Insulin Sensitivity Factor ISF

Contribution to the discussion among DIY loopers

The author assumes no liability B/ V.3.2 30Oct22

1. What is ISF
2. Rough estimates for your ISF
3. Refined estimate via dynamic ISF (277,700 rule, Chris Wilson)
4. Determination of … ISF
5. Circadian ISF
6. Signs for a wrong ISF when looping
7. Tuning ISF in closed loop
8. Avoiding hypos
9. Short-time boosts via Automation
10. auto-ISF
11. Differential diagnosis occlusion
12. Remarks regarding (ISF and) Full Closed Looping

1. What is a ISF?

Your **insulin sensitivity factor** (unit: (mg/dl)/U or (mmol/dl)/U) says how sensitive you react to insulin, so how much your glucose level will come down per unit of insulin .

For instance, ISF = 30 (mg/dl)/U means, per unit of insulin our glucose level goes down 30 mg/dl. For mmol, divide by 18 g/mol: ISF = 30/18 = 1.7 (mmol/dl)/U

In Hybrid Closed Loop, the other profile factor, the carb ratio IC, plays the main role around meal times. However, two to three hours after any meal, the administered meal bolus rapidly loses power, but now proteins, fat, and carbs exceeding what could be digested in the first 2 hours come to absorption and require more insulin. This is where your ISF plays the (or a) major role (in iOSLoop, the IC factor continues to play a very strong role).

Unfortunately, the ISF is not one fixed number you or your loop can count on (count with):

* can vary between times of day => use **circadian pattern**

> => determine it for several times during the day and night, and check whether you see a typical „circadian“ sensitivity pattern corresponding to your bio-rhythm..(As a <u>high</u> basal rate and a <u>low</u> ISF factor are signs of <u>lower</u> insulin sensitivity (=of elevated insulin need), the pattern must be like a mirror-image. More see in section Circadian Pattern, below)
>
> See also: AndroidAPS Users / Files / ..circadian … Duesterhoff.xls: https://www.facebook.com/groups/AndroidAPSUsers/permalink/2869638923257506/

* will vary between days e.g. when hormones play into it => **Autosense**

***** can also fluctuate within shorter episodes, e.g. @ stress, or @ high glucose values because of **insulin resistance** after a fatty meal

***** varies also with **TDD** (although, to me at least, it is unclear whether this is an independent effect, or just owed to the fact that on days with high TDD typically more fatty meals are consumed, accompanied by increased insulin resistance

=> Short time boosts may be required, see later chapter 3, further down

=> Based on statistical findings, Chris Wilson recently proposed a dynamic ISF formula that roughly considers most of these effects (see chapter further down)

2. Rough estimates for your ISF

a) … using Autotune (not recommended by the author)

Autotune gives you ONE „average" ISF. Reliability of this result is seen controversially in the looper community, largely, because Autotune relies on your data inputs which probably contain errors (like your carb inputs).

b) … via daily total insulin dose, TDD

The daily total insulin need describes, roughly, the insulin sensitivity of a type1 diabetic. The following table reflects values seen in a big study.

Example: At 35 – 40 units of daily insulin (mid of left column) an ISF of around 45 mg/dl/U (35-60) would be suspected, or around 2.6 mmol/dl/U

Estimating the Sensitivity Factor based on Total Daily Insulin

Average Total Daily Insulin (all basal + all boluses)	Sensitivity Factor (mg/dl) how much 1 unit lowers blood sugar	Sensitivity Factor (mmol) how much 1 unit lowers blood sugar
5 units	320-360	18-20
7 units	220-260	12-14
10 units	155-185	8.6-10.3
12 units	125-155	6.9-8.6
15 units	95-125	5.3-6.9
18 units	80-110	4.4-6.1
20 units	70-100	3.9-5.5
25 units	60-80	3.3-4.4
30 units	50-70	2.8-3.9
35 units	40-60	2.2-3.3
40 units	35-50	2.0-2.8
45 units	30-45	1.7-2.5
50 units	30-40	1.7-2.2
60 units	25-35	1.4-2.0
70 units	20-30	1.1-1.7
80 units	20-25	1.1-1.4
100 units	15-20	0.8-1.1
120 units	13-17	0.7-1.0
140 units	11-15	0.6-0.8
160 units	10-12	0.5-0.7
180 units	9-11	0.5-0.6
200 units	8-10	0.4-0.6

c) Estimate using the **„1700“ rule**

According to a study *https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4454102/* the ISF can be estimated using:

$$ISF = 1700 \,/\, TDD$$

where *TDD* is the total daily insulin, determined for each virtual patient, and assuming an average diet of 180 g of carbs for adolescents and adults, and 135 g for children.

Example as above, 35…40 U TDD: ISF = 49 … 42 mg/dl/U (or divided by 18 for mmol)

3. Refined estimate for your ISFs via 277,700 rule (dynamic ISF, Chris Wilson)

According to https://www.facebook.com/groups/TheLoopedGroup/posts/2996545667228746/ and https://youtu.be/oL49FhOts3c (Chris Wilson/ see also in Loop&LearnFB and FreeAPS-X Telegram) the ISFs can be dynamically modelled using the formula:

$$ISF = 277{,}700 \,/\, (\mathrm{TDD} * \mathrm{bg})$$

Plug-ins in DEV branches are available for both, AndroidAPS (by Tim Street), and also of Free-APS-X, so there is some testing being done currently.

Discussion: I see the results from the formula for myself too high, so that would make more cautious corrections, than what I really use. But people who have put less effort into optimizing their ISFs, who have bg values over 200 often, and maybe also a diet requiring higher TDD than I, very likely would benefit.

As the formula is statistically derived from real-life data (sort of „top-down“), it should be useful for refining how any ISF you are working with (notably if you work with only 1 number) should be tweaked to reflect your true needs for correction-insulin better.

We can assume that this provides an easier-to-implement alternative to what below in this paper is suggested as kind of a bottom-up personalized approach to good ISF settings via:

- Circadian pattern of ISFs
- Boosting ISFs at high glucose values, notably if owed to post-meal insulin resistance
- Use of Autosense
- Use of refined autoISF.

If done right, the bottom-up approach based on your personal data should be better, and also more real-time reactive to changing insulin requirements - notably in full closed looping, but also in situations like this:

For instance, with the 27… formula you could use the prior day's TDD or the prior week's average TDD. The problem there is, that in real life, you are probably (hopefully!) jumping day-to-day between TDD (because a diet sin with high TDD on one day is followed by a day involving some fasting, i.e. low TDD, to make up for it). That is just one example why I prefer an ISF modulation that is tagged to real-time data. (The TDD is not accessible real-time; I guess algo developers could enable entering a "intended today's TDD“ (or % reduction from yesterday’s „pigging-out" ☺) somewhere.

Other further refinements, and regarding some research regarding dynamic ISF, see . https://youtu.be/Ug8LgJPTyOI (Tim Street) , debates on dynamic ISF in FreeAPS-X telegram group, and below in chapters on boosting, and on autoISF. Especially autoISF (see chapter further down) allows nearly „surgical“ intervention to tweak ISF depending on a variety of real-time conditions. But this is still in development (and will be complicated to understand and probably also to implement (=to tune the additional settings coming with it)).

Please apply caution if you *combine* some of the discussed methods, as they probably work in the same direction, and *could „over-adjust“* your ISFs.

4. Determination of Carb Rise Ratio CRR and Insulin-Sensitivity Factor ISF

Verify that you have a correct **basal rate** before determining factors!

> Else your ISF will have to balance errors in your hourly basal pattern, and you may lose some of the
>
> Helpful links regarding basal rate, here is a good article in support of that process: https://www.mysugr.com/en/blog/basal-rate-testing/ , and a process description including a tool : https://www.mevita.de/.../online-zugang-erstellung-einer.../
>
> *Although I advise to determine your true, likely **circadian**, basal rate: It can be „good enough“, and advantageous especially for small kids, to simplify things and assume a (in case of doubt, low) **flat basal rate**, and a well-contoured ISF pattern („with enough bite where it counts“). See discussion here: https://www.facebook.com/groups/1900195340201874/user/100000686997322/?__cft__[0]=AZX3WqWLR9vrra1dZbGeP_sglvA41eVlZg2R0l98iDk_djnAtJfk4jN2nPiQyxIMM_NgbZayhhglfdCW7FGSjFgV54KPJ_0qBGVekw60g4ahse6_izL9OX4HXkhGkCyWODQ&__tn__=-UC%2CP-R*

To determine ISF, select a day without prior strenuous exercise/activity, without exceptional stress, or infection.

As testing for ISF must start out with a high plateau of glucose level, it makes a lot of sense to do a determination of your **CarbRiseRatio CRR** first:

(In a relatively stable, normo-glycemic phase) take a sweet drink with a known carb content, and watch (in Open Loop, not giving any insulin except profile basal rate) by how many (mg/dl) glucose rises, until reaching a plateau (in about 1 hour).

Example: After consuming 20g, my glucose rises from 90 to 190 mg/dl.

CRR = (190-90)mg/dl / 20g = 5 mg/dl / g

Now, having reached a high plateau – *at 190 mg/dl in our example* ,-we move on to determine the **ISF**.

Inject (still in Open Loop, with profile basal running), an amount of insulin as you typically would (considering your personal insulin sensitivity). *Example: 2 units.*

Then watch for about three hours until a deeper plateau is reached, *for example at 110 mg/dl.* Then you can evaluate:

Example: 2 U of insulin lowered my glucose by 80 mg/dl . Hence, ISF = 40 mg/dl / U

Off topic remark: Using CRR and ISF you can also calculate your IC for this time of day, or *check the* ***plausibility of the IC*** you are already using, because:

ISF (mg/dl decrease per unit of insulin) / **CRR** (mg/dl increase per g of carb) = **IC** (g carb per unit of insulin) (similar in mmol) or also: CRR = ISF / IC

IC in the example from above: ISF / CRR = 40 (mg/dl)/U / 5 (mg/dl)/g = 8 g/U = IC

See also:: http://seemycgm.com/2017/10/29/fine-tuning-settings/

The CRR is relatively stable (does not vary by much over 24 hours):
- while both ISF and IC do vary (in similar ways, related to your insulin sensitivity)
- if you divide one by the other, deviations owed to sensitivity cancel out.

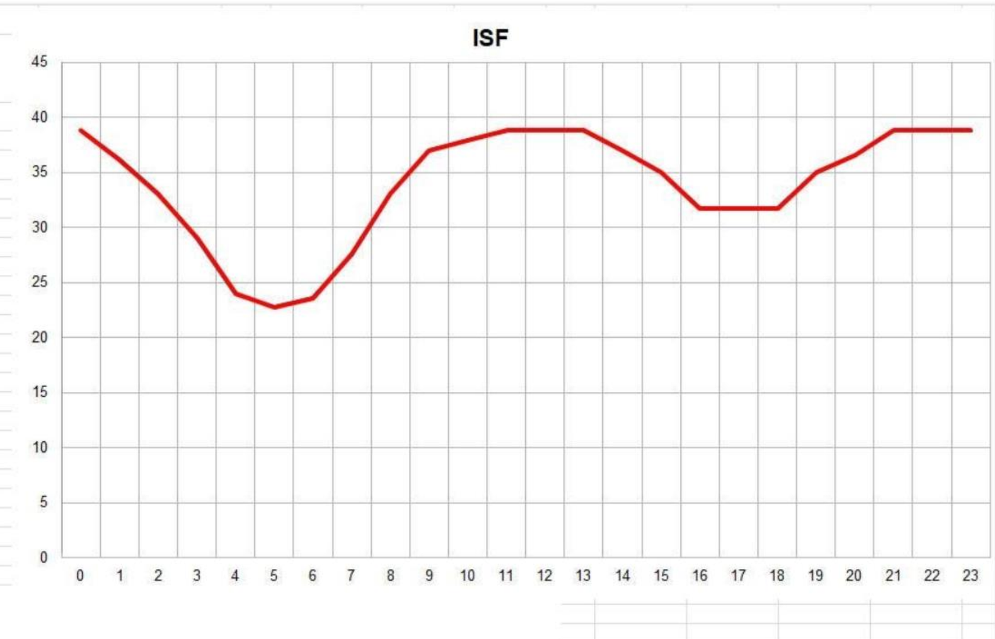
5. Circadian pattern of ISF

ISF varies over 24 hours according to a „circadian“ pattern of varying sensitivity to insulin.

See AndroidAPS Users / Files / ..circadian … Duesterhoff.xls: https://www.facebook.com/groups/AndroidAPSUsers/permalink/2869638923257506/

Example

ISF	
ISF HM 12-99 Jahre	
100%	
Zeit	**ISF**
00:00	39
01:00	36
02:00	33
03:00	29
04:00	24
05:00	23
06:00	24
07:00	28
08:00	33
09:00	37
10:00	38
11:00	39
12:00	39
13:00	39
14:00	37
15:00	35
16:00	32
17:00	32
18:00	32
19:00	35
20:00	37
21:00	39
22:00	39
23:00	39



Do not just copy a pattern that may be valid on average for T1Ds out there, or for a single other person! For instance, I am more of a night owl, so that curve shifts 1-2 hours to the right.Likewise, if you do shift work or cross-time-zone travel, keep an eye on your (probably time-delayed) adjustments of your body‘s insulin sensitivity.

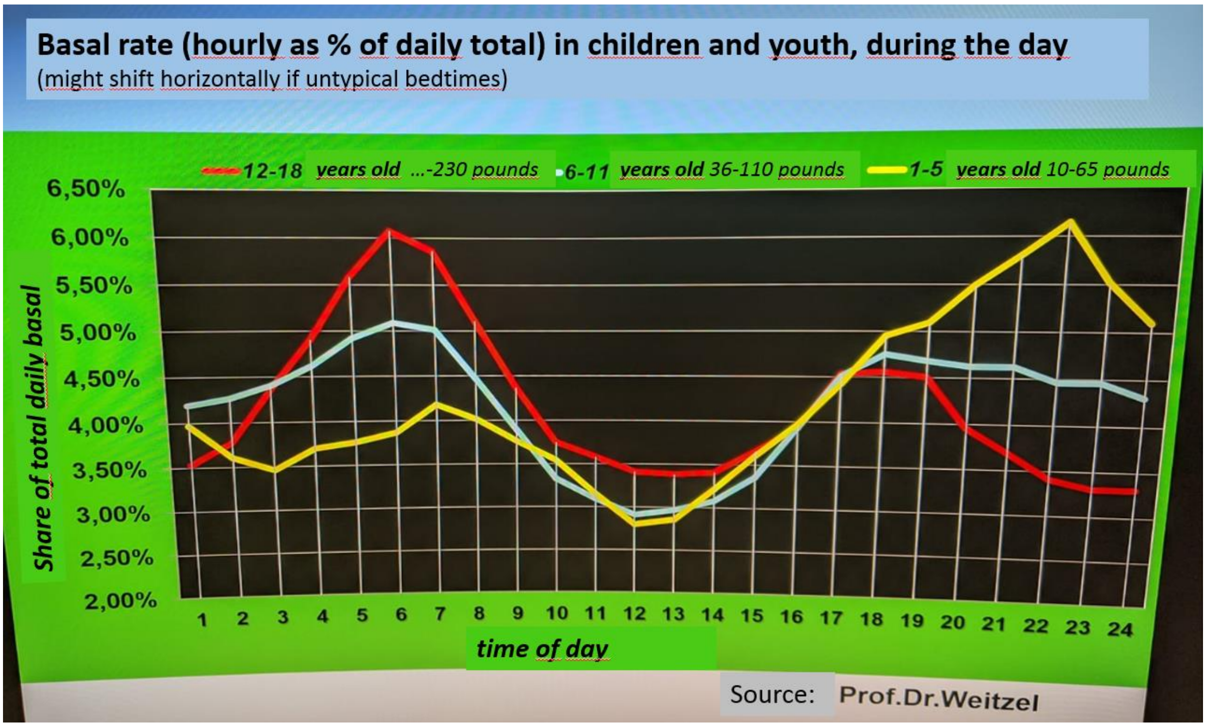
The deepest dip in ISF, and highest hill in hourly basal, is typically an hour or twao before getting up and breakfast. However, I usually skip breakfast, so I „break my fasting“ (!) for lunch and see higher insulin sensitivity there than the standard curve suggests. (Another explanation could be, that others would have tail of insulin sensitivity from breakfast still at work supporting insulin for lunch (with a typical breakfast, low in fat and protein).

> *In case you work with a flat basal rate*, your ISF should look even more „contoured“: For instance, if you use your low hourly basal, as approriate in most night hours, for all 24 hours, the daytime ISFs would need to be make up for „deficiencies“ in your hourly basal; likely morning and evening ISFs would dip relatively lower even than in picture on preceding page. YDMV!

So, determine **your** ISF for a couple of times, in which you find high values which your loop needs to regulate down (one of them, inconveniently, probably being after midnight; see also 2 chapters lower).

You can interpolate for the hours for which you did *not* „experimentally determine“ your ISF. Follow the idea (or data) you have regarding your general pattern of insulin sensitivity.

Caution regarding kids: For kids, the 24 hour sensitivity patterns can differ strongly from adults and can also change strongly in certain growth phases, see the following graph (shared by „basal rate guru“ Heiko Mueller, www.mevita.com) . The graph is about basal rates, and normalized to 100% being the total daily basal.(So, the graph below says how much %of the daily total is needed in every hour of the day. It does NOT show the very significant differences in basal insulin need between the age groups! We are looking only on the 24 hr patterns here!)



The graph is for basal pattern.

For ISF the curve would look „opposite“ (~mirrored on x-axis), because **higher** basal („mountain“) = lower insulin sensitivity = higher insulin need = **lower** ISF factor („valley“). *For the red curve (age 12-18 resembles adults fairly well already) we see this mirrored pattern beautifully in the ISF graph, 2 pictures higher up.*

6. Signs for a wrong ISF when looping

If your ISF is too strong (in other words the actual number is too low compared to reality of your real insulin sensitivity), one of the most common symptoms you'll see in closed-looping is a roller coaster of glucose values, and the temporary basals might be cycling between zero and high temping.

If ISF is too weak, the glucose is not corrected down to target (or this happens only extremely slow).

If you observe such behaviour in certain time periods of your day, you might consider changing the ISF by plus 10-20% (if ISF seemed too strong) or by minus 10-20% (if ISF seemed too weak), and see (with alarms set on loud), whether the situation improves next day.

7. Tuning of ISF in Closed Loop

The effects of a suboptimal ISF settings as discussed in the preceding chapter can be seen nicely in the night following a bigger dinner. When the meal time bolus loses power, about 2-3 hours after starting the meal, often relatively high glucose values exist at bedtime for the loop to take care of.

With a good ISF setting, the glucose curve will, within a few hours (while you are sleeping), go down and „hug“ the glucose target you have set for the night time.

If the glucose curve swings very much below target, or even triggers a hypo alarm, ISF is probably too strong, and you should elevate the nightime values in your profile and observe again in the following nights.

Conversely, if the glucose hovers for hours above target: In that case you would try with lower ISF next nights. Make changes in small steps (hypo danger) . Don't do this after a day with strong activity/exercise. Setting the nighttime glucose target a bit higher for the time of testing can also be a good idea.

8. Avoiding hypos after correcting high glucose values

The course of insulin activity during the 5 hour+ „DIA“ period of any administered bolus implies that anything you try to achieve for short-term glucose control (1-2 hour time frame), comes with a „burden“ of effects that tend to drive your glucose below target a bit later

A high glucose value therefore can not be corrected rapidly without running a risk to go into a hypoglycemia 2 or more hours later (as probably everybody has experienced with impatient „rage“ bolusses).

All loops apply caution in that they analyze the situation every 5 minutes, and generally apply much less than the calculated InsulinRequired (AndroidAPS generally applies 50% of what the calculation suggests).
(To avoid misunderstanding: This does not mean that 50% of insulin is really withheld ; what was not given in the preceding 5 minute segment, is likely missing „on top“ also in the following 5 minutes, and gets delivered to, again, 50% then , now making up already 75% of the previous InsulinRequ. …).

It is important to watch your loop with some patience, and analyze what (if anything) you might try differently for the next day..

In case your loop is acting too aggressively, you may need to take a couple of grams of carbs to avoid going low. (If you use AndroidAPS, your main screen probably shows carbs required in next minutes, to boost the cob number which is given right next to it), Note down how many grams you took *for example,. 4 grams* . Using your IC factor, *e.g.. 8 g/U*, calculate how much insulin (*0,5 U*) would be needed for absorbing these extra grams. Your post-prandial SMBs should have given that amount of insulin **less** (assuming you observe that things go well with these extra 4 grams) For example, if your pp. SMBs were *4 U* in total, *you should try 3.5 U next time..Your ISF factors in the relevant hours should be elevated by 14% (=4 / 3.5).*

To summarize:

1. Try to avoid high glucose values (see material on meal management in closed loop)
2. If high values occur: First exclude occlusion as underlying problem. (see chapter 11).
3. Watch and analyze your loop preferably without giving extra boli (but take, and record, anti-hypo snacks if needed), Define what you might change to improve in the next days.
4. In case you think you see very short-lasted resistance, e.g. from fatty meals, consider brief boost of ISF (next section).

9. **Short-time „boost“ of ISF** (in AndroidAPS also via **Automation**)

Background: Often a short phase of insulin resistance is encountered a few hours after a fatty meal. Insulin seems to have lost it‘s potency for 1 or 2 hours. This is often assigned to receptors blocked by free fatty acids, and requires extra insulin – but only for a very limited time, until the receptors are cleared -.

Rather than taking care of this phenomenon by giving an extra manual bolus, AndroidAPS allows an elegant automated solution for this problem: When conditions like glucose above 180 mg/dl (and others, like iob thresholds, time of day where the problem is likely etc) are fulfilled, a more aggressive ISF can be selected *(for example, by setting 130% profile (‚override‘) for 8 minutes.*

Such automations are technically very easy to „program“, but require attention to one’s own daily patterns, to define a truely personalized problem solution.

Details see for example at page 23 in: Meal-Management II
https://www.facebook.com/groups/AndroidAPSUsers/permalink/2864720013749397/

10. **Auto-ISF**

Auto-ISF is a feature (currently, Feb.2022, not yet included in AndroidAPS Master), which allows flexible boosting of ISF. (So it is not a program for automatic determination of your ISF values. It uses your ISF, to adjust it in certain situations).

The working principle of Auto-ISF is, to provide sharpened ISF factors (and there are various new parameters in /Settings to tune this individually), depending on;

1) Glucose plateau = how high glucose is, and for how long it is that high already.
Two new parameters in /Settings allow to make more or less aggressive (or no) use of this added feature. This was mainly developed to take care of fat-related stubborn highs. More see p. 33 in Meal-Management III.
https://www.facebook.com/groups/AndroidAPSUsers/permalink/2932619803626084/
In Full Closed Loop (UAM no bolus) this feature can be valueable to improve control after heavy, multi-course meals.

2) Glucose level = allows to sharpen ISF at high glucose levels (I am not using this, because it is kind of late to react then; also, if my glucose runs really high, this is usually occlusion related = ISF does not help at all then, see chapter 11. But in Hybrid Closed Loop, if you often find yourself high when your given bolus expires, the feature might help/YDMV)

3) Glucose delta = sharpen ISF when glucose rises strongly. It makes a lot of sense to act BEFORE glucose is high, and you can tune aggressiveness according to personal needs.
4) Acceleration of glucose rise = enables strongest ISF as soon as a meal-related rise is detected (and reduced aggressiveness when curve bends around heading towards a peak). This is extremely helpful in full UAM looping, but requires a smoothly operating CGM, so what the sophisticated math makes of your glucose curve, to fire off SMBs, is not just a temporary artefact. (Also, in case the user still gives occasionally a bolus, which totally deforms any meal related curves the full UAM loop is expecting, this feature 4) should probably be switched off!).

Any of these 4 autoISF varieties can (must) be tuned separately, but also can interfere with eachother and with Autosense. Developments are partially open to see in Github (search terms: autoISF, determine basal emulator, ga-zelle; and in loopercommunity.org).

PS: As an adjunct to autoISF but also to the Automations, the **allowed max. size of SMBs** probably must be extended, so boosts can effectively be carried out. Therefore, another new parameter setting, SMB range extention, is needed by most loopers then. (It is included already in FreeAPS-X). .

11. „Differential-diagnosis“ occlusion when dealing with high glucose values

Back to easier stuff, relevant to us all.

When we encounter high glucose values it is very **important**, as a first step, **to exclude a compromised insulin delivery** as underlying cause.

Users of Lyumjev or of Fiasp seem to experience occlusions quite often, especially if not changing cannula frequently. Under the skin, near the insulin insertion site, partial or full blockages can occur and insulin is being pushed back out into the adhesive patch of the cannula. This results in glucose rising despite **(fake) high iob**.

In such instances, loop should be switched off. The share of insulin that did not effectively enter the body system must be estimated (conservatively, in case of doubt), and erased (in AAPS: Treatments tab/Bolus/erase a number of entries there), so a realistic iob number is there for the loop to continue .The occlusion must be resolved (usually by setting a new cannula) , and the high glucose must be treated. As your iob estimate is likely uncertain, do this conservatively, and continue in closed loop with alarms set on loud .

In case the high values were realized only after a couple of hours, watch out for signs of **Keto-Acidosis** and be ready to follow a treatment scheme for it. Hospitalization may be required.

Off Topic: Also **fake cob** can happen. This would develop **after throwing up** shortly after a meal which you bolussed for. As insulin cannot be removed from the body, and glucose snacks etc.- which are needed in significant quantity quickly now - , likely are also thrown up, a glucagon injection may be needed.

Caution: To stop more insulin delivery, and erase false cob, is by far not sufficient to resolve this crisis. This is an **emergency** with extremely high hypo danger at the rapidly approaching point of time, when your meal bolus reaches maximum activity! Call for emergency:
* if you can’t eat/drink (or not keep it in)
* if you go below a certain glucose value that you laid out for yourself when still clear in the head. (It probably would be too complicated to build a strategy on the 5-minute delta glucose values, and amount of carbs you were successfully taking)

Especially users of Lyumjev or Fiasp should be aware how fast glucose can sink in absence of sufficient carbs for the iob that is present.!

12. Remarks regarding „UAM“ Full Closed Loop

With well-functioning CGM and a rapid insulin (preferably Lyumjev), more and more – currently mostly adult – loopers establish a reasonably functioning Full Closed Loop with AndroidAPS (also without having a low carb diet).

Currently (Febp.2022) this is only possible with good results when using the oref(1) algorithm wSMB+UAM feature, as offered by OpenAPS and AndroidAPS. iOS Loop, in contrast, requires fairly exact carb inputs. However, for i-Phone users there is a new looping option „FreeAPS-X“ in late stage development that also incorporates oref(1).

In Full Closed Loop (UAM/no bolus/no carb inputs) **ISF is the key parameter because** no more meal boli are given, and **the loop exclusively reacts on rising glucose**.

Full Closed Looping had been pioneered by some low carb OpenAPS Loopers, but picked up momentum when in 2020 Lyumjev became available in an increasing number of markets.

-

There are currently several avenues established or explored towards a no-bolus no carb-inputs Full UAM Closed Loop,

1) Full UAM loop using Master AndroidAPS
   It has been demonstrated that using AndroidAPS Master software, a reasonably functioning full closed loop can be possible:

   Lyumjev – a fully closed loop case study with oref1 (from this author)

   https://bionicwookiee.com/2022/02/09/one-year-of-no-bolus-looping-data-analysis/ (David Burren)
   Discussion: This is the highly recommended starting point for anyone interested. Well-determined (and relatively stable?) personal factors, and a precise insulin model seem to be necessary. Refined methods (2) & 3)) should be tried if well-described problem areas are evident which additional features might help solve.
   Warning: Adding several new (and hard-to-understand) parameters easily leads to a instable and hard-to-recalibrate system!

2) Automations to boost ISF at critical moments Reference: (Bernie & others, de.loopercommunity.org „Loopen ohne Bolus“; https://www.facebook.com/groups/AndroidAPSUsers/permalink/2932619803626084/)

   After correctly identifying a rising glucose as sign of a meal, microboli (SMBs) must be automatically launched that quickly raise iob in the first hour following any meal to a level comparable to what was seen in HybridClosedLoop. To enable the loop to do this, over the course of several weeks **the personal patterns** around meals must be analyzed, and step by step suitable Automations must be defined that can sharpen ISF factors for several minutes at a time. .

   Profile-ISF values are the ***baseline*** **upon which these boosts build**, which try to make up for the minutes lost compared to the bolus given before/at each meal in Hybrid Closed Loop or also in conventional therapy.
   The longer the DIA of the insulin in use, the more difficult to avoid a hypo several hours later. (Advantage for the shorter DIA with Lyumjev compared to Fiasp!)

   Discussion: Beauty of this method is that everybody can define things based on Master AndroidAPS software (maybe with minor modification re. allowed SMB size or %profile or %corr.factor>50%).This implies that all users are intimately aware what they defined there, for which situations etc. and can within seconds, on their smartphone, also modify any of their Automations if they see that indicated

   Patching over a identified problem with an Automation is easy. But to define Automations that are working well for a range of situations as possible in the future, is a complex task, requiring careful analysis of one’s personal patterns.
   The multiple %profile switches around every meal make for „jumpy“ adjustments.

3) Boost https://www.diabettech.com/artificial-pancreas/the-insulin-only-full-closed-loop-an-n1-experiment/ (Tim Street)
   Uses TDD plus additional bolusing decision adjustments. 1. A "Boost" function that can deliver the equivalent of the current hour's basal if it looks as there are carbs causing a rise. The amount you can "boost" can be scaled by the user. 2. An "Enhanced" openapsSMB model, where the max bolus that can be delivered is set by the user directly, rather than the number of minutes of basal.

4) AIMI 15.1 for full closed loop
   https://www.facebook.com/groups/1900195340201874/user/612815080 (Matthieu Tellier/ MTR...../Gitlab)
   Anectodal reports pop up in AndroidAPS Users FB that show quite impressive results. There is virulent developement (including branches that suggest to include pre-bolus or carb inputs):
   Discussion: - (The author is not familiar with it).

5) Auto-ISF 2.2 for full closed loop (Early dev, info in Github, and in de.loopercommunity.org)
   With (the most advanced version of) autoISF it is possible to tune the AndroidAPS loop for optimized response to typical meal-related glucose curves.
   Finding an initial setting for the many new parameters is not easy (probably best done with the determine basal emulator), and most parameters are also difficult to understand in the sense to imagine (or even calculate) effects of changing parameter settings.
   But when adjusted, this provides a smooth full closed loop experience.
   Discussion: Given the multitude of additional parameters, it is currently easy to get lost, and very hard to un-tangle then to tune for an improved solution.

6) Pilot implementation of a full closed loop in **FreeAPS-X**
   (Very few first attempts, nothing substantial to report yet. FreeAPS-X-ers are in contact with Boost and with AndroidAPS autoISF developers, so eventually you will see interesting features on both platforms.)

There are also proposals to retain features from Hybrid Closed Looping, notably making pre-boli, or also making carb inputs:

7) Using a pre-bolus before meals

Approaches that involve occasional user bolus help solve the initial meal rise problem, notably for consumers of rapid carbs (see e.g. UAM mit pre-bolus/de.loopercommunity.org)

Discussion: This comes at the cost of running into complex issues relating to the glucose management „hand-over" to the loop
Makes life easier by allowing sloppiness regarding pre-bolus timing and carb entries.
The method is similar to hybrid closed looping as AndroidAPS users know it.

8) Making detailed carb inputs for each meal *(but no bolus)* A few people are going that route which gives the loop optimal info, but requires careful everyday attention to carb inputs, very much like in iOS Loop.

Discussion: It does not really qualify for „Unannounced Meals“ but relies on the loop giving insulin, and might have very good performance. So far, not much was published.