

Version 5.2 Update Notes for Piano-5 Users

• About Piano 5.2

Piano version 5.2 consolidates incremental improvements introduced since Piano 5.0. Some recent customers will find that many or most of these were already included in their copy, but installing this release is still recommended for all. If you are uncertain about how anything in this document might apply to your own particular setup, please do not hesitate to contact me directly as usual.

The new features and changes do not affect the interface. The 'look and feel' of Piano will be exactly the same.

The aircraft database has grown with a variety of new and revised models. All of your own existing aircraft and engine files can be loaded into v5.2 and will continue to yield the same results (other than for any minor differences explicitly stated in these notes).

• How to install

Before installing you should move or rename your existing '**data**' folder (and preferably also make a backup of everything), to ensure a clean installation for v5.2. The '**data**' folder is where all planes, engines and other files used by Piano are located. If you are unsure how to find it, do the following:

- Start your existing copy of Piano and select '**Load Plane**'.
- Look at the filepaths usually shown near the top of the dialog. The location of the '**data**' folder is one level above the '**planes**' folder and below the '**Piano**' folder. You can create a convenient shortcut on your desktop.
- Quit Piano, then find and move or rename your '**data**' folder manually.

You can now install v5.2 by double-clicking the **Piano** installer exec that you have received with this update note. Finally, move any of your own planes and engines from your old '**data**' folder to the new one as necessary.

The new features are as follows:

• Economy cruise at a Cost Index added to Flight Manoeuvres

If you are unfamiliar with Cost Indices and not interested in them, just skip this section!

Cost Index (CI) is the ratio of time-dependent costs to fuel costs. It is most commonly defined as (hundreds of \$ per hour) / (\$ per lb of fuel), and therefore has units of 100*lb/hr. Piano sticks with these units under all circumstances. This was a conscious decision to minimise confusion. Please be aware that some may define Cost Indices in kg/min. These should be multiplied by 1.32 to convert them to 100*lb/hr. There are however no 'natural'

units with CI, minutes being no more 'metric' than lbs. Costs and Cost Indices go beyond metric vs imperial choices.

The **Flight Manoeuvre** dialog now displays a new item labelled '**economy at cost index**' at the end of the '**Airspeed**' menu. This lets you input your chosen Cost Index. Piano can then run a variable-Mach Economy Cruise that will minimise cost at the given CI. The '**Thrust**' setting must be set to 'maintain level cruise' or 'maintain level flight'.

If instead of a single manoeuvre you generate an arbitrary **manoeuvre sequence** that does not necessarily maintain constant altitude but **does** include at least one level-flight segment specifying the Cost Index, you will see a total **cost** for the entire sequence at the very end of the output. You can therefore obtain costs for a complete flight, or any part thereof. If your flight profile includes multiple segments that use a CI, all your CI entries must be the same value otherwise no cost can be calculated.

The **cost** that you see is given in **normalised** form, based on 1 currency unit per lb of fuel. You can simply factor it for fuel price in any currency: At the time of writing, jet fuel is roughly 0.5 US\$ per lb, therefore multiply the normalised cost by 0.5 to give US\$.

Conceptually, tracking the CI is only meaningful in level cruise. However, suitable airspeed schedules for the climb and descent phases are often published by manufacturers. These are generated via the following (somewhat debatable) procedure:

To determine hypothetical 'best climb' airspeeds at a given CI, the popular practice (or universally accepted subterfuge) is to arbitrarily define the 'climb' to include a portion of cruise by specifying a fixed combined distance, typically 250 nm. You can then pick several values of climb CAS (say in steps of 5 kts) and run an iteration of the cruise distance in each case to maintain 250 nm, finally selecting the CAS that gives the minimum cost. It is generally assumed that the constant-Mach upper portion of the climb (if any) can use roughly the same Mach as that calculated during the cruise portion.

Descent is treated in the same manner, i.e. again over a fixed 250 nm, with initial descent Mach set equal to cruise Mach, and conducting an iterative search for the descent CAS that yields minimum cost at your chosen CI.

Some examples of Manoeuvre Sequences generated according to this philosophy are included in the 'manoeuvres' folder, based on the "B787-8 (502)eis v12ae" plane file (make sure you load the plane before loading and running the sequence).

The more rigorous alternative is to examine total cost for several different choices of CAS while using '**Iterate Sequence**' to fix the overall stage length, instead of relying on arbitrary distance definitions for climb and descent. Strictly speaking you should set up an iteration around top-of-descent mass to maintain a target 'mass at end of sequence' and then select 'Double iteration' to find the TOW that matches your distance. Several examples of entire flights constructed in this way to minimise Block Cost at a given CI are also included in the 'manoeuvres' folder for the 787 (these assume a limit of 350 kcas).

I would emphasize that over a complete flight, total Cost tends to be highly **insensitive** to climb and descent CAS choices (unless you stray very far from optimal altitudes and weights). An operational perspective is key: It's simply not worth trying to greatly refine

such choices given the tremendous uncertainties and assumptions involved in any airline's determination of 'real' CI.

Useful tip: If you run a level Flight Manoeuvre at a given CI with the '**Keep dialog open**' box ticked, you can also check the Cost implications of entering alternative speed options such as a fixed Mach, max SAR, or 99% SAR. The total cost will be calculated for all such entries, using the CI that you entered originally.

Cost Index is only implemented in conjunction with 'Flight Manoeuvres' and is not used in standard range calculations such as 'Range', 'Mission @mass' and 'Mission @range'.

- **New parameters for VMO and MMO**

New parameters are now available to let you define maximum operating speed (VMO) and maximum operating Mach number (MMO) restrictions. These will affect the '**Flight Envelope**' picture and will alter the shape of the red line at low altitudes (that line is now labelled as '*Max Cruise Thrust / VMO /MMO*'). Your inputs may limit some speed choices and therefore affect range calculations in extreme cases, for example if you use '**Range Modes**' to set altitude to a low value like FL 200 and simultaneously choose the '**Max-speed**' option. Normal operations inside the flight envelope are unaffected.

The new parameters are:

user-limited-vmo-eas
user-limited-vmo-cas
user-limited-mmo

Defaults are all set to very high values, so no restrictions will apply unless the user explicitly inputs some.

VMO is a structural limit that is in principle defined in terms of EAS, which is a measure of dynamic pressure. In practice a mixture of philosophies is used. A CAS-based definition offers the operational convenience of being easy to fly and to placard (at the expense of being slightly conservative structurally). You may choose to input either one or both of **user-limited-vmo-eas** and **user-limited-vmo-cas**. If both are provided, the more restrictive limit will be used depending on altitude. For example the 787 appears to be restricted to 350 KEAS at low altitude and 360 KCAS above the corresponding crossover altitude.

Note that operational issues such as low-altitude bird strike on the windscreen, or a desire by the manufacturer to avoid having to certificate certain corners of the flight envelope, can sometimes result in intricately variable VMO limits. The great majority of the aircraft files in the database have nonetheless been assigned straightforward VMO limits in EAS terms. These should be regarded as approximate numbers.

The MMO limit (**user-limited-mmo**) is aerodynamic, not structural, and will naturally dominate at higher altitudes. Once again, suitable values have now been added to most files in the aircraft database.

Note that all these new VMO/MMO parameters are provided purely for the purpose of defining realistic performance constraints. They do NOT influence structural weight estimation, which is conducted exactly as before on the basis of other existing parameters.

Also note that you can still generate performance calculations at speeds that exceed these restrictions, for example when you directly input an arbitrary fixed Mach number in '**Range Modes**' or in '**Flight Manoeuvres**'. The relevant output is shown exactly as before and will simply include a caution.

On a related matter, I've had many conversations with some of you over the years on the question of 'high-speed' Range Modes for business jets. I'd like to stress again that whilst the other modes (such as maxSAR or 99%maxSAR) follow industry practices that are reasonably uniform, there is no such thing as a 'typical high-speed mode'. It is subjective and open to interpretation. A reasonable approach (adopted under 'Range Modes') is to derive a 'representative' Mach at 100% MCR rating combined with 'sensible' use of altitude. If you understand and wish to emulate someone else's notion of a 'high speed' mode (or arbitrary Mach choice), you can always implement it using 'Flight Manoeuvres'. Ultimately however, given VMO and MMO, the answer is unambiguous and simple: Generate a '**Flight Envelope**' picture, making sure to select the '**Airspeed axis**' (you can also untick the 'Find max..' option to reduce outputs). The extreme rightmost point of the red line will correspond to a unique altitude choice with the highest usable true airspeed (it most often occurs at the intersection of the VMO and MMO limits). Simply input that altitude alone in '**Range Modes**' and select '**Max speed +dM**'. That is your definitive 'high speed' mode, period. (To be pedantic, the precise altitude may show minor variation as fuel is burnt off but that effect is totally insignificant and even marginally descending under some circumstances). Although this is indeed the fastest way to fly the aircraft from A to B, it is of course extremely fuel-inefficient and operationally near-irrelevant (except rarely for some turboprops) or ATC-problematic.

- **Drag due to aeroelasticity**

For most aircraft, any additional drag caused by aeroelastic distortions of the airframe is negligible. If you have some knowledge to the contrary, it is now possible to input a simplified correction. The relevant new parameters are:

user-aeroelastic-drag-curve
user-aeroelastic-mdep-curve

The **user-aeroelastic-drag-curve** must be a list of numbers representing alternately the aircraft gross weight (in kg) and a factor to be applied to total aircraft drag at that weight.

The **user-aeroelastic-mdep-curve** need only be used if drag due to aeroelasticity is Mach-dependent. It is a list of numbers representing alternately a Mach number and a factor. This factor will be applied to any weight-dependent drag increments determined per the above. (So this is a factor on any delta-Cd due to aeroelasticity, not on total drag.)

The "B787-8 (502)eis v12ae" is an example file with some estimated adjustments. For this type, significant aeroelasticity is evident in flight at the higher weights. It may also be that

the variable camber trailing edge is used in a controlled manner to shift spanwise loading inboard and provide structural load alleviation at the expense of drag. The situation is likely to improve with increasing operational experience and structural modifications.

Note that using these aeroelastic parameters may generate visible distortions in drag polar plots (as these are plotted at an altitude, so each point can be nominally associated with a lift force).

- **Drag sensitivity to Reynolds number**

A new parameter called **reynolds-sensitivity-factor** is now available. This can be used to alter the internal calibration of the drag estimation methods. The default value of 1 corresponds to the standard methods used in Piano. Reducing the value makes drag less Reynolds-sensitive.

At the extreme, a value of zero can be used to generate drag polars that are totally insensitive to Reynolds number. Such simplified polars were traditionally used by Douglas (and still are today for their legacy aeroplanes) and by many other manufacturers.

In physical terms, sensitivity to Reynolds number reflects the presumed balance between skin friction and pressure drag related components, a near-impossibility to pin down. All Reynolds-related methods are empirical. To the best of my knowledge, Boeing's style of corrections appears to yield behaviour very close to the default Piano estimates, while Airbus-style corrections may be somewhat less sensitive. There does not seem to be any reason to use values higher than 1 in the new parameter, but the option is there.

Changing the value of this parameter will of course modify drag predictions and therefore performance. It is only intended for expert users. No aircraft in the database is affected or includes this parameter.

- **Net gradient in OEI enroute driftdowns**

The following relates to 'net' driftdown paths with one engine inoperative (OEI). Net paths include a regulatory (conservative) adjustment and this can sometimes have operational importance when checking clearances in mountainous terrain. (However, a straightforward engine-out '**Ceiling at..**' calculation is enough in most circumstances. All ceiling calculations remain unchanged and already allow for a net gradient adjustment.)

OEI enroute driftdowns can be easily implemented using **Flight Manoeuvre** sequences. An example of a 90-minute driftdown path is included in the manoeuvres folder (see "b787 netgrad driftdown 90min"). As this example shows, it is best to use reasonably short individual segments at the beginning, when altitude loss is rapid.

The relevant new option is provided in the Flight Manoeuvre dialog under the 'engines operative' item and entitled '**one eng inop, net grad**'. This tells Piano to apply the regulatory 'net gradient' decrement during calculations as specified in FAR/JAR25.123 (which is -1.1% for twin-engined aircraft).

Incidentally, the legal phrasing of FAR/JAR25.123 is ambiguous and merely talks about diminishing 'performance'. It can easily be (mis)interpreted as follows: Establish the entire geometry of the actual (i.e. gross) path, then apply the requisite gradient reduction to get the net path (a trivial geometric transformation). The alternative interpretation is: Reduce instantaneous calculated gradient by the stated amount at each step during flightpath calculations. These two interpretations differ significantly. The latter appears to be what everybody assumes, and this is how the '**one eng inop, net grad**' feature is implemented. It is however more optimistic and (unwisely) tampers with the physics of the solution instead of simply adjusting the actual result. The former interpretation yields a more degraded path.

- **MLW restrictions in Payload-Range diagrams**

For aircraft whose maximum landing weight (MLW) is not sufficiently greater than MZFW, a rare situation can arise whereby taking off at MTOW and maximum payload results in a landing weight exceeding MLW on arrival. This effectively 'cuts off' a small section at the top-right corner of the Payload-Range diagram. Few aircraft (predominantly freighters) display such behaviour. Example files include the "Airbus A340-600 280t" and "B757-200F (255)r".

If this happens, Piano will now show both the original full shape of the Payload-Range diagram and the corner cutoff line, plus a suitable caution in the printout. (The mission can still be flown in principle, but would necessitate a hold on arrival until you have burnt off enough reserve fuel to land).

A warning will also be shown in the Range reports, on the line where Reserves and calculated Landing Mass are listed.

For some types, certificated (or declared, to reduce landing fees) MLW and MZFW may be such that maximum payload is rendered unusable at all ranges with realistic reserves.

- **Maximum Ramp Weight (MRW)**

The certificated maximum ramp weight can now be assigned properly through the new parameter '**max-ramp-minus-mto-mass**' (obviously defined in terms of mto-mass). Default value is zero.

Ramp weights are rarely relevant in design terms. They can however have a minor influence at the extreme-right edge of the payload-range diagram where fuel capacity restrictions apply and when the ramp burn should come out of a full tank. This effect will now be reflected accurately whenever **max-ramp-minus-mto-mass** has been assigned.

The old parameter called '**ramp-fuel-allowance**' is still retained, essentially for backward compatibility. It should now only be used if you want to override the normal taxi-out fuel calculation (which is based on '**taxi-out-time**' and ground idle flows) and instead fix the fuel burn to reflect some known operational behaviour.

By request, the upper limit for **taxi-out-time** has also been increased, to a mind-numbing level. At some of the busiest US hubs you can now regularly expect to be still sat on the ground 60 minutes or more after startup. If APU usage is significant, you may choose to apply a **user-factor-on-taxi-out-fuel** or set the ramp burn directly. If you do this, be aware that taxi-out pollutant calculations (other than CO2) will be invalidated, as these depend not just on fuel burn but also on how it happens - and it happens differently inside an APU and an idling engine. Piano has no knowledge of APU behaviour.

Note that cautions about 'required versus available' fuel volume may appear slightly more conservative when **max-ramp-minus-mto-mass** is used. They will be issued if there is insufficient tank capacity at MRW and nominal design payload, even though takeoff may still be just possible at full MTOW for normal taxi fuel. These messages are only advisory and intended for design sizing purposes. They don't alter any calculations or behaviour.

(p.s. An earlier restriction of **ramp-fuel-allowance** to values not exceeding **max-ramp-minus-mto-mass** no longer applies.)

- **Alternative 'theta-exponent' for engine fuel flows**

Recall that if you need to create a new engine model, you can input fuel flows in a normalised form based on a **theta-exponent**, whose value you specify in the engine's 'fuel flow' file. As explained in the guide, 'theta' refers to the atmospheric temperature ratio (fraction of sea level value).

Some engineers prefer to work with data normalised in terms of stagnation (or 'total') temperatures and pressures instead of atmospheric values. They will then refer to 'corrected fuel flows' based on total theta and total delta. Thermodynamic justifications aside, the difference is merely one of convenience. Since atmospheric and stagnation values are related purely through Mach, and fuel flows are given for lines of constant Mach, either modelling approach will yield precisely the same final result.

It is now equally possible to input engine data based on the 'stagnation' definition as for the usual 'atmospheric' definition. You simply inform Piano that you intend to use the stagnation convention by replacing the word '**theta-exponent**' with the word '**theta-total-exponent**' in the engine's 'fuel flow' file. Obviously, you must also be sure to give your data in the form corresponding to your choice! The new option is restricted to 'fuel flow' files and is not available with 'sfc loops' files.

- **Other changes**

The following bug fixes are implemented:

- The fuel burn shown for 'T/O to screen height' in mission profiles produced via the automated sequence generator ('Flight' -> 'Build a Sequence') was previously set incorrectly. The right value is now shown. This does not affect standard range calculations or in-flight manoeuvres. It only makes a small difference to block totals from the sequence builder.

- Piano could under some circumstances fail to locate the 'manoeuvres' folder when Saving or Loading manoeuvre sequences, forcing you to navigate to it manually. This has now been corrected.

The following minor change in logic has been implemented:

- Recall that in 'Flight Manoeuvres', you can use 'Stop when ready to stepup by..' to determine the appropriate cruise stepup point. Previously, this option demanded cruising at a strictly constant Mach. This logic has now been relaxed to also permit airspeeds such as 'max SAR' or '99% SAR' or 'economy at cost index'. Theoretically this introduces a small inaccuracy when selecting the 'best' stepup point since Mach number is mildly variable. However, performance is very insensitive to mild deviations from optimal stepup point under normal conditions. The ease of defining the cruise more readily is well worth the slight approximation implicit in this approach, should you choose to use it.

As usual, a variety of minor efficiency and stability improvements are also included that have no externally obvious effects.

Looking forward to your feedback.

Regards,
Dimitri