Error case 1 – Negative temp basal is rounded to 0, but return reason says setting – Infinity/hr.

Explanation:

Bugged, but only mildly. setTempBasal() function has a proper fault tolerance check.

oref0/lib/basal-set-temp.js lines 24-28

    if (rate < 0) {

        rate = 0;

    } else if (rate > maxSafeBasal) {

        rate = maxSafeBasal;

    }

However, the reason field returned in this case is misleading; it says, “setting temp basal to -Infinity U/hr”, but the returned rate is actually 0 (correctly). Worth mentioning, but not a safety issue.

Consequences:

User confusion, but shouldn’t be a safety hazard.

expoSE replay '/home/dylan/oref0/test.js' '{"gs\_delta":0,"gs\_glucose":39,"gs\_long\_avgdelta":0,"gs\_short\_avgdelta":0,"gs\_noise":0,"ct\_duration":0,"ct\_rate":0,"ct\_temp":"","iob\_iob":0,"iob\_activity":0,"iob\_bolussnooze":0,"autosens\_ratio":-1,"profile\_max\_iob":0,"profile\_dia":0,"profile\_type":"","profile\_current\_basal":-10,"profile\_max\_daily\_basal":0,"profile\_max\_basal":0,"profile\_max\_bg":0,"profile\_min\_bg":0,"profile\_sens":0,"profile\_carb\_ratio":0,"md\_carbs":0,"md\_nsCarbs":0,"md\_bwCarbs":0,"md\_journalCarbs":0,"md\_mealCOB":0,"md\_currentDeviation":0,"md\_maxDeviation":0,"md\_minDeviation":0,"md\_slopeFromMaxDeviation":0,"md\_slopeFromMinDeviation":0,"md\_allDeviations":[],"md\_bwFound":false,"\_bound":13}'

Full output

SMB disabled (!microBolusAllowed)

profile.sens: 0 sens: -0 CSF: NaN

Carb Impact: 0 mg/dL per 5m; CI Duration: 0 hours; remaining CI (~2h peak): NaN mg/dL per 5m

Problem with iobArray. Optional feature Advanced Meal Assist disabled

UAM Impact: 0 mg/dL per 5m; UAM Duration: 0 hours

minPredBG: 999 minIOBPredBG: 999 minZTGuardBG: 999 avgPredBG: 999 COB: 0 / 0

BG projected to remain above 0 for 240 minutes

naive\_eventualBG: 39 bgUndershoot: -19 zeroTempDuration: 240 zeroTempEffect: 0 carbsReq: NaN

**Rate: 0**

Duration: 30

Temp: absolute

Deliver at: 2022-05-02T23:38:50.412Z

Error: undefined

Reason: COB: 0, Dev: 0, BGI: 0, ISF: 0, CR: 0, Target: 0, minPredBG 999, minGuardBG 999, IOBpredBG 39; Eventual BG 39 >= 0, 0m@0.00 > 2 \* insulinReq. Setting temp basal of -InfinityU/hr.

Error case 2 – Negative rate returned.

Explanation:

Bugged. bg stands for ‘blood glucose.’ In this test case, gs\_glucose:0 indicates that ExpoSE passed in 0 as the blood glucose value, which indicates a sensor error. This seems to be a composite bug.

So, the key parameters from the test case that trigger the fault are gs\_glucose=0 and profile\_current\_basal=-0.05. gs\_glucose=0 causes control to the enter the block if(bg <= 10 || …) which replaces the current temp basal with the user-provided fixed basal rate. However, the user-provided rate is invalid here (-0.05), but no check occurs for this case.

Solution:

profile\_current\_basal is passed to a round\_basal() function which converts a raw input to a valid pump input. It should *probably* go ahead and round negative inputs up to zero. That is already what setTempBasal() does, so code that goes through setTempBasal() is safe. If that was acceptable for setTempBasal(), it should be acceptable here too. Alternatively, determine-basal itself could perform the check. It is also worth noting that this control block has a commented out call to setTempBasal()… if control had proceeded through setTempBasal() instead, then the value *would* be correctly clamped to 0 as in the first case analyzed. Why *doesn’t* it go through setTempBasal when it used to?

Unclear. It’s from this commit. <https://github.com/openaps/oref0/commit/3a87b49de5d07c5f0ba6459e523f8d19c313975f>

Determine-basal, lines 125-126.

    var profile\_current\_basal = round\_basal(profile.current\_basal, profile);

    var basal = profile\_current\_basal;

Lines 139-173, determine-basal. Commented out call is relevant and highlighted.

 if (bg <= 10 || bg === 38 || noise >= 3) {  //Dexcom is in ??? mode or calibrating, or xDrip reports high noise

        rT.reason = "CGM is calibrating, in ??? state, or noise is high";

    }

    [Irrelevant code omitted]

    if (bg <= 10 || bg === 38 || noise >= 3 || minAgo > 12 || minAgo < -5 || ( bg > 60 && glucose\_status == 0 && glucose\_status.short\_avgdelta > -1 && glucose\_status.short\_avgdelta < 1 && glucose\_status.long\_avgdelta > -1 && glucose\_status.long\_avgdelta < 1 ) ) {

        if (currenttemp.rate > basal) { // high temp is running

            rT.reason += ". Replacing high temp basal of "+currenttemp.rate+" with neutral temp of "+basal;

            rT.deliverAt = deliverAt;

            rT.temp = 'absolute';

            rT.duration = 30;

            rT.rate = basal;

            return rT;

            //return tempBasalFunctions.setTempBasal(basal, 30, profile, rT, currenttemp);

        } else if ( currenttemp.rate === 0 && currenttemp.duration > 30 ) { //shorten long zero temps to 30m

            rT.reason += ". Shortening " + currenttemp.duration + "m long zero temp to 30m. ";

            rT.deliverAt = deliverAt;

            rT.temp = 'absolute';

            rT.duration = 30;

            rT.rate = 0;

            return rT;

            //return tempBasalFunctions.setTempBasal(0, 30, profile, rT, currenttemp);

        } else { //do nothing.

            rT.reason += ". Temp " + currenttemp.rate + " <= current basal " + basal + "U/hr; doing nothing. ";

            return rT;

        }

    }

Reference test:

Lines 450-456 determine-basal.test.js.

Rate should be undefined in this case (blood glucose < 30 indicates a sensor error).

    it('should let low-temp run when bg < 30 (Dexcom is in ???)', function () {

        var currenttemp = {"duration":30,"rate":0,"temp":"absolute"};

        var output = determine\_basal({glucose:10},currenttemp, iob\_data, profile, autosens, meal\_data, tempBasalFunctions);

        //console.log(output);

        (typeof output.rate).should.equal('undefined');

        output.reason.should.match(/CGM is calibrating/);

    });

expoSE replay '/home/dylan/oref0/test.js' '{"gs\_delta":0,"gs\_glucose":0,"gs\_long\_avgdelta":0,"gs\_short\_avgdelta":0,"gs\_noise":0,"ct\_duration":0,"ct\_rate":0.95,"ct\_temp":"","iob\_iob":0,"iob\_activity":0,"iob\_bolussnooze":0,"autosens\_ratio":0,"profile\_max\_iob":0,"profile\_dia":0,"profile\_type":"","profile\_current\_basal":-0.05,"profile\_max\_daily\_basal":0,"profile\_max\_basal":0,"profile\_max\_bg":0,"profile\_min\_bg":0,"profile\_sens":0,"profile\_carb\_ratio":0,"md\_carbs":0,"md\_nsCarbs":0,"md\_bwCarbs":0,"md\_journalCarbs":0,"md\_mealCOB":0,"md\_currentDeviation":0,"md\_maxDeviation":0,"md\_minDeviation":0,"md\_slopeFromMaxDeviation":0,"md\_slopeFromMinDeviation":0,"md\_allDeviations":[],"md\_bwFound":false,"\_bound":8}'

Full output

Rate: -0.05

Duration: 30

Temp: absolute

Deliver at: 2022-05-02T23:42:27.677Z

Error: undefined

Reason: CGM is calibrating, in ??? state, or noise is high. Replacing high temp basal of 0.95 with neutral temp of -0.05

Error case 3 – Unusually high rate returned

Explanation:

“High temp basal of 11” corresponds to ct\_temp parameter. OpenAPS interprets this as saying that the currently running temp basal is set to 11. Noise is 3, so the high-noise block is triggered. In this case, OpenAPS has a fallback behavior of replacing basal rate with the user preference profile\_current\_basal. Is this okay? OpenAPS doesn’t check this in determine-basal. Is a user desired default basal rate of 10 ever okay?

expoSE replay '/home/dylan/oref0/test.js' '{"gs\_delta":0,"gs\_glucose":61,"gs\_long\_avgdelta":0,"gs\_short\_avgdelta":0,"gs\_noise":3,"ct\_duration":0,"ct\_rate":11,"ct\_temp":"","iob\_iob":0,"iob\_activity":0,"iob\_bolussnooze":0,"autosens\_ratio":0,"profile\_max\_iob":0,"profile\_dia":0,"profile\_type":"","profile\_current\_basal":10,"profile\_max\_daily\_basal":0,"profile\_max\_basal":0,"profile\_max\_bg":0,"profile\_min\_bg":0,"profile\_sens":0,"profile\_carb\_ratio":0,"md\_carbs":0,"md\_nsCarbs":0,"md\_bwCarbs":0,"md\_journalCarbs":0,"md\_mealCOB":0,"md\_currentDeviation":0,"md\_maxDeviation":0,"md\_minDeviation":0,"md\_slopeFromMaxDeviation":0,"md\_slopeFromMinDeviation":0,"md\_allDeviations":[],"md\_bwFound":false,"\_bound":11}'

Full output:

Rate: 10

Duration: 30

Temp: absolute

Deliver at: 2022-05-03T00:10:24.971Z

Error: undefined

Reason: CGM is calibrating, in ??? state, or noise is high. Replacing high temp basal of 11 with neutral temp of 10