going the other way, 0.328 is 13.5 % more).

Example 2 (-50%) reduces _weight 2.5 times lower than example 1 (-20%), and the resulting effect (-12% vs. -4.9% insulin Required) is also factor 2.5 different.

Note: “Your” internal factor “iF” might differ; for sure it is very different between the various …_ISF components. (Also, never forget to look into how other .._ISFs play into the effective_ISF which overall results).

Ideally, one should set the bgAccel_ISF_weight such, 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 de-celeration **(braking)** follows very soon (-> section 4.4).

Note regarding acceleration happening “again” in late part of dropping glucose

After the peak, in the late stage of *falling* bg, the glucose curve is like an accelerating parabola again. The algorithm tries to evaluate when and at which bg level complete digestion of the meal and a bg minimum will result. Insulin required to stabilize around target bg is usually very small, and the adaptation of ISF in that stage relatively unimportant.

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.

4.3 Managing strong bg rises: pp_ISF

With **higher carb load** 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 a more or less linear further increase of insulin need in these cases.

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.

Tune your **pp_ISF_weight** *after* you have set a halfway suitable (not too aggressive) bgAccel_ISF_weight. You now should check meals in the upper spectrum of your g carb load, and carefully start with a starting value for *pp_ISF_weight* of 0.005. Observe the reactions and check the SMB-tab before you increase it with care for the next day.s

> Best practice is to analyze the emulator tables (discussed in section 10, and example given in the pizza case study 4.1)

Normally (except for very low carb meals) 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. (That limit can be lower, e.g. with gastroparesis or certain medications, but that would make things even easier)

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, with bgAccel_ISF and pp_ISF in the leading role), will be relatively close…for meals with similar *initial* glucose acceleration and rises, anyways…

The others, **low carb** with much slower initial acceleration and rise, are easy recognized as different by the loop, see section 4.4 that follows.

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"), no more insulinReq is seen by the loop.

Therefore the **bgBrake_ISF_weight** is often unimportant in meals with a relevant carb content.

For potential relevance in low carb meals, see section 4.4.

Warning: **Occasionally consult the SMB tab to see how your settings really work.**
A setting (...ISF_weight) that is actually set 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). You probably will have to iterate back and forth doing this for two or three different kinds of meals …

- 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.
- Case Study 8.2 shows that it is not worth it to seek “optimized” settings based on just one meal.

… until you find *one* good enough set of settings *for all* of them. Do not rush this, establishing a solid foundation will be well worth your time.

4.4 Sluggish rise towards bg peak: bgBrake_ISF

At a **low carb** meal, or an attempt at doing a **weight reduction diet**, (and probably also with gastroparesis, or if you take one of these novel GLP-1 drugs that slow meal absorption - Somebody, please supply a case study!) the glucose goes up only sluggishly, and iobTH should not be reached at all.

In case you *exclusively* do very slow absorbing meals, you could of course also adjust your iobTH setting low enough to suit your *uniform* situation.

Acceleration, and the phase of strong glucose rise, are quickly over at slow-absorbing meals, and there can be:

- a decelerating bulge of insulin action that projects over the hour or longer. This is where the importance of the **bgBrake_ISF** can come in.
- a bg curve that hovers for an hour or longer around an elevated bg level, because additionally absorbed carbs and consumation of the moderate SMBs delivered tend to keep

a balance for a while. **Dura_ISF** can deal with this (see next chapter). An example for this is given in Case study 4.2.

Note that in some data outputs (e.g. the csv/xls tables coming from the Emulator, e.g. in Case study 4.2, big table at the end there), you will see only "**acce_ISF**" results.

- In case of positive acceleration, these are driven by the bgAccel_ISF_weight setting, and results are >1.
- I**n case of negative acceleration** (decelerating rise), **bgBrake_ISF_weight is applied**, , and results are < 1. (Example see in graph in section 10.3.3.3).

In full closed 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 coming from small values.

Please observe that **this tuning must strictly be done with types of meals for which there is insulin need at de-celerating but still rising bg**.

bgBrake_ISF is totally irrelevant for hi carb meals where your loop shot over iobTH already by the time your rising towards the bg peak slows down!

Likewise, if your initial bgAccel_weight is set so strong that your first SMBs catapult you over the iobTH, no matter what type of meal: Then you must **first** find a reasonable setting for this parameter, one that works "good enough" to control your carb loaded meals, but still leaves room for milder loop response at low carb meals.

In case you cannot quite get all the ISF_weights "right" so the occasional low carb meal will not get over-treated: Avenues to adapt your loop aggressiveness are discussed in section 5. For instance you will be able to (if needed);

- use a temp. reduced %profile
- temp. lower iobTH or bgAccel_ISF_weight
- construct for yourself an extra snack or low carb button ("DIY cockpit") with an underlying suitable Automation

In the **late stage of still rising (!) glucose**, the Full Closed Loop typically sharply reduces SMBs already because it is "painfully aware" of the following principal conflict:

- iob (like formerly given in HCL via your bolus) must go high quickly, in order to limit the high
- However, if there is too much insulin in the system, a **hypoglycemia can happen later** within the DIA time window, because the loop can, later, 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, and high bg values (out of range, >180 mg/dl) can not always be avoided.

4.5 Plateauing and High Glucose Values: dura_ISF and bg_ISF

4.5.1 dura_ISF

With **large or high fat/protein meals,** a 2nd hill will form in the bg curve, or a long high plateau. For such situations, autoISF features the modulation of ISF depending on bg level and duration of **plateau** formation.

A (in that case, often not-so.high) plateau can also form in **low carb meals**, when, basically, carb and insulin "burn rates" might keep a balance over an hour or longer, requiring occasional moderate size SMBs.

So, depending how your personal diet spectrum looks, you need to tune-in your dura_ISF primarily with large hi-FPU meals, or for meals at the low carb end of your diet.

Absolute "pros" could also primarily calibrate their dura_ISF for low carb. Dura_ISF has in-built amplification at higher bg levels. If needed this could be further boosted for much higher plateaus developing after greasy feasts:

- by adding an Automation that gives an extra boost "against" the temporary insulin resistance associated with fats (via increasing the baseline, in terms of a temp.130% profile switch, for instance.
  Compare at: https://androidaps.readthedocs.io/en/latest/Usage/FullClosedLoop.html#stagnation-at-high-bg-values),
- *or by* making additional use of the bg_ISF (or dynamicISF) (-> Tune it in parallel).

Conditions for dura_ISF to become active:

1) glucose is varying within a **+/- 5% interval** only;
2) the average glucose (*dura_ISF_average)* within that interval is **above target**;

3) this situation lasted **at least for** the last **10 minutes**

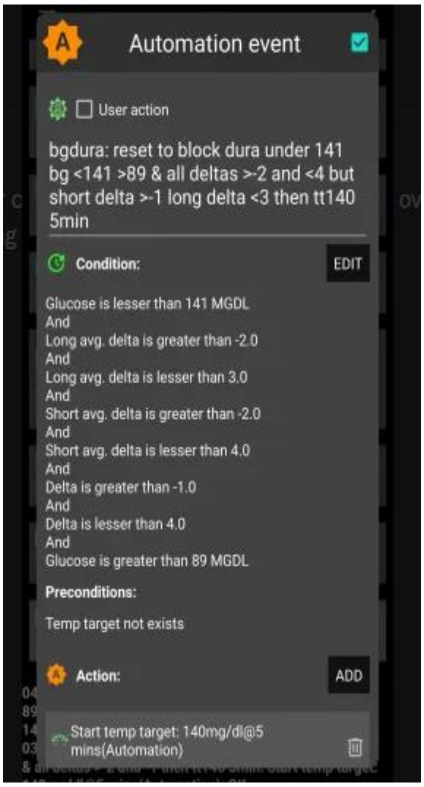
Effect: Formula is given in section 3 (-> Quick Guide Github/ga-zelle)

4) The strengthening of ISF is stronger the longer the situation lasts, and the higher the average glucose is above target:
5) This can be individually tuned by the **duraISF_weight to automatically manage** hgh plateaus in bg values

This feature is also very useful in Hybrid Closed Loop. It can be used to elegantly manage, fully automatically, a temporary insulin resistance from fatty acids. Please refer to other papers for details (for instance, section „Late stage of meals“ of “Meal Management Basics”, available here: https://github.com/bernie4375/HCL-Meal-Mgt.-ISF-and-IC-settings).

Set a **start value of 0.2** for your dura_ISF_weight, and increase only cautiously with an eye on hypo prevention 2-3 hours later.

Caution: Fine tuning this parameter only makes sense ***after*** you tuned your bgAccel_ISF and pp_ISF well (so your thin yellow insulin activity curve shifts *as far to the left*, towards meal start, *as possible*, which will lower bg peaks and ease the job for dura_ISF).



To limit the danger of going low, it can make sense to design an Automation which pauses the delivery of more insulin.

This one was suggested by Alex999

If a glucose plateau built under 140 mg/dl, do not treat via dura_ISF (because the defined Action is to set an elevated TT to a level that will not require more correction insulin.

An alternative Action would be to set, near the actual glucose target, an odd-numbered TT (which blocks any SMB be given, while valid).

4.5.2 bg_ISF

Since in Full Closed Loop we make our loop give us the maximum SMB size it 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 Closed Loop we do not make much use of the ***bg_ISF*** component of autoISF.

- Wanting to get most of our insulin from SMBs delivered at fairly low (but beginning-to-rise) bg implies that we do **not** make ISF weaker at low bg. Under preferences/OpenAPS SMB/autoISF/bg_ISF settings you could set **lower ISF_range_weight** = 0.0.

  If you want to analyze in your data, whether you might benefit from a milder ISF at low bg values (e.g. if you often go below target after correction of only mildly elevated bg in the preceding hours), you may want to try lower ISF_range_weight = 0.1 or 0.2. Study the effects from bgISF, and increase, or decrease, the bgISF_weight to fine tune the sought-after affect.

- The **higher_ISF_range_weight** is used when bg is above target, It then strengthens ISF the more the higher the set weight is. 0 disables this contribution, i.e. ISF is constant in the whole range above target.

  In FCL, this factor should be fairly irrelevant: Near glucose peak, zero-temping usually prevails anyway, so the settings we try might often not be used really by the loop. Very likely, you can live with setting the weight to = 0.0 here, too.

  If you want to analyze in your data, whether you might benefit from a stronger ISF at high bg values (e.g. if you often remain above target after correction of elevated bg in the preceding hours), you may want to try higher ISF_range_weight = 0.1 or 0.2. Study the effects from bg_ISF, and increase, or decrease, the higher_ISF_range_weight to fine tune the sought-after affect.

Caution: Investigating effects of set ISF_weights is not really possible in periods of zero-temping. Too aggressive settings might not come into play most of the time.
However, some *other* time they might come into play, and *then* produce a hypo 1-2 hours later.
Therefore, **carefully study the SMB tab** (or better yet, do an emulator based analysis, see sections 10-11) to see

- what the selected weights **would** do, **if** there was **no zero-temping** at the time, and

- whether you bump into a set limitation already (if your bgAccel_ISF_weight makes you exceed allowed max. SMB size, then further tuning your settings only makes sense with either allowing bigger SMBs, or limiting bgAccel_ISF_weight to a lower number at whicjh you will not frequently bounce into the SMB limit)

- at which **other** times (rather than the one you currently look at and try to improve) that selected setting might backfire

Very important: Also try a **completely different meal** (within your common spectrum), to see how your settings work *there*.

- Iterate between 2 or 3 such meals to find one set of settings that works *good-enough for all.* That should be possible.

- If you can't make it work for certain meal types, see sections 4.7 and 5. what you can do then.

4.5.3 How your "UAM" concludes insulin need for your un-declared carbs

The UAM Full Closed Loop doesn't get any information from you as to how many grams of carbs will be there, for absorption. Not knowing when your steady-state max carb absorption phase…

- the earlier mentioned 30g/h, or
- with gastroparesis, or if on GLP-1 drug treatment, probably on a lower g/h level
- sometimes prolonged ("faked") by a brief episode of insulin resistance to fats

…might 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**s (=calculation of how much got absorbed in the *past* 5 minute segments), and phase out further *expected* 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

- *or* chapter 1.2 in "IC (carb ratio)…pdf" at: https://github.com/bernie4375/HCL-Meal-Mgt.-ISF-and-IC-settings

- *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).

4.5.4 Conclusion on managing bg highs

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.

The author is sceptical about using the **bg_ISF** in Full Closed Loop:

- In FCL you probably can afford to shut it entirely off via setting both related _weights to 0.0.
- At least be careful, use small ISF_range_weights and check whether you are happy with the contributions to effectively used ISFs
    - *Off topic:* If, coming from dynamicISF usage, you stay in Hybrid Closed Loop, but now with autoISF, you probably can use the bg_ISF parameter with higher _weights to emulate what you like to replicate from your dynamicISF experience.

bg highs will take time to resolve.
Interestingly, an after-dinner walk can work wonders sometimes (take glucose tablets along).

4.6 Tuning your initial settings

Be pro-active: **The earlier large SMBs come** (driven by bgAccel_ISF and pp_ISF) …

Note: Also your CGM smoothing may play a role here, that you may want to look into !

…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 FCL tuning effort into determining suitable weights for bgAccel_ and for pp_ISF, and for finding a suitable iobTH_percent**.
Low carbers probably should pay more attention on **dura_ISF**, besides seeing to it that bgAccel_ISF is not too aggressive (see case study 4.2).

Later, your FCL cockpit will give you access to **temporarily modulate** these essential parameters (see section 5.2.), providing you an opportunity

- in your tuning phase, for more research on the fly, so to speak
- everyday, for temp. adaptations to altered insulin sensitivity, or to special disturbances (if you occasionally see a need).

After you tuned your **initial settings** well, there should rarely arise a need for “fine tuning” later, see section 8 and case study 8.2!

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 come to the conclusion that **differentiated settings** (for different meals or meal time clusters) would be easier to establish, and/or work better for you, the following sections suggest many options you could try and use.

4.7 Covering more complex scenarios

You now can move on, to accommodate more complex scenarios.

- Depending
  - how satisfied you are with your initially reached result, or which **more extreme meals** (smaller? faster/slower carbs? totally different fat/protein content?) you would like your FCL to manage as well, or
  - whether you seek **temporary** adjustments that **make your FCL act more aggressive, or softer**

  you have a variety of options to deal with that, and this will be the topic in section 5.

- It is suggested to do **major exercise** still *in your hybrid closed loop* setting, *until* you have your FCL up and running for meals on normal days with no or only moderate exercise. Later, implement extras as discussed in section 6 to fully implement your FCL.

To deal with ***different*** **disturbances than presented by the meal spectrum you were calibrating for**, there will be **temporary modulations** of your FCL possible**.**

- Manual, making use of the top 3 buttons (%profile, exercise, TT; TT;section 5.2.2.2) *or*
- Semi-automatic (user triggered), aided by Automations you would set up, with a user defined extra button in your cockpit for it (section 5.2.2.3) *or*
- fully automatic (via pre-defined settings and/or Automations that e.g. that use different iobTH and/or different bgAccel_ISF-weights for different rough meal-time slots in your days: section 5.1.4)
- In future autoISF versions we could also pre-program 4 different clusters in /preferences, and call them up within a second from the TT button in the AAPS home screen (*only after implementation of an improved cockpit,* see section 5.3.3.1 (4) and section 6.4.3)

So, while FCL is about fully automatic cruising, your **AAPS main screen** will serve you as your cockpit to check how everything is running, and to aid your loop manouvering through some special disturbances.

In the **SMB tab** you can see how the autoISF modulation of ISF is overall applied to arrive at the actually used **effective ISF ("sens")**: See also example given in section 5.4.5

- In the SMB tab, *above the "start autoISF.." line*, the profile ISF is given ("ISF unchanged"), eventually with adaptation by activity monitor ("adjusting …ISF from … to ..) or by a TT ("adjusting …ISF from … to ..") or by a %temp. profile set (still called "ISF unchanged" then, meaning unchanged yet by autoISF).
- *Then* follows the autoISF section, explaining in detail how the recently encountered bg curve characteristics suggest adaptations, and what overall the conclusion is ("final ISF factor", calculated following the flowcharts as explained in detail in section 03.).
- *Below the autoISF section*, the effective ISF (sens) results from dividing the (unchanged or adapted) ISF *prior to* "start autoISF", with the determined "final ISF factor" at the end of the autoISF section of the SMB tab.

4.8 Profile helper

xls based tool is still under development / needs more user data / chapter will follow later