Supporting Information

A bivariate mixed-model perspective on the QCL-EC vs. ALPHA-IHF method comparison

2026-07-17

Table of contents

1. Overview

This Supporting Information provides additional methodological details and results that complement the main analysis.

It is organised as follows: Section 3 sets out the theoretical basis of the Bland-Altman procedure, in particular why a non-zero slope in the plot does not always mean that one method is biased. Section 4 fits a bivariate mixed model for both endpoints (NH_3 concentration and NH_3 emission) and uses it as a diagnostic tool. Section 5 shows how this bivariate view relates algebraically to the simpler univariate model used in the main text. Section 6 extends the model with covariates. Section 7 discusses limitations of the statistical analysis and open issues for further investigation.

The next section summarises all notation used throughout; the sections that follow can be read against it.

2. Notation

Throughout, the index i labels a single measured time point - one exposure interval at which both methods produced a value. For each such i, X_{1i} and X_{2i} denote the two measurements on the **log scale**, with method 1 = QCL(-EC) and method 2 = ALPHA(-IHF). Consequently $d_i = \log(\mathrm{QCL}/\mathrm{ALPHA})$ is the log ratio and m_i the log mean (denoted `log_mean` in the main-text models). The general mean structure is $\mu_{ki} = \delta_k + \beta_k T_i$: each method may carry a constant offset δ_k and scale the common true signal T_i by a factor β_k. The additive model used from Section 4 on is the special case $\beta_1 = \beta_2$, normalised to $\beta_1 = \beta_2 = 1$ since the scale of the latent T_i is free (a constant offset only, no proportional bias); with $\delta_1 = \delta_2$ in addition there is no bias at all (used in point (ii) and in the toy example).

Symbol	Meaning
$X_{1i},\ X_{2i}$	log-scale measurement of unit i by method 1 (QCL-EC) and method 2 (ALPHA-IHF)
$\mu_{1i},\ \mu_{2i}$	expected values of the two measurements
T_i	common underlying (“true”) signal on the log scale
$e_{1i},\ e_{2i}$	method-specific errors
$d_i = X_{1i} - X_{2i}$	Bland-Altman difference, i.e. the log ratio $\log(\mathrm{QCL}/\mathrm{ALPHA})$
$m_i = 1/2\,(X_{1i} + X_{2i})$	unweighted mean (`log_mean` in the main text)
$w_i = a\,X_{1i} + (1-a)\,X_{2i}$	precision-weighted mean
$\sigma_1^2,\ \sigma_2^2$	marginal variances of the two measurements, $\sigma_k^2 = \beta_k^2\tau^2 + \sigma_{e,k}^2$ - the

Symbol	Meaning
	overall variability of each method, combining how strongly it scales the signal and its measurement noise; the trend identity $\mathrm{cov}(d, m) = \frac{1}{2}(\sigma_1^2 - \sigma_2^2)$ is stated in these
$\sigma_{12} = \rho\,\sigma_1\sigma_2$	covariance between methods; ρ is estimated from the bivariate model on the log scale (distinct from the original-scale Pearson r)
τ^2	variance of the common signal T_i; under the additive model this equals the between-method covariance, $\tau^2 = \sigma_{12} = \rho\sigma_1\sigma_2$; in general $\sigma_{12} = \beta_1\beta_2\tau^2$ (with independent measurement errors)
σ_e^2	method-specific residual variance (additive model)
β_k	factor by which method k scales the true signal T_i; $\beta_1 = \beta_2$ means no proportional bias; the additive model fixes the common value to 1
a	weight on method 1 in w_i

3. Theoretical foundation

This section sets out the statistical reasoning behind the Bland-Altman procedure used in the main text. When inspecting a Bland-Altman plot (BAP), the first step is to establish whether the two methods differ on average: a systematic vertical offset of the differences away from zero indicates a *constant bias*, one method reading consistently higher or lower than the other regardless of the level. The next step is to check whether the differences also show a trend across the measurement range. A non-zero slope is often taken to indicate a *proportional bias* between the two methods, i.e. that the gap between them grows or shrinks systematically as the measured level increases. While this interpretation is intuitive, it is not the only possibility for a non-zero slope: a difference in *precision* alone - one method simply being noisier than the other - can generate the same trend even when neither method is biased. Therefore, a slope can have two possible causes: a proportional bias or a difference in precision. As panel F of Figure S1 shows, the two causes cannot be distinguished from the plot alone.

Figure S1 illustrates the possibilities. The columns show the *systematic* difference between the methods (no bias, a constant bias, or a level-dependent proportional bias), and the rows show the *precision* (equal or unequal measurement noise). The top row represents the familiar, straightforward case in which both methods are equally precise and where the BAP performs as expected. The bottom row illustrates what happens when one method is noisier than the other. The reasoning below first sets out the

general model (point (i)) and considers the top row (point (ii)), before turning to the bottom row (point (iii)).

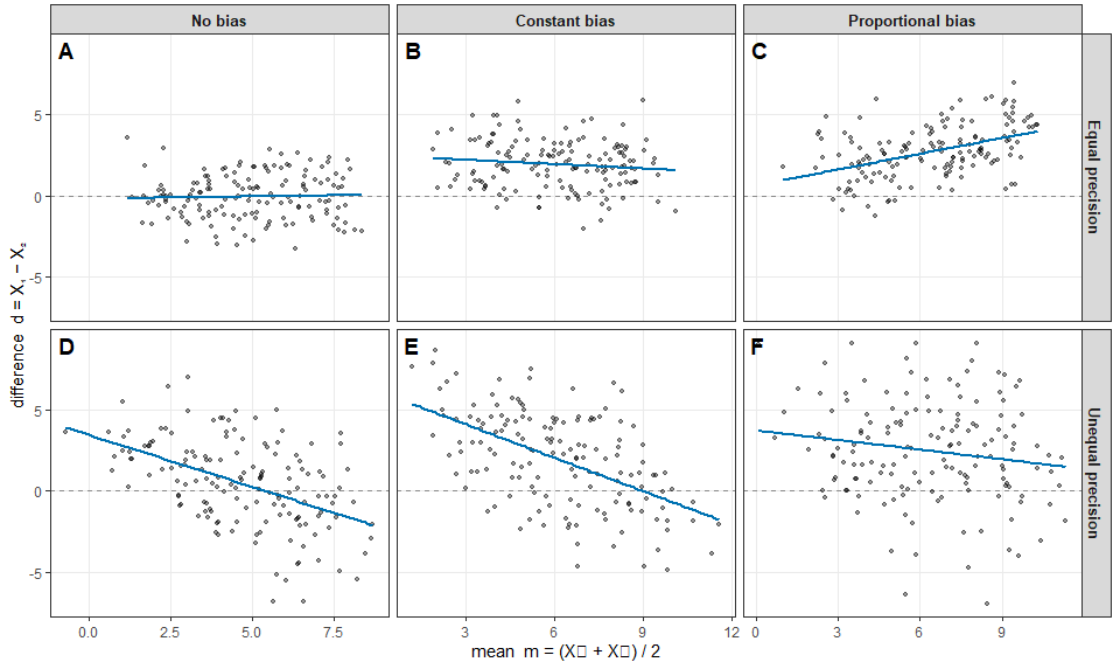

Fig. S1: Bland-Altman patterns of the six ways two methods can differ, crossing the systematic difference (columns: no bias; a constant bias; a proportional bias with method 1 = k × method 2) with precision (rows: equal vs. unequal error variances). Synthetic data, $n = 150$ per panel: true level T uniform on $[2,8]$; error SDs $\sigma_{e,1} = \sigma_{e,2} = 1$ in the top row and $\sigma_{e,1} = 1$, $\sigma_{e,2} = 2.5$ in the bottom row; constant offset $c = 2$; proportional factor $k = 1.5$. Dashed line $d = 0$; blue line the fitted slope. ***(A)*** *No bias, equal precision: the methods agree on average with equal scatter, and the plot shows a flat cloud around zero - the ideal reference.* ***(B)*** *Constant bias: the methods differ by a fixed $c = 2$ at every level; the plot shows the same flat cloud shifted up to $d \approx 2$, so a constant bias reads as a vertical offset, not a slope.* ***(C)*** *Proportional bias: the gap genuinely grows with the level, and the plot shows a real upward slope (≈ 0.3) - here the slope is a true signal.* ***(D)*** *No bias, unequal precision: on average the methods agree exactly ($E(d) = 0$), only the second scatters more, yet the plot shows a clear downward slope (≈ -0.6) around zero - a pure variance artefact.* ***(E)*** *Constant bias with unequal precision: the fixed offset and the artefactual slope simply add, giving a downward slope around $d \approx 2$.* ***(F)*** *Proportional bias with unequal precision: the data carry a genuine upward proportional trend, but the plot shows a slight downward slope (≈ -0.2) - the variance artefact masks and even reverses the real trend, and the two sources cannot be separated from this plot alone (point (viii)).*

(i) A simple model underlying the BAP writes each measurement as an expected value plus error,

$$X_{1i} = \mu_{1i} + e_{1i}, \qquad X_{2i} = \mu_{2i} + e_{2i},$$

where X_{1i}, X_{2i} are the measurements of unit i by methods 1 and 2, e_{1i}, e_{2i} their errors, and the expected values $\mu_{ki} = \delta_k + \beta_k T_i$ let each method carry a constant offset δ_k and scale the common true level T_i by a factor β_k; the errors are taken to be independent of the true level T_i. These three types of systematic difference correspond to different constraints on these parameters:

- *No bias*: $\delta_1 = \delta_2$ and $\beta_1 = \beta_2$.
- *Constant bias*: $\delta_1 \neq \delta_2$ (with $\beta_1 = \beta_2$).
- *Proportional bias*: $\beta_1 \neq \beta_2$.

In the BAP the difference $d_i = X_{1i} - X_{2i}$ is plotted against the mean $m_i = \frac{1}{2}(X_{1i} + X_{2i})$.

(ii) *Equal precision (top row).* Assume that both methods are equally precise and observe the effect of each type of systematic difference on the plot.

- *No bias* (panel A): both methods have the same expected value at every level, so on average they measure the same thing and the differences centre on zero, $E(d_i) = 0$. The points scatter around the horizontal zero line with the same spread from left to right - a flat cloud, no trend and no funnel. This is the ideal reference.
- *Constant bias* (panel B): one method reads a fixed amount higher at every level ($\delta_1 \neq \delta_2$). Every difference is shifted by that same amount, so the identical flat cloud simply moves up (or down) to $E(d_i) = \delta_1 - \delta_2$. A constant bias shows as a vertical offset and never as a tilt.
- *Proportional bias* (panel C): the gap between the methods grows with the level ($\beta_1 \neq \beta_2$; one method reads a fixed *fraction* higher). The differences then increase systematically from left to right and the cloud genuinely tilts. Here the slope is a real signal about the methods.

At equal precision, the reading is therefore clear: in this model a vertical offset represents a constant bias, and a slope indicates a proportional bias. This is the case in which the BAP is usually read.

(iii) *Unequal precision (bottom row).* The picture changes as soon as one method scatters more than the other. The reason is visible in the covariance between the difference and the mean, which is a measure of how the two move together, and therefore of the trend in the plot:

$$\operatorname{cov}(d_i, m_i) = \operatorname{cov}\left(X_{1i} - X_{2i},\, \frac{1}{2}(X_{1i} + X_{2i})\right) = \frac{1}{2}(\sigma_1^2 - \sigma_2^2),$$

where σ_1^2 and σ_2^2 are the *marginal* variances - the overall variability of each method, which is not the same as its precision (point (iv)). Put simply, the plot only tilts to the extent that the two methods differ in overall variability, and if these marginal variances are equal the covariance is zero and no trend arises. As point (iv) shows, this overall variability also carries any proportional bias, which is why panel C could tilt despite both methods being equally precise. The simplest way for the variabilities to differ is through a pure difference in noise. Neither method is biased, both track the same signal, but the

second is noisier. A downward trend appears with no underlying bias: panel D, where the methods agree on average ($E(d_i) = 0$) yet the fitted line clearly slopes (by about -0.6). A trend created solely by unequal noise like this is referred to as a *variance artefact*. This is an apparent slope that reflects differing precision, rather than a real disagreement between the methods. It is the effect the Pitman-Morgan test for equal variances in paired samples picks up through the correlation between d_i and m_i (see Piepho, 1997). Plotting d_i against the mean m_i, rather than against one method, is the standard choice, precisely because this reduces the induced correlation. However, it does not remove the correlation when the variances differ.

(iv) *The two causes are confounded.* The overall variability of a method has itself two sources - how strongly it scales the signal and how much noise it adds:

$$\sigma_k^2 = \beta_k^2\,\tau^2 + \sigma_{e,k}^2,$$

with τ^2 the variance of the true level and $\sigma_{e,k}^2$ the measurement noise. Substituting this into the covariance splits the trend into two additive contributions,

$$\operatorname{cov}(d_i, m_i) = \underbrace{\frac{1}{2}(\beta_1^2 - \beta_2^2)\tau^2}_{\text{proportional bias}} + \underbrace{\frac{1}{2}\left(\sigma_{e,1}^2 - \sigma_{e,2}^2\right)}_{\text{difference in precision}}.$$

The first term is a genuine proportional bias (panel C), while the second is a variance artefact (panel D). Crucially, the plot only responds to their *sum*: it shows a single slope and cannot reveal which term produced it. Panels E and F illustrate this.

- *Panel E* (constant bias, unequal precision): a constant bias enters neither term, adding only a vertical shift (point (ii)), so the slope remains a pure variance artefact, essentially equivalent to that of panel D (≈ -0.6), now positioned around $d \approx 2$ instead of zero.
- *Panel F* (proportional bias, unequal precision): here both terms are active but they have opposite signs. The data carry *exactly the same* genuine proportional bias as panel C, which has an *upward* tilt (of about $+0.3$) - but the second method is noisier, so the negative precision term is larger and dominates. The two contributions add in the covariance, and the visible slope is *negative* (approximately -0.2). The artefact not only dampens the real trend, but also outweighs and reverses its sign. Consequently, the plot points in the wrong direction with regard to the underlying relationship.

The lesson is that a single BAP slope cannot, on its own, be split into “how much is bias” and “how much is precision difference”: a non-zero slope is equally compatible with a genuine proportional bias, with a pure variance artefact, or with any mixture of the two. (Technically, the fitted slope equals $\operatorname{cov}(d_i, m_i)/\operatorname{var}(m_i)$, and because $\operatorname{var}(m_i)$ itself differs across panels, it is the two covariance *terms* that add, not the panel slopes. The net slope in panel F is not the arithmetic sum of the slopes in panel C and panel D.) Therefore, reading the plot again calls for a better x-axis than m_i - the subject of point (v).

(v) *Choosing a more appropriate x-axis.* Points (iii)-(iv) leave the reader stuck: a non-zero slope is *compatible* with pure variance heterogeneity, but the classical plot cannot show whether this is the case. Our data fall firmly within the bottom row, and the two methods differ measurably in precision (Section 4). Therefore, a clear top-row interpretation, in which a slope can only mean a proportional bias, is not directly applicable. Ideally, one would plot the difference against the *true level* T_i; the mean m_i is only a substitute for it and may be an inaccurate one, because it carries the noise of *both* methods - including the noise of the less precise method that generated the artefactual trend in the first place. Adjusting the x-axis does not determine whether the trend is a real proportional bias or a variance artefact - the two cannot be separated here (point (viii)) - but it allows us to verify the validity of the artefact reading.

A better alternative for T_i relies on the more reliable method. It replaces m_i by a precision-weighted mean

$$w_i = a\,X_{1i} + (1-a)\,X_{2i},$$

where a is the weight on method 1 (empirically the more precise one here, so $a > 1/2$). Its covariance with the difference is

$$\mathrm{cov}(d_i, w_i) = a\,\sigma_1^2 - (1-a)\,\sigma_2^2 + (1-2a)\,\sigma_{12},$$

which becomes zero for

$$a = \frac{\sigma_2^2 - \sigma_{12}}{\sigma_1^2 - 2\,\sigma_{12} + \sigma_2^2}.$$

In the special case of equal marginal variances $\sigma_1^2 = \sigma_2^2$ (which, under the additive model, means equal precision), $a = 1/2$, and w_i is simply the ordinary mean m_i: the correction departs from the familiar plot only to the extent that the two methods differ in overall variability. Estimating Σ from a bivariate model (Section 4) yields a and hence w_i, and plotting d_i against w_i gives the corrected BAP. The same construction has been proposed independently as a solution for Bland-Altman analysis under unequal precision (Borga, 2023): under the additive model a is exactly the inverse-error-variance weight used there, supporting the validity of the weighting approach. The approaches differ in one respect that matters here - Borga estimates each method's precision directly from replicate measurements, meaning his correction can genuinely recover the underlying relationship. In contrast, we have a single measurement per method on each occasion and must infer the error variances from the fitted Σ under the additive model. This is why the admissibility check below is important here, but not in his setting.

To observe this scenario, we apply the correction to case D of Figure S1: no bias is present, but the second method is noisier. Figure S2 shows the result: its middle panel (H) is qualitatively the same case as panel D of Figure S1 - the classical Bland-Altman plot with its variance-artefact slope - while the right panel (I) replaces m_i with w_i and flattens the trend; the left panel (G) shows the underlying scatter. The real-data version is Figure S3.

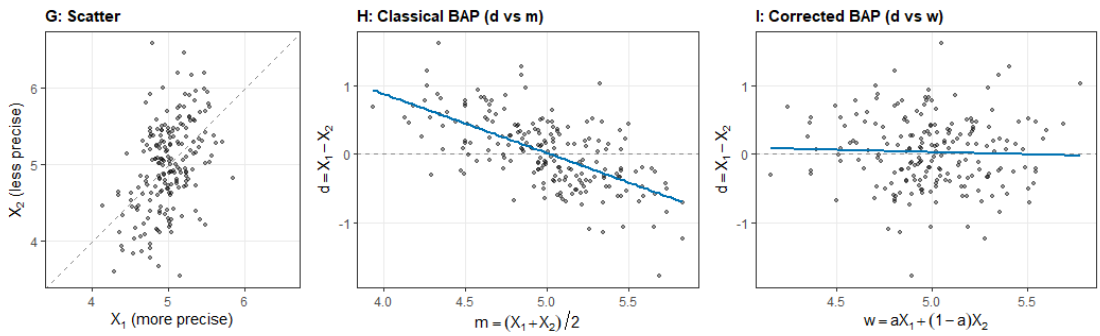

Fig. S2: Synthetic data with identical true means ($\mu_1 = \mu_2 = 5$) but different error standard deviations ($\sigma_1 = 0.3$, $\sigma_2 = 0.6$, $\rho = 0.4$, $n = 200$) - a clean instance of case D of Fig. S1 (no bias, unequal precision). The panels are lettered G-I to keep them distinct from the cases A-F above. Read them left to right: no bias (G) still produces a clear slope (H), and re-choosing the x-axis removes it (I). ***Panel G:*** *X_1 vs. X_2. The cloud is centred on the identity line (equal means, hence no bias) but tilted and vertically stretched because X_2 is the noisier method - and it is this tilt that produces the apparent slope in Panel H.* ***Panel H:*** *the classical BAP shows a clear negative trend (slope ≈ -0.87) - a variance artefact, not a method bias, and the same pattern as panel D of Fig. S1.* ***Panel I:*** *plotting d against the precision-weighted mean $w = aX_1 + (1-a)X_2$ removes the trend (slope ≈ -0.07). Because w leans on the more precise method, it is a better stand-in for the unknown true level than m - here $a = 0.94$, placing most of the weight on the more precise method 1. This is the precision-weighted correction introduced in point (v).*

Two caveats apply. First, a near-zero slope against w_i is built in. The weight a is defined precisely so as to cancel the trend: if a were taken directly from this sample's own covariances the corrected slope would be exactly zero, and because we instead estimate a from the fitted Σ it comes out near zero (up to estimation error). Either way, flatness is essentially guaranteed by construction, so a flat corrected plot on its own proves nothing. What makes it meaningful is where a comes from. It is not tuned by hand but derived from the estimated variance-covariance structure Σ of the two methods (Section 4), and that same Σ must be consistent with a simple model of the data: one shared true signal that both methods track, each adding its own independent measurement noise (the *additive model*). Whether the data are compatible with this model can be checked directly - it requires the estimated noise variances to come out non-negative. When they do, a difference in precision is enough to account for the trend, and one need not suppose the methods genuinely disagree. When they do not, the trend cannot be dismissed as a precision artefact, and forcing w_i would hide a real effect rather than remove a spurious one - the difficult case discussed in point (viii).

Second, w_i serves as a *diagnostic and display* device, not as a covariate: because it is uncorrelated with d_i by construction, it cannot *control* for the measurement level in a regression and would absorb nothing. That control role falls instead to the mean level itself in the main-text covariate model (Section 5).

The effect of the weighting can be placed against the three features a Bland-Altman plot can show (offset and trend, points (i)-(iv); scatter, point (vi)). The *constant bias* - the vertical offset of the differences, read off as the dashed line - is left untouched, since a genuine average difference between the methods may be real and is reported separately. The *trend* is flattened; under the additive model it is read as a precision artefact rather than as a proportional bias. This does not *rule out* a proportional bias; the two are indistinguishable from the plot alone (point (viii)). It adopts the more parsimonious interpretation offered by an admissible additive model. The *scatter* of the differences remains unaffected - point (vi) addresses this.

(vi) The weighting removes the *trend* in the differences; it says nothing about their *spread*. If $\mathrm{var}(d_i)$ is itself not constant across observations - the scatter growing or shrinking with the measurement level - the variance function can be estimated and used to standardize the BAP (Sadler, 2019). This also works when w_i is used in place of m_i.

(vii) Repeated measures: the procedure is essentially unchanged, since d_i and w_i (or m_i) can still be formed per observation unit; the unit is simply extended to include time (time point × physical unit). In other words: as long as both methods share the same time-correlation pattern, the difference and the mean inherit it and the earlier argument remains unchanged. Formally, the problem simplifies when the joint error variance-covariance can be written as a Kronecker product $\Sigma \otimes R$ (the between-method structure Σ repeated across a time-correlation structure R common to both methods); then d_i and m_i inherit R. Points (i)-(v) assumed $R = I$ (independent units).

(viii) The genuinely difficult case arises when the methods do not merely differ in noise level but also scale the common signal differently; then a trend and unequal variances look alike and cannot be distinguished - panel F of Figure S1, where a genuine proportional trend and a variance artefact combine into a single, uninterpretable slope. Formally, the open question is how to estimate the variance structure Σ. If both methods scale the true value T_i in the same way (so any bias is at most a constant offset), a simple mixed model suffices. It becomes difficult if, on top of variance heterogeneity, the two methods scale T_i differently, since this implies a multiplicative method × unit interaction. A non-zero BAP slope does not by itself point to this case - it is equally compatible with pure variance heterogeneity (point (iv)) - which is exactly why the two cannot be separated from the plot alone. Distinguishing a genuine interaction from variance heterogeneity, let alone fitting both jointly, is difficult (Snee, 1982).

4. The bivariate model as a diagnostic tool

Point (v) introduced the precision-weighted mean but took two things for granted: the weight a, and the assurance that the differences can genuinely be read as one shared signal measured at two precisions (the admissible additive decomposition). Both come from fitting the two methods *jointly* as a bivariate model, which is the step this section makes explicit. The main-text Bland-Altman analysis works with the difference d_i and the mean of the two methods; the joint fit is an equivalent but more explicit view of the same data that recovers the variance-covariance structure Σ directly and, with it, the quantities of Section 3: the method-specific error variances, the within-unit correlation, and the

weight a. It is used here as a diagnostic device - to check whether the Bland-Altman trend is a variance artefact - not as a competing primary analysis.

4.1 Model specification

For each endpoint the two log-scale measurements of every unit are stacked into long format and modelled with a method-specific mean, a within-unit correlation shared across methods (`corCompSymm`), and method-specific error variances (`varIdent`):

```
# Bivariate model for one endpoint. `ycols` names the two method columns; the
# first becomes the reference level of `method`.
fit_bivariate <- function(d, ycols) {
  dl <- d |>
    mutate(id = row_number()) |>
    select(id, all_of(ycols)) |>
    pivot_longer(all_of(ycols), names_to = "method", values_to = "y") |>
    mutate(method = factor(method, levels = ycols), log_y = log(y))

  m_het <- gls(log_y ~ method,
               correlation = corCompSymm(form = ~1 | id),
               weights     = varIdent(form = ~1 | method),
               data = dl, method = "REML")
  m_hom <- gls(log_y ~ method,
               correlation = corCompSymm(form = ~1 | id),
               data = dl, method = "REML")

  # gls$sigma is the residual SD of the *reference* method; varIdent returns
a
  # multiplier for the other one. The reference need not be the first factor
  # level, so recover both SDs explicitly via setdiff().
  base  <- m_het$sigma
  mult  <- coef(m_het$modelStruct$varStruct, unconstrained = FALSE)
  ref   <- setdiff(levels(dl$method), names(mult))
  sigma <- c(setNames(base, ref), base * mult)
  s1 <- unname(sigma[ycols[1]]); s2 <- unname(sigma[ycols[2]])
  rho <- unname(coef(m_het$modelStruct$corStruct, unconstrained = FALSE))

  # Precision weight a (so that cov(d, w) = 0) and additive decomposition
  # Sigma = tau^2 J + diag(sigma_e^2).
  a    <- (s2^2 - rho * s1 * s2) / (s1^2 + s2^2 - 2 * rho * s1 * s2)
  tau2 <- rho * s1 * s2
  lrt  <- anova(m_hom, m_het)

  tibble(sigma1 = s1, sigma2 = s2, rho = rho, a = a, tau2 = tau2,
         se1 = s1^2 - tau2, se2 = s2^2 - tau2,
         dAIC = lrt$AIC[1] - lrt$AIC[2])
}

fit_conc <- fit_bivariate(dat_conc, c("QCL", "ALPHA"))
fit_em   <- fit_bivariate(dat_pos,  c("QCL-EC_flux", "ALPHA-IHF_flux"))
```

The `varIdent` term relaxes the equal-variance assumption (Section 3, points (iii)-(iv)), and `corCompSymm` introduces the correlation ρ that links the two measurements of the same unit. In the notation of Section 3, this single model estimates the whole variance-covariance matrix Σ at once.

> ⓘ Technical note: recovering the two error SDs in nlme
>
> In `nlme`, `gls$sigma` is the residual SD of the *reference* method, while `varIdent` reports a multiplier for the other method - and the reference is not necessarily the first factor level. Both method SDs are therefore recovered explicitly (via `setdiff()` in the code above) rather than read off directly.

4.2 Empirical results: concentrations and emissions

The estimated model parameters for both endpoints are summarised below; they reproduce Table 3 of the main text.

Quantity	Meaning / interpretation	Concentration (n = 86)	Emission (n = 59)
σ (QCL)	Total observed standard deviation of QCL measurements, including shared signal and method-specific error	0.705	1.123
σ (ALPHA)	Total observed standard deviation of ALPHA measurements, including shared signal and method-specific error	1.130	1.542
ρ (within-unit)	Within-unit correlation between paired QCL and ALPHA measurements	0.612	0.587
a (weight on QCL)	Relative weight of the QCL measurement in the precision-weighted mean	0.987	0.848
τ² (shared-signal variance)	Variance component representing the shared true signal between both methods	0.488	1.017
σ²_e (QCL)	Estimated method-specific error variance of QCL measurements	0.010	0.244
σ²_e (ALPHA)	Estimated method-specific error variance of ALPHA measurements	0.789	1.362
ΔAIC (heterogeneous vs. homogeneous model)	AIC improvement of the heterogeneous-variance model over the homogeneous one (larger	25.5	6.6

Quantity	Meaning / interpretation	Concentration (n = 86)	Emission (n = 59)
	= heterogeneous model clearly preferred)		

For both endpoints the ALPHA(-IHF) side carries the larger measurement (marginal) SD (ratio 1.6 for concentrations, 1.4 for emissions); the underlying error-variance gap (σ²_e in the table) is larger still, so ALPHA(-IHF) is the less precise method. The weight a is 0.99 for concentrations and 0.85 for emissions: the precision-weighted mean w therefore places almost all (concentrations) or most (emissions) of its weight on the QCL(-EC) side. The heterogeneous-variance model is strongly preferred over its homogeneous counterpart (ΔAIC = 25.5 and 6.6, respectively).

4.3 Variance decomposition

Under the additive model - which sets $\beta_1 = \beta_2 = 1$, ruling out proportional bias (point (viii)) - the variance-covariance matrix factorises as

$$\Sigma = \tau^2 J + \mathrm{diag}\left(\sigma_{e,1}^2,\ \sigma_{e,2}^2\right),$$

where τ^2 is the variance of the shared underlying signal T_i and $\sigma_{e,k}^2$ is the error variance specific to method k (method 1 = QCL, method 2 = ALPHA); J is the 2×2 matrix of ones. This decomposition is valid only if both method-specific variances are non-negative. That holds for both endpoints (table above), so the data are consistent with both methods measuring the same true value at different precision, and the additive model is admissible. A multiplicative method × unit interaction is therefore not required to explain the Bland-Altman trend, though it is not excluded either (Section 7).

For the concentration endpoint, however, $\sigma_{e,QCL}^2 = 0.01$ is only marginally positive. This near-zero value depends on excluding two ALPHA measurement failures; without that exclusion the additive model would fail. The sensitivity is taken up in Section 7.

4.4 Corrected Bland-Altman plots

Replacing the unweighted mean by the precision-weighted mean $w = a\,X_1 + (1-a)\,X_2$ on the x-axis removes the trend for both endpoints (Fig. S3) - the real-data counterpart of panel I in Figure S2. As explained in point (v), a flat corrected plot does not provide evidence on its own. It is informative only in combination with the estimated covariance matrix Σ, from which the weight a is derived and which is compatible with an admissible additive model: both error variances are positive in the variance decomposition above (subject to the outlier noted there for concentrations). Given admissibility, the trend *may* be read as a pure precision artefact rather than a proportional bias (the two cannot be separated here, point (viii)): the residual scatter no longer carries the variance-driven slope - the panel-D pattern of Figure S1 restored to the panel-A ideal - and conventional limits of agreement (bias $\pm$ 1.96 SD) become appropriate as far as the trend is concerned (their width still presumes constant scatter, revisited in Section 7). The mean log ratio (dashed line) is positive for concentrations but negative for emissions: QCL reads higher than ALPHA for concentrations, whereas QCL-EC reads lower than

ALPHA-IHF for emissions. The opposite signs reflect that the two endpoints are derived through entirely different physical pathways.

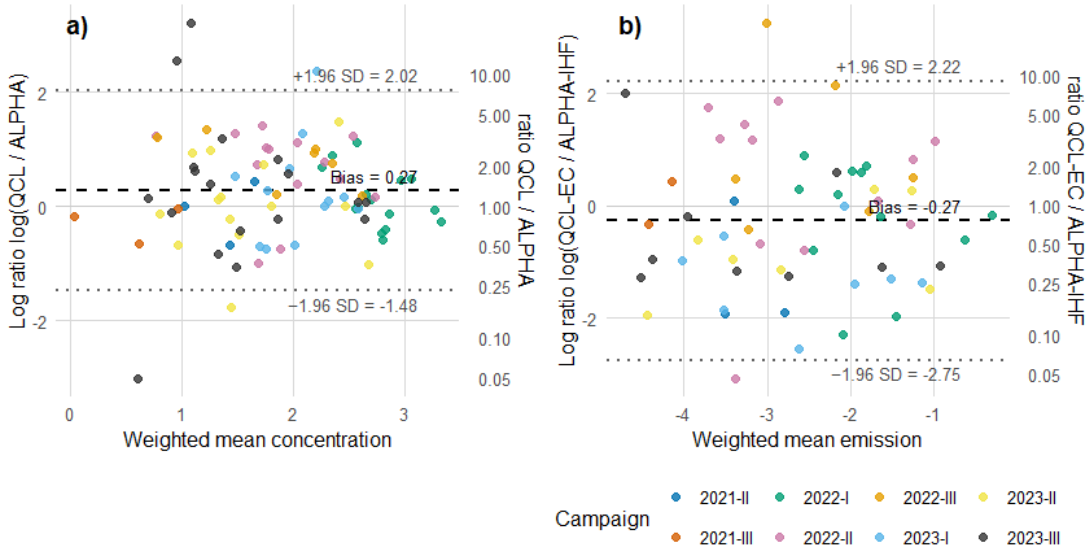

Fig. S3: Bland-Altman plots with the precision-weighted mean w on the x-axis, for (a) concentrations and (b) emissions. Points are coloured by campaign; the dashed line is the mean log ratio (bias) and the dotted lines the 95% limits of agreement. The right-hand axis shows the corresponding ratio. After weighting, the residual scatter is free of the variance-driven trend.

5. Relationship between the bivariate and univariate analyses

The bivariate model of Section 4 and the univariate log-ratio analysis of the main text are not competing descriptions of the data; they are algebraically linked. This section makes that link explicit for one representative covariate, explains why the main text adopts the simpler univariate form, and outlines what the bivariate view adds.

5.1 Algebraic relationship

Let U_i be a covariate measured once per unit and therefore shared by both methods (for the emission endpoint we use the friction velocity USTAR). Allowing a method-specific slope, the bivariate model is

$$X_{ki} = \beta_{0k} + \beta_{1k}\, U_i + e_{ki}, \qquad k \in \{1,2\},$$

with method 1 = QCL-EC and method 2 = ALPHA-IHF. Taking the within-unit difference returns the log ratio,

$$d_i = X_{1i} - X_{2i} = (\beta_{01} - \beta_{02}) + (\beta_{11} - \beta_{12})\, U_i + (e_{1i} - e_{2i}),$$

so the slope of the log ratio on U is exactly the difference of the two method-specific slopes, $\beta_{11} - \beta_{12}$. Because U_i is shared by both methods, it cancels in the difference: the univariate regression $d \sim U$ and the method $\times U$ interaction of the bivariate model estimate the same quantity and coincide numerically, not merely approximately.

Quantity	Slope (per 1 SD)
Bivariate: USTAR slope of ALPHA-IHF	0.566
Bivariate: USTAR slope of QCL-EC	0.035
Bivariate slope difference (QCL-EC − ALPHA-IHF)	-0.530
Univariate: log ratio ~ USTAR (no log mean)	-0.530
Univariate: log ratio ~ log mean + USTAR (main-text control structure, USTAR alone)	-0.440

The check confirms the identity for USTAR on the emission endpoint: the bivariate slope difference equals the univariate slope of the log ratio on USTAR to the displayed precision (both -0.530 per 1 SD). The decomposition is itself informative - ALPHA-IHF carries essentially all of the USTAR dependence (+0.566 per 1 SD) while QCL-EC barely responds (+0.035); this attribution is pursued in Section 6.

Adding the log mean as a control regressor - as the main-text models do - shifts the USTAR slope slightly, to -0.440 (shown here for USTAR alone; the full main-text model adjusts for the remaining covariates as well). The log mean absorbs the variance-driven trend of Section 3, so that trend does not leak into the covariate slopes; its own coefficient is a control and is not interpreted. Including it therefore isolates the USTAR effect without conflicting with the algebraic relationship above.

5.2 Visual reading: parallel or divergent regression lines

This algebraic relationship has an immediate graphical reading. If both methods are regressed on the same covariate U, the per-method regression lines have slopes β_{11} and β_{12}. Their *difference*, $\beta_{11} - \beta_{12}$, is exactly the slope of the log ratio on U derived above. Hence

- **parallel lines** ($\beta_{11} = \beta_{12}$) ⇔ no method $\times U$ interaction ⇔ the log ratio is flat in U, and
- **divergent lines** ⇔ a method $\times U$ interaction ⇔ the log ratio has a non-zero slope in U.

Fig. S4 shows the two endpoints side by side, each plotted against the covariate associated with the largest difference between the methods. The interaction can be read directly from the plot - parallel lines mean a flat log ratio, diverging lines a sloped one.

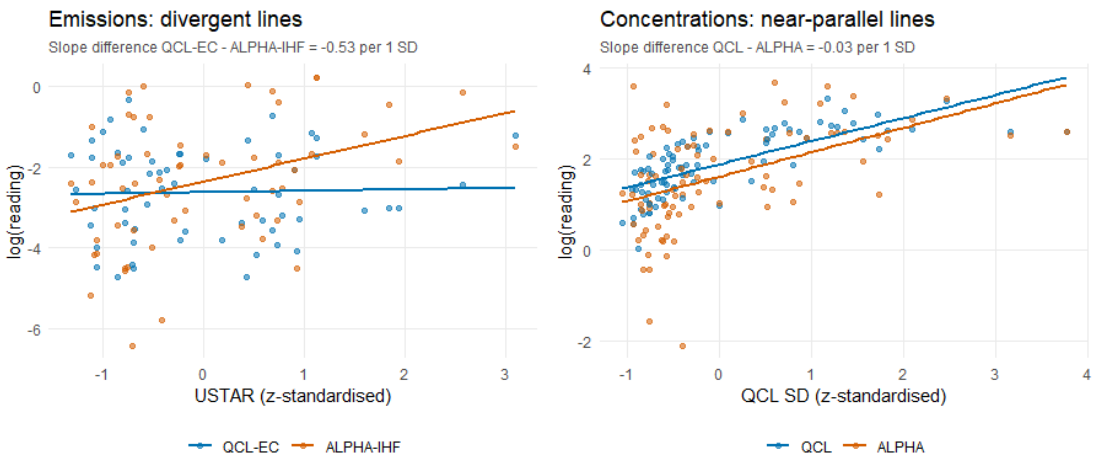

Fig. S4: Per-method log readings against a representative covariate, with simple linear regressions (blue = QCL(-EC), vermillion = ALPHA(-IHF)). Left: emissions on USTAR - the two lines diverge (QCL-EC 0.04, ALPHA-IHF 0.57 per 1 SD; difference -0.53), so the log ratio has a non-zero slope in USTAR. Right: concentrations on QCL SD - the two lines are nearly parallel (QCL 0.51, ALPHA 0.54; difference -0.03), so the unconditional log ratio is flat in QCL SD. Each panel is a single-covariate regression, so its slopes match those derived above (same one-covariate model) and differ slightly from the jointly adjusted slopes of the full covariate model in Section 6. The slope difference in either panel equals the corresponding univariate log-ratio slope derived above.

This comparison highlights the key point of this relationship: the bivariate model isolates *where* a covariate-driven discrepancy comes from (one method's slope, the other's, or both), and reproduces the univariate log-ratio slope as their difference.

5.3 Why the main text uses the univariate model

For the log ratio *without* a level control, the two formulations estimate the same covariate effect exactly (previous subsection) and therefore agree in sign and magnitude. The univariate log-ratio regression is the more economical of the two: it carries a single response per unit, requires no explicit variance-covariance structure, and maps directly onto the Bland-Altman display. The main text builds on this univariate form but additionally conditions on the log mean, which shifts the covariate slopes slightly relative to the exact identity; the resulting distinction between unadjusted and level-adjusted effects is taken up in Section 6. The bivariate view in this appendix complements the main text by addressing why, and under which conditions, the two methods diverge.

5.4 Added value of the bivariate view

The bivariate model is nonetheless worth reporting as a supporting analysis, because it answers three questions the univariate form cannot, all taken up in Section 6:

1. **Mechanistic attribution** - it splits a covariate effect on the log ratio into the two method-specific slopes, identifying which method drives the divergence.

2. **Variance heterogeneity** - it allows a direct test of whether a covariate accounts for part of the difference in method precision, by comparing the method-specific error variances with and without the covariate.
3. **Method × campaign structure** - it accommodates a method × campaign interaction, i.e. whether the methods diverge to different degrees across campaigns; the univariate log ratio cannot expose this, because campaign information is largely confounded with the measurement level.

6. Bivariate model with covariates

The algebraic link of Section 5 showed that a covariate effect on the log ratio equals the difference of the two method-specific slopes. Estimating those slopes directly - by fitting the bivariate model with a method × covariate interaction - is therefore the natural way to ask *which* method responds to a given driver, and whether the covariates account for part of the variance heterogeneity. We use the covariate set of the main text - air temperature (AT), relative humidity (RH), friction velocity (USTAR), the within-exposure SD of the QCL concentration (QCL SD), and the ALPHA profile fit R^2 - each z-standardised, and fit, for both endpoints,

$$\log Y_{ki} = \beta_{0k} + \sum_j \beta_{jk}\ U_{ji} + e_{ki},$$

i.e. `log_y ~ method * (AT + RH + USTAR + QCL_SD + R2)`. The log mean is deliberately omitted: it is a linear combination of the two responses and cannot enter a model in which those responses are the outcome (Section 5). The covariate effects estimated here therefore correspond to the unadjusted univariate slopes, not to the level-adjusted ones of the main text - a distinction that matters for the concentration endpoint (below).

6.1 Method-specific covariate slopes

The method-specific slopes for both endpoints are shown in Fig. 8 of the main text; the corresponding numerical values are given in the table below. Each slope is the change in the log reading of one method per 1 SD of the covariate; the difference between the two methods is the corresponding log-ratio slope. Unlike the single-covariate illustrations of Section 5, the slopes below come from the joint model with all five covariates, so a given covariate's slope is adjusted for the others and differs slightly from its single-covariate value.

Endpoint	Covariate	β ALPHA(-IHF)	β QCL(-EC)	Difference (QCL − ALPHA)	p (interaction)
Concentration	Air temperature	-0.327	-0.135	0.192	0.066
Concentration	Relative humidity	0.011	-0.020	-0.031	0.765
Concentration	Friction velocity (USTAR)	0.003	0.047	0.044	0.657
Concentration	QCL SD	0.499	0.490	-0.009	0.932

Endpoint	Covariate	β ALPHA(-IHF)	β QCL(-EC)	Difference (QCL − ALPHA)	p (interaction)
Concentration	Profile R^2	0.024	0.130	0.105	0.282
Emission	Air temperature	-0.523	-0.109	0.414	0.002
Emission	Relative humidity	-0.152	-0.022	0.131	0.347
Emission	Friction velocity (USTAR)	0.624	0.136	-0.488	0.000
Emission	QCL SD	0.590	0.798	0.208	0.117
Emission	Profile R^2	0.648	0.144	-0.504	0.000

The two endpoints behave very differently. For **emissions** the method × covariate interaction is jointly highly significant (likelihood-ratio $p < 10^{-5}$): ALPHA-IHF responds strongly to USTAR (+0.62 per 1 SD), profile R^2 (+0.65) and air temperature (-0.52), whereas QCL-EC barely responds to any of them. Because ALPHA-IHF is more sensitive than QCL-EC to changes in these covariates, the covariate dependence of the emission ratio is localised to the ALPHA-IHF method. A mechanistic interpretation of this pattern is left to the discussion of the main text.

For **concentrations** the interaction is not significant ($p = 0.30$): the two methods respond to the covariates in much the same way. QCL SD is the clearest case - both methods rise by about 0.5 log units per 1 SD of QCL SD (ALPHA +0.50, QCL +0.49), so the unconditional difference between them is essentially zero (-0.01, $p = 0.93$). This does not contradict the main text, which reports a significant QCL SD effect on the concentration ratio after conditioning on the log mean: conditioning isolates the component of QCL SD that is not shared with the overall concentration level, and it is that component which differentiates the methods. The bivariate model addresses the unconditional question, the main-text model the level-adjusted one; both are valid, and together they locate the QCL SD effect specifically in the level-adjusted contrast.

In short: QCL SD separates the two concentration methods only after adjusting for the overall level, not on its own; and the emission ratio's covariate dependence sits almost entirely in ALPHA-IHF, whereas the two concentration methods respond to the covariates alike.

6.2 Method × campaign interaction

Whether the method discrepancy itself varies between campaigns - something the univariate log ratio cannot show, since campaign differences are largely absorbed into the measurement level - can be read from a method × campaign interaction. It improves the fit for emissions (likelihood-ratio $p = 0.004$) but not for concentrations ($p = 0.27$), so the emission discrepancy is campaign-dependent while the concentration discrepancy is stable.

6.3 Does covariate adjustment explain the variance heterogeneity?

If the covariates carried the variance heterogeneity identified in Section 4, the method-specific residual SDs - and especially their ratio - should move towards equality once the covariates are included. They do not. The comparison uses the full main-text covariate set (AT, RH, USTAR, QCL SD, profile R^2) together with all method × covariate interactions, so both common and method-specific covariate dependence are absorbed. Adding the covariates lowers both residual SDs, as any added predictor would, but the ratio of the QCL to the ALPHA SD barely changes for emissions (0.73 to 0.77) and moves the other way for concentrations (0.62 to 0.48). Covariate adjustment therefore explains, at most, a small and endpoint-specific part of the precision difference between the methods; the bulk of the heterogeneity remains and appears intrinsic to the methods rather than driven by the measured conditions. The empirical finding above rules out only that the *recorded* covariates drive the heterogeneity; treating it as an intrinsic precision difference is therefore a parsimonious modelling choice, not a conclusion the data can force (with a single observation per method-by-unit cell the underlying non-identifiability is unavoidable, as discussed in Section 7).

In short: the covariates do not close the precision gap between the methods - it appears intrinsic to them rather than driven by the measured conditions.

Table: Method-specific residual SDs without and with covariate adjustment.

Quantity	Concentration	Emission
σ (ALPHA) (without → with covariates)	1.13 → 0.97	1.54 → 1.04
σ (QCL) (without → with covariates)	0.71 → 0.47	1.12 → 0.80
ratio QCL / ALPHA (without → with covariates)	0.62 → 0.48	0.73 → 0.77

7. Limitations and open issues

7.1 Sensitivity to the two ALPHA outliers

The near-zero QCL error variance reported for the concentration endpoint (Section 4: $\sigma^2_{e,QCL} = 0.01$) is the most fragile figure in the analysis. It depends on excluding two ALPHA values from campaign 2023-III, both essentially zero (0.0002 and 0.0004 µg/m³ - below the detection limit) while QCL read 3.0 and 3.9 µg/m³ on the same occasions. These are ALPHA profile-fit failures rather than genuine low concentrations, and both exceed a Cook's distance of 1 in the Bland-Altman regression. Figure S5 shows the concentration scatter on log-log axes with both failures highlighted.

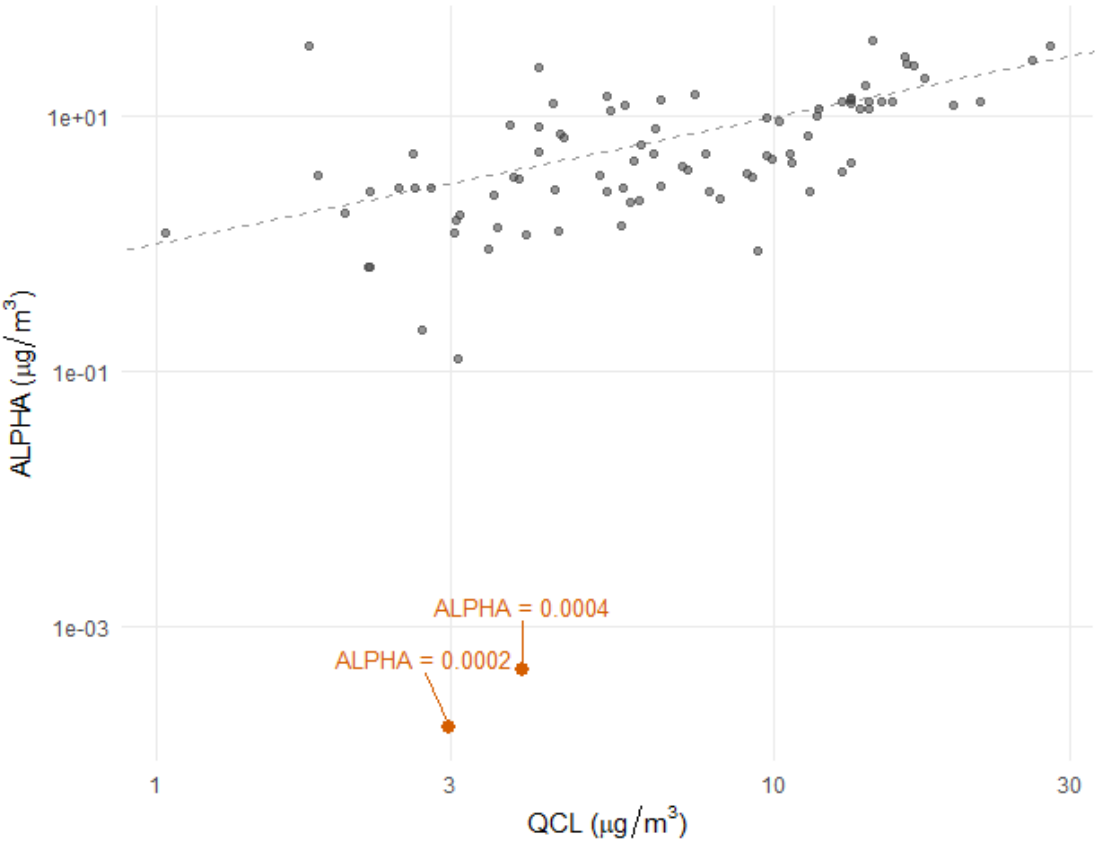

Fig. S5: Concentration scatter, $n = 88$, including the two ALPHA failures from campaign 2023-III (red). For those two occasions the ALPHA profile fit yielded essentially zero while QCL read 3-4 µg/m³ on the same air. Both points exceed a Cook's distance of 1 in the Bland-Altman regression and force the additive variance decomposition outside its admissible range (next table). Excluding them is therefore a precondition for the additive model in Section 4, not a cosmetic step.

Retaining them inflates the variance of log ALPHA roughly threefold and pushes the additive decomposition outside its admissible range (table below): the implied QCL error variance turns negative, which is mathematically impossible and means the additive model $\Sigma = \tau^2 J + \mathrm{diag}(\sigma_e^2)$ cannot hold. Removing the two failures restores a small positive value and roughly halves the Bland-Altman slope. We therefore report $\sigma^2_{e,QCL} \approx 0$ as a model-based estimate conditional on this exclusion, not as a hard measurement error variance.

Quantity	With the 2 ALPHA failures	Excluded (main analysis)
n	88	86
σ (QCL)	0.704	0.705
σ (ALPHA)	1.851	1.130
ρ (within-unit)	0.477	0.612

Quantity	With the 2 ALPHA failures	Excluded (main analysis)
τ² (shared-signal variance)	0.622	0.488
σ²_e (QCL)	-0.125	0.010
σ²_e (ALPHA)	2.806	0.789
Bland-Altman slope (d ~ log mean)	-1.13	-0.57

The negative $\sigma^2_{e,QCL}$ in the left-hand column is not an estimate to be interpreted but a signal that the additive parametrisation is inadmissible for that data set. The exclusion is thus a precondition for the additive model, and we make it explicit here rather than leave it implicit in the main-text results.

7.2 Additive versus multiplicative models

Throughout we adopt an additive model with method-specific error variances. A substantive alternative is a multiplicative method × unit interaction (Snee, 1982), in which the two methods scale the common signal differently rather than adding independent errors to it (Section 3, point (viii)). With only one paired observation per measurement occasion, however, a multiplicative interaction and plain variance heterogeneity are not separately identifiable: both produce the same fanning pattern, and no test can attribute it to one mechanism rather than the other. The additive parametrisation is therefore adopted as the parsimonious working model, not because the multiplicative mechanism has been ruled out.

7.3 A variance function for the limits of agreement

The corrected limits of agreement (Section 4) assume the log-ratio variance is constant once the variance-driven trend is removed; if it still varies with the measurement level, a variance function $\mathrm{var}(d) = f(U)$ (e.g. `gls(..., weights = varExp())`; Sadler, 2019) would let the limits' width track a covariate and so capture the random (scatter) component of agreement, which neither the main text nor the models here address. We flag this as a refinement rather than carry it out.

7.4 Selection in the emission endpoint

The emission analysis uses only the 59 observations on which both methods returned a positive flux. Of the 95 paired emission observations, in about 38% of the cases no ALPHA-IHF emission could be derived from the profiles. These missing values are profile-reconstruction failures, not values below a detection limit, so they are not naturally handled by censoring or survival analysis methods (Onofri et al., 2019). Restricting to positive pairs is a form of positive selection: it preferentially retains occasions with a well-determined ALPHA profile, and the excluded cases cluster in conditions of poor profile quality. The emission results therefore describe agreement when both methods deliver a usable flux and may not extend to the conditions under which ALPHA-IHF fails; a two-part (hurdle) model would be the appropriate way to use the full set.

References

Borga, M.: Investigating the agreement between methods of different precision, Research Square [preprint], https://doi.org/10.21203/rs.3.rs-2533522/v1, 2023.

Onofri, A., Piepho, H. P., and Kozak, M.: Analysing censored data in agricultural research: a review with examples and software tips, Ann. Appl. Biol., 174, 3–13, https://doi.org/10.1111/aab.12477, 2019.

Piepho, H. P.: Tests for equality of dispersion in bivariate samples - review and empirical comparison, J. Stat. Comput. Simul., 56, 353–372, https://doi.org/10.1080/00949659708811799, 1997.

Sadler, W. A.: Using the variance function to generalize Bland-Altman analysis, Ann. Clin. Biochem., 56, 198–203, https://doi.org/10.1177/0004563218806560, 2019.

Snee, R. D.: Nonadditivity in a two-way classification: is it interaction or nonhomogeneous variance?, J. Am. Stat. Assoc., 77, 515–519, https://www.jstor.org/stable/2287704, 1982.

Code appendix

The complete, runnable code for this document is reproduced below in document order.

```
library(tidyverse)
library(readxl)
library(nlme)      # bivariate gls with heterogeneous variances
library(emmeans)   # method-specific slopes (emtrends)
library(car)       # Type-III tests (Anova)
library(lmtest)    # Breusch-Pagan test
library(broom)     # tidy() for forest plots
library(flextable) # neutral DOCX tables
library(patchwork) # combine plots
library(ggrepel)   # non-overlapping point labels

# Manuscript colour palette (colour-blind friendly; from final_figures.R) ---
-
# Methods: blue = QCL(-EC), vermillion = ALPHA(-IHF)
method_colors <- c("QCL-EC" = "#0072B2", "ALPHA-IHF" = "#D55E00",
                   "QCL"     = "#0072B2", "ALPHA"      = "#D55E00")
# Campaigns (Okabe-Ito), one colour per year-campaign combination
kamp_colors <- c("2021-II"  = "#0072B2", "2021-III" = "#D55E00",
                 "2022-I"   = "#009E73", "2022-II"  = "#CC79A7",
                 "2022-III" = "#E69F00", "2023-I"   = "#56B4E9",
                 "2023-II"  = "#F0E442", "2023-III" = "grey20")

theme_set(theme_minimal(base_size = 11))

# Neutral table helper (matches the main analysis style: flextable + theme_va
nilla)
```

```r
ft <- function(df, size = 10) {
  df |> flextable() |> autofit() |> theme_vanilla() |>
    fontsize(size = size, part = "all")
}
# Current data version (2026-05-20); first 21 columns hold the analysis varia
bles.
# Several columns are stored as text in Excel and are coerced to numeric here
.
dat_all <- read_excel("../data/data 20260520.xlsx", range = cell_cols(1:21))
|>
  mutate(
    Kampagne         = factor(paste(year, camp, sep = "-")),
    Jahr             = factor(year),
    QCL_SD           = suppressWarnings(as.numeric(QCL_SD)),
    ALPHA            = suppressWarnings(as.numeric(ALPHA)),
    `ALPHA-IHF_flux` = suppressWarnings(as.numeric(`ALPHA-IHF_flux`)),
    RH               = suppressWarnings(as.numeric(RH)),
    R2_ALPHA_PROF    = suppressWarnings(as.numeric(R2_ALPHA_PROF)),
    VPD              = suppressWarnings(as.numeric(VPD)),
    USTAR            = suppressWarnings(as.numeric(USTAR)),
    AT               = suppressWarnings(as.numeric(AT))
  )

# Exclude the 2 ALPHA measurement failures (Cook's D >> 1, campaign 2023-III)
.
# The Limitations section quantifies why this exclusion is load-bearing for
# the additive model; dat_all keeps them for that sensitivity analysis.
dat <- dat_all |> filter(is.na(outliers) | outliers != "ALPHA")
# --- Concentration endpoint (QCL vs ALPHA): no zeros, n = 86 ---
dat_conc <- dat |>
  filter(!is.na(QCL), !is.na(ALPHA), QCL > 0, ALPHA > 0) |>
  mutate(
    log_ratio = log(QCL) - log(ALPHA),
    log_mean  = (log(QCL) + log(ALPHA)) / 2
  )

# --- Emission endpoint (QCL-EC vs ALPHA-IHF) ---
# All pairs with both methods present (incl. ALPHA-IHF = 0), n = 95
dat_flux_all <- dat |>
  filter(!is.na(`QCL-EC_flux`), !is.na(`ALPHA-IHF_flux`))

# Main analysis: positive pairs only (log scale requires > 0), n = 59
dat_pos <- dat_flux_all |>
  filter(`QCL-EC_flux` > 0, `ALPHA-IHF_flux` > 0) |>
  mutate(
    log_ratio = log(`QCL-EC_flux`) - log(`ALPHA-IHF_flux`),
    log_mean  = (log(`QCL-EC_flux`) + log(`ALPHA-IHF_flux`)) / 2
  )
# Synthetic Bland-Altman cases: systematic difference (bias) x precision.
# True level T varies across units; method 2 = T, method 1 = T (no bias),
# T + c (constant bias), or k*T (proportional bias). Unequal-precision row
```

```r
# inflates method 2's error SD. Same seed/parameters reproduce the caption va
lues.
set.seed(20260714)
ba_n <- 150; ba_c <- 2; ba_k <- 1.5; ba_s2u <- 2.5

make_ba_case <- function(bias_type, prec_type) {
  Tval <- runif(ba_n, 2, 8)
  s2   <- if (prec_type == "unequal") ba_s2u else 1.0
  X1   <- switch(bias_type, none = Tval, constant = Tval + ba_c,
                 proportional = ba_k * Tval) + rnorm(ba_n, 0, 1.0)
  X2   <- Tval + rnorm(ba_n, 0, s2)
  tibble(bias_type, prec_type, d = X1 - X2, m = (X1 + X2) / 2)
}

lab_ba <- function(df) {
  df |>
    mutate(bias_lab = factor(bias_type, levels = c("none", "constant", "propo
rtional"),
                             labels = c("No bias", "Constant bias", "Proporti
onal bias")),
           prec_lab = factor(prec_type, levels = c("equal", "unequal"),
                             labels = c("Equal precision", "Unequal precision
")))
}

ba_cases <- purrr::pmap_dfr(
  expand_grid(bias_type = c("none", "constant", "proportional"),
              prec_type = c("equal", "unequal")),
  function(bias_type, prec_type) make_ba_case(bias_type, prec_type)) |> lab_b
a()

ba_labels <- tibble(
  bias_type = c("none", "constant", "proportional", "none", "constant", "prop
ortional"),
  prec_type = c("equal", "equal", "equal", "unequal", "unequal", "unequal"),
  lab       = c("A", "B", "C", "D", "E", "F")) |> lab_ba()

ggplot(ba_cases, aes(m, d)) +
  geom_hline(yintercept = 0, colour = "grey55", linetype = "dashed") +
  geom_point(alpha = 0.40, size = 1) +
  geom_smooth(method = "lm", se = FALSE, colour = method_colors[["QCL"]],
              linewidth = 0.9, formula = y ~ x) +
  geom_text(data = ba_labels, aes(label = lab), x = -Inf, y = Inf,
            hjust = -0.5, vjust = 1.4, fontface = "bold", size = 5, inherit.a
es = FALSE) +
  facet_grid(prec_lab ~ bias_lab, scales = "free_x") +
  labs(x = "mean  m = (X₁ + X₂) / 2", y = "difference  d = X₁ − X₂") +
  theme_bw(base_size = 11) +
  theme(panel.grid.minor = element_blank(),
        strip.text = element_text(face = "bold", size = 10))
```

```r
set.seed(20260528)
n_toy   <- 200
mu_toy  <- c(5, 5)
s1_toy  <- 0.3
s2_toy  <- 0.6
rho_toy <- 0.4
Sigma_toy <- matrix(c(s1_toy^2, rho_toy*s1_toy*s2_toy,
                      rho_toy*s1_toy*s2_toy, s2_toy^2), 2, 2)
X_toy <- MASS::mvrnorm(n_toy, mu = mu_toy, Sigma = Sigma_toy)
toy <- tibble(x1 = X_toy[, 1], x2 = X_toy[, 2]) |>
  mutate(d = x1 - x2,
         m = (x1 + x2) / 2)
a_toy <- (s2_toy^2 - rho_toy*s1_toy*s2_toy) /
         (s1_toy^2 - 2*rho_toy*s1_toy*s2_toy + s2_toy^2)
toy <- toy |> mutate(w = a_toy * x1 + (1 - a_toy) * x2)

toy_theme <- theme_bw(base_size = 11) +
  theme(panel.grid.minor = element_blank(),
        plot.title = element_text(face = "bold", size = 11))
lim_toy <- range(c(toy$x1, toy$x2))

p_toy_A <- ggplot(toy, aes(x1, x2)) +
  geom_abline(slope = 1, intercept = 0, colour = "grey55", linetype = "dashed
") +
  geom_point(alpha = 0.40, size = 1) +
  coord_equal(xlim = lim_toy, ylim = lim_toy) +
  labs(title = "G: Scatter",
       x = expression(X[1]~"(more precise)"),
       y = expression(X[2]~"(less precise)")) +
  toy_theme

p_toy_B <- ggplot(toy, aes(m, d)) +
  geom_hline(yintercept = 0, colour = "grey55", linetype = "dashed") +
  geom_point(alpha = 0.40, size = 1) +
  geom_smooth(method = "lm", se = FALSE, colour = method_colors[["QCL"]],
              linewidth = 0.9, formula = y ~ x) +
  labs(title = "H: Classical BAP (d vs m)",
       x = expression(m == (X[1] + X[2])/2),
       y = expression(d == X[1] - X[2])) +
  toy_theme

p_toy_C <- ggplot(toy, aes(w, d)) +
  geom_hline(yintercept = 0, colour = "grey55", linetype = "dashed") +
  geom_point(alpha = 0.40, size = 1) +
  geom_smooth(method = "lm", se = FALSE, colour = method_colors[["QCL"]],
              linewidth = 0.9, formula = y ~ x) +
  labs(title = "I: Corrected BAP (d vs w)",
       x = expression(w == a*X[1] + (1 - a)*X[2]),
       y = expression(d == X[1] - X[2])) +
  toy_theme
```

```r
wrap_plots(p_toy_A, p_toy_B, p_toy_C, ncol = 3)
mk_col <- function(f) c(
  sprintf("%.3f", f$sigma1), sprintf("%.3f", f$sigma2), sprintf("%.3f", f$rho
),
  sprintf("%.3f", f$a), sprintf("%.3f", f$tau2),
  sprintf("%.3f", f$se1), sprintf("%.3f", f$se2), sprintf("%.1f", f$dAIC))

tibble(
  Quantity = c("σ (QCL)", "σ (ALPHA)", "ρ (within-unit)",
               "a (weight on QCL)", "τ² (shared-signal variance)",
               "σ²_e (QCL)", "σ²_e (ALPHA)",
               "ΔAIC (heterogeneous vs. homogeneous model)"),
  `Meaning / interpretation` = c(
    "Total observed standard deviation of QCL measurements, including shared
signal and method-specific error",
    "Total observed standard deviation of ALPHA measurements, including share
d signal and method-specific error",
    "Within-unit correlation between paired QCL and ALPHA measurements",
    "Relative weight of the QCL measurement in the precision-weighted mean",
    "Variance component representing the shared true signal between both meth
ods",
    "Estimated method-specific error variance of QCL measurements",
    "Estimated method-specific error variance of ALPHA measurements",
    "AIC improvement of the heterogeneous-variance model over the homogeneous
one (larger = heterogeneous model clearly preferred)"),
  `Concentration (n = 86)` = mk_col(fit_conc),
  `Emission (n = 59)`      = mk_col(fit_em)
) |> ft()
# Corrected Bland-Altman plot for one endpoint (matches final_figures.R fig 3
/7).
corrected_ba_plot <- function(d, ratio_lab, x_lab, y_lab, tag, legend = "none
") {
  b <- mean(d$log_ratio); s <- sd(d$log_ratio); lo <- b - 1.96 * s; hi <- b +
1.96 * s
  ggplot(d, aes(x = w, y = log_ratio, color = Kampagne)) +
    geom_hline(yintercept = 0, color = "grey80", linewidth = 0.6) +
    geom_hline(yintercept = b, linetype = "dashed", linewidth = 0.9, color =
"black") +
    geom_hline(yintercept = c(lo, hi), linetype = "dotted", color = "grey40",
linewidth = 0.8) +
    geom_point(size = 2.2, alpha = 0.8) +
    annotate("text", x = quantile(d$w, 0.98), y = b,
             label = paste0("Bias = ", round(b, 2)), hjust = 1, vjust = -0.6,
size = 3.5) +
    annotate("text", x = quantile(d$w, 0.98), y = hi,
             label = paste0("+1.96 SD = ", round(hi, 2)), hjust = 1, vjust =
-0.6, size = 3.5, color = "grey30") +
    annotate("text", x = quantile(d$w, 0.98), y = lo,
             label = paste0("−1.96 SD = ", round(lo, 2)), hjust = 1, vjust =
1.4, size = 3.5, color = "grey30") +
```

```
    annotate("text", x = -Inf, y = Inf, label = tag, hjust = -0.5, vjust = 1.
5, size = 5, fontface = "bold") +
    scale_color_manual(values = kamp_colors) +
    scale_y_continuous(sec.axis = sec_axis(~ exp(.), name = ratio_lab,
                       breaks = c(0.05, 0.1, 0.25, 0.5, 1, 2, 5, 10))) +
    labs(x = x_lab, y = y_lab, color = "Campaign") +
    theme_minimal(base_size = 12) +
    theme(legend.position = legend, panel.grid.minor = element_blank(),
          panel.grid.major = element_line(linewidth = 0.3, color = "grey85"))
}

dat_conc <- dat_conc |> mutate(w = fit_conc$a * log(QCL) + (1 - fit_conc$a) *
log(ALPHA))
dat_pos  <- dat_pos  |> mutate(w = fit_em$a * log(`QCL-EC_flux`) + (1 - fit_e
m$a) * log(`ALPHA-IHF_flux`))

f_conc <- corrected_ba_plot(dat_conc, "ratio QCL / ALPHA",
                            "Weighted mean concentration", "Log ratio log(QCL
/ ALPHA)", "a)")
f_em   <- corrected_ba_plot(dat_pos, "ratio QCL-EC / ALPHA-IHF",
                            "Weighted mean emission", "Log ratio log(QCL-EC /
ALPHA-IHF)", "b)",
                            legend = "bottom")
wrap_plots(f_conc, f_em, ncol = 2)
# Emission endpoint, covariate USTAR (the strongest emission driver). The
# bivariate model allows method-specific USTAR slopes; the univariate models
# regress the log ratio on USTAR with and without the log-mean control. Clean
# method labels (QCL_EC / ALPHA) keep the interaction coefficient name tidy.
db <- dat_pos |>
  transmute(
    id         = row_number(),
    QCL_EC     = `QCL-EC_flux`,
    ALPHA      = `ALPHA-IHF_flux`,
    log_ratio,
    USTAR_z    = as.numeric(scale(USTAR)),
    log_mean_z = as.numeric(scale(log_mean))
  )

dl <- db |>
  pivot_longer(c(QCL_EC, ALPHA), names_to = "method", values_to = "y") |>
  mutate(method = factor(method, levels = c("ALPHA", "QCL_EC")),  # ALPHA = r
eference
         log_y  = log(y))

m_int <- gls(log_y ~ method * USTAR_z,
             weights     = varIdent(form = ~1 | method),
             correlation = corCompSymm(form = ~1 | id),
             data = dl, method = "REML")

# Coefficients: USTAR_z = ALPHA-IHF slope; methodQCL_EC:USTAR_z = QCL-EC minu
```

```
s
# ALPHA-IHF slope (the linking quantity).
cf      <- coef(m_int)
b_alpha <- unname(cf["USTAR_z"])
b_diff  <- unname(cf["methodQCL_EC:USTAR_z"])

# Univariate log-ratio slopes, with and without the log-mean control
s_raw <- unname(coef(lm(log_ratio ~ USTAR_z, data = db))["USTAR_z"])
s_lm  <- unname(coef(lm(log_ratio ~ log_mean_z + USTAR_z, data = db))["USTAR_
z"])

tibble(
  Quantity = c("Bivariate: USTAR slope of ALPHA-IHF",
               "Bivariate: USTAR slope of QCL-EC",
               "Bivariate slope difference (QCL-EC − ALPHA-IHF)",
               "Univariate: log ratio ~ USTAR (no log mean)",
               "Univariate: log ratio ~ log mean + USTAR (main-text control s
tructure, USTAR alone)"),
  `Slope (per 1 SD)` = sprintf("%.3f", c(b_alpha, b_alpha + b_diff, b_diff, s
_raw, s_lm))
) |> ft()
# Long-form data for both endpoints with their representative covariates.
# Reuses dat_conc / dat_pos prepared in `prep-endpoints`.
em_long <- dat_pos |>
  transmute(QCL = `QCL-EC_flux`, ALPHA = `ALPHA-IHF_flux`,
            x = as.numeric(scale(USTAR))) |>
  pivot_longer(c(QCL, ALPHA), names_to = "method", values_to = "y") |>
  mutate(log_y = log(y))

co_long <- dat_conc |>
  transmute(QCL = QCL, ALPHA = ALPHA,
            x = as.numeric(scale(QCL_SD))) |>
  pivot_longer(c(QCL, ALPHA), names_to = "method", values_to = "y") |>
  mutate(log_y = log(y))

# One labelled-panel helper so both endpoints share styling.
panel <- function(d, xlab, title, sub, labs_methods) {
  d <- mutate(d, method = factor(method, levels = c("QCL", "ALPHA")))
  ggplot(d, aes(x = x, y = log_y, color = method)) +
    geom_point(alpha = 0.55, size = 1.6) +
    geom_smooth(method = "lm", se = FALSE, linewidth = 0.9, formula = y ~ x)
+
    scale_color_manual(values = c(QCL = method_colors[["QCL"]],
                                  ALPHA = method_colors[["ALPHA"]]),
                       labels = labs_methods, name = NULL) +
    labs(x = xlab, y = "log(reading)", title = title, subtitle = sub) +
    theme_minimal(base_size = 11) +
    theme(legend.position = "bottom",
          plot.subtitle = element_text(size = 9, color = "grey30"),
          panel.grid.minor = element_blank())
}
```

```r
p_em <- panel(em_long, "USTAR (z-standardised)",
              "Emissions: divergent lines",
              "Slope difference QCL-EC - ALPHA-IHF = -0.53 per 1 SD",
              c("QCL-EC", "ALPHA-IHF"))

p_co <- panel(co_long, "QCL SD (z-standardised)",
              "Concentrations: near-parallel lines",
              "Slope difference QCL - ALPHA = -0.03 per 1 SD",
              c("QCL", "ALPHA"))

wrap_plots(p_em, p_co, ncol = 2)
covars  <- c("AT", "RH", "USTAR", "QCL_SD", "R2_ALPHA_PROF")
cov_lab <- c(AT = "Air temperature", RH = "Relative humidity",
             USTAR = "Friction velocity (USTAR)", QCL_SD = "QCL SD",
             R2_ALPHA_PROF = "Profile R²")

# Method-specific residual SDs from a gls fit (reference level recovered safe
ly).
method_sd <- function(m, lv) {
  base <- m$sigma; mult <- coef(m$modelStruct$varStruct, unconstrained = FALS
E)
  ref  <- setdiff(lv, names(mult)); s <- c(setNames(base, ref), base * mult);
s[lv]
}

# Fit the bivariate covariate model for one endpoint and return: method-speci
fic
# slopes (with 95% CI from vcov), the joint method:covariate and method:campa
ign
# LRT p-values, and the residual SDs with and without covariates.
fit_cov <- function(d, ycols, endpoint_label) {
  dd <- d |>
    transmute(QCL = .data[[ycols[1]]], ALPHA = .data[[ycols[2]]], Kampagne,
              across(all_of(covars))) |>
    mutate(across(all_of(covars), ~ as.numeric(scale(.)), .names = "{.col}_z"
),
           id = row_number())
  zc   <- paste0(covars, "_z")
  long <- dd |> select(id, Kampagne, QCL, ALPHA, all_of(zc)) |>
    pivot_longer(c(QCL, ALPHA), names_to = "method", values_to = "y") |>
    mutate(method = factor(method, levels = c("ALPHA", "QCL")), log_y = log(y
))

  vc <- list(weights = varIdent(form = ~1 | method),
             correlation = corCompSymm(form = ~1 | id))
  f_add <- as.formula(paste("log_y ~ method +", paste(zc, collapse = " + ")))
  f_int <- as.formula(paste("log_y ~ method * (", paste(zc, collapse = " + ")
, ")"))
  m0    <- do.call(gls, c(list(log_y ~ method, data = long, method = "REML"),
```

```
vc))
  m_add <- do.call(gls, c(list(f_add, data = long, method = "REML"), vc))
  m_int <- do.call(gls, c(list(f_int, data = long, method = "REML"), vc))

  cf <- coef(m_int); V <- vcov(m_int); tt <- summary(m_int)$tTable
  slopes <- map_dfr(covars, function(v) {
    bz <- paste0(v, "_z"); ix <- paste0("methodQCL:", bz)
    est <- c(unname(cf[bz]), unname(cf[bz] + cf[ix]))
    se  <- c(sqrt(V[bz, bz]), sqrt(V[bz, bz] + V[ix, ix] + 2 * V[bz, ix]))
    tibble(endpoint = endpoint_label, covariate = unname(cov_lab[v]),
           method = c("ALPHA(-IHF)", "QCL(-EC)"), slope = est, se = se,
           diff = unname(cf[ix]), p = unname(tt[ix, "p-value"]))
  })

  int_lrt  <- anova(update(m_add, method = "ML"), update(m_int, method = "ML"
))
  mc  <- do.call(gls, c(list(log_y ~ method + Kampagne, data = long, method =
"ML"), vc))
  mcX <- do.call(gls, c(list(log_y ~ method * Kampagne, data = long, method =
"ML"), vc))
  camp_lrt <- anova(mc, mcX)

  list(slopes = slopes,
       int_p  = int_lrt$`p-value`[2],  camp_p = camp_lrt$`p-value`[2],
       sd0 = method_sd(m0, c("ALPHA", "QCL")),
       sdi = method_sd(m_int, c("ALPHA", "QCL")))
}

cov_conc <- fit_cov(dat_conc, c("QCL", "ALPHA"), "Concentration")
cov_em   <- fit_cov(dat_pos,  c("QCL-EC_flux", "ALPHA-IHF_flux"), "Emission")
bind_rows(cov_conc$slopes, cov_em$slopes) |>
  pivot_wider(id_cols = c(endpoint, covariate, diff, p),
              names_from = method, values_from = slope) |>
  transmute(Endpoint = endpoint, Covariate = covariate,
            `β ALPHA(-IHF)` = sprintf("%.3f", `ALPHA(-IHF)`),
            `β QCL(-EC)`    = sprintf("%.3f", `QCL(-EC)`),
            `Difference (QCL − ALPHA)` = sprintf("%.3f", diff),
            `p (interaction)` = sprintf("%.3f", p)) |>
  ft(size = 9)
sd_row <- function(label, conc, em) tibble(
  Quantity = label,
  `Concentration` = sprintf("%.2f → %.2f", conc[1], conc[2]),
  `Emission`      = sprintf("%.2f → %.2f", em[1], em[2]))

bind_rows(
  sd_row("σ (ALPHA) (without → with covariates)",
         c(cov_conc$sd0["ALPHA"], cov_conc$sdi["ALPHA"]),
         c(cov_em$sd0["ALPHA"],   cov_em$sdi["ALPHA"])),
  sd_row("σ (QCL) (without → with covariates)",
         c(cov_conc$sd0["QCL"], cov_conc$sdi["QCL"]),
```

```
        c(cov_em$sd0["QCL"],   cov_em$sdi["QCL"])),
  sd_row("ratio QCL / ALPHA (without → with covariates)",
        c(cov_conc$sd0["QCL"] / cov_conc$sd0["ALPHA"],
          cov_conc$sdi["QCL"] / cov_conc$sdi["ALPHA"]),
        c(cov_em$sd0["QCL"] / cov_em$sd0["ALPHA"],
          cov_em$sdi["QCL"] / cov_em$sdi["ALPHA"]))
) |> ft()
dat_conc_outliers <- dat_all |>
  filter(!is.na(QCL), !is.na(ALPHA), QCL > 0, ALPHA > 0) |>
  mutate(is_outlier = !is.na(outliers) & outliers == "ALPHA",
        lbl = ifelse(is_outlier, sprintf("ALPHA = %.4f", ALPHA), NA_characte
r_))

ggplot(dat_conc_outliers, aes(QCL, ALPHA)) +
  geom_abline(slope = 1, intercept = 0, colour = "grey60", linetype = "dashed
") +
  geom_point(data = filter(dat_conc_outliers, !is_outlier),
             alpha = 0.55, size = 1.6, colour = "grey25") +
  geom_point(data = filter(dat_conc_outliers, is_outlier),
             colour = "#D55E00", size = 2.6) +
  geom_text_repel(data = filter(dat_conc_outliers, is_outlier),
                  aes(label = lbl), colour = "#D55E00", size = 3.3,
                  nudge_y = 0.5, segment.colour = "#D55E00",
                  min.segment.length = 0) +
  scale_x_log10() +
  scale_y_log10() +
  labs(x = expression(QCL~"("*mu*g/m^3*")"),
       y = expression(ALPHA~"("*mu*g/m^3*")")) +
  theme_minimal(base_size = 11) +
  theme(panel.grid.minor = element_blank())
# Concentration endpoint with the 2 ALPHA failures retained (n = 88). fit_con
c
# (n = 86, failures excluded) was computed above (bivariate model, Section 3)
.
dat_conc_with <- dat_all |>
  filter(!is.na(QCL), !is.na(ALPHA), QCL > 0, ALPHA > 0) |>
  mutate(log_ratio = log(QCL) - log(ALPHA),
        log_mean  = (log(QCL) + log(ALPHA)) / 2)
fit_with <- fit_bivariate(dat_conc_with, c("QCL", "ALPHA"))

ba_slope <- function(d) unname(coef(lm(log_ratio ~ log_mean, data = d))["log_
mean"])
sens_col <- function(f, d) c(
  nrow(d), sprintf("%.3f", f$sigma1), sprintf("%.3f", f$sigma2),
  sprintf("%.3f", f$rho), sprintf("%.3f", f$tau2),
  sprintf("%.3f", f$se1), sprintf("%.3f", f$se2), sprintf("%.2f", ba_slope(d)
))

tibble(
  Quantity = c("n", "σ (QCL)", "σ (ALPHA)", "ρ (within-unit)",
               "τ² (shared-signal variance)", "σ²_e (QCL)", "σ²_e (ALPHA)",
```

```
              "Bland-Altman slope (d ~ log mean)"),
  `With the 2 ALPHA failures` = sens_col(fit_with, dat_conc_with),
  `Excluded (main analysis)`  = sens_col(fit_conc, dat_conc)
) |> ft()
```