

# Comparison of straight line curve fit approaches for determining parameter variances and covariances

Vishal Ramnath\*

Department of Mechanical and Industrial Engineering, University of South Africa, Private Bag X6, Florida 1710, South Africa

Received: 24 August 2020 / Accepted: 23 October 2020

**Abstract.** Pressure balances are known to have a linear straight line equation of the form  $y = ax + b$  that relates the applied pressure  $x$  to the effective area  $y$ , and recent work has investigated the use of Ordinary Least Squares (OLS), Weighted Least Squares (WLS), and Generalized Least Squares (GLS) regression schemes in order to quantify the expected values of the zero-pressure area  $A_0 = b$  and distortion coefficient  $\lambda = a/b$  in pressure balance models of the form  $y = A_0(1 + \lambda x)$ . The limitations with conventional OLS, WLS and GLS approaches is that whilst they may be used to quantify the uncertainties  $u(a)$  and  $u(b)$  and the covariance  $cov(a, b)$ , it is technically challenging to analytically quantify the covariance term  $cov(A_0, \lambda)$  without additional Monte Carlo simulations. In this paper, we revisit an earlier Weighted Total Least Squares with Correlation (WTLSC) algorithm to determine the variances  $u^2(a)$  and  $u^2(b)$  along with the covariance  $cov(a, b)$ , and develop a simple analytical approach to directly infer the corresponding covariance  $cov(A_0, \lambda)$  for pressure metrology uncertainty analysis work. Results are compared to OLS, WLS and GLS approaches and indicate that the WTLSC approach may be preferable as it avoids the need for Monte Carlo simulations and additional numerical post-processing to fit and quantify the covariance term, and is thus simpler and more suitable for industrial metrology pressure calibration laboratories. Novel aspects is that a Gnu Octave/Matlab program for easily implementing the WTLSC algorithm to calculate parameter expected values, variances and covariances is also supplied and reported.

**Keywords:** Ordinary least squares (OLS) / weighted least squares (WLS) / generalized least squares (GLS) / weighted total least squares with correlation (WTLSC) / pressure measurement

## 1 Introduction

### 1.1 Research motivation

The use of least squares based statistical regression analysis is a widely used tool to analyse experimental data when estimating and fitting model parameter values and uncertainties. In the particular case of straight line equations of the form  $y = ax + b$  known input data  $x_i$  and output data  $y_i$  for data-points  $i = 1, \dots, n$  inclusive of possible available uncertainty information are used in the regression analysis where standard uncertainty analysis theory from the GUM [1] specifies the model uncertainty as

$$y = ax + b \quad (1)$$

$$\begin{aligned} u^2(y) = & \left(\frac{\partial y}{\partial x}\right)^2 u^2(x) + \left(\frac{\partial y}{\partial a}\right)^2 u^2(a) \\ & + \left(\frac{\partial y}{\partial b}\right)^2 u^2(b) + 2\frac{\partial y}{\partial a}\frac{\partial y}{\partial b} cov(a, b) \\ & + 2\frac{\partial y}{\partial x}\frac{\partial y}{\partial a} cov(x, a) + 2\frac{\partial y}{\partial x}\frac{\partial y}{\partial b} cov(x, b) \quad (2) \end{aligned}$$

In the above formula since  $x$  is a statistically independent random variable it follows that  $cov(x, a) = 0$ ,  $cov(x, b) = 0$ , and in general  $cov(a, b) \neq 0$ . As a result, the probability density function (PDF) of the output  $g_y(\eta_y)$  may be determined where  $\eta_y$  is a univariate random variable of the model output  $y$ ,  $g_x(\xi_x)$  is a univariate random variable of the model input  $x$ , and  $g_{a,b}(\xi_a, \xi_b)$  is a bivariate random variable of the joint PDF of the model parameters  $a$  and  $b$  that incorporates the coupling effect

\* Corresponding author: [ramnav@unisa.ac.za](mailto:ramnav@unisa.ac.za)

between random variables  $\xi_a$  and  $\xi_b$  for the model parameters  $a$  and  $b$  that are modelled in terms of  $cov(a, b)$ .

The research problem that occurs when OLS, WLS and GLS regressions are used to estimate the parameters  $a$  and  $b$  to varying degrees of accuracies, is that additional Monte Carlo simulations must then be performed in order to determine the variances  $u^2(a)$  and  $u^2(b)$  along with the covariance  $cov(a, b)$ , and that this information must then be further processed in order to infer  $u^2(A_0)$ ,  $u^2(\lambda)$  and  $cov(A_0, \lambda)$ . Consequently, in the absence of covariance information the model uncertainty will be systematically under-estimated if the covariance is unknown or unavailable and only the variances of the model parameters are utilized in uncertainty calculations.

## 1.2 Research focus

This paper focuses on demonstrating the conceptual ease and functionality of the WTLSC approach and how it is able to incorporate measurement variance information  $u^2(x_i)$  and  $u^2(y_i)$  along with covariance information  $cov(x_i, y_i)$  of the data when available, in order to determine both the variances  $u^2(a)$  and  $u^2(b)$  along with the covariance information  $cov(a, b)$  of the model parameters for statistical completeness. A new simple analytical calculation approach to directly infer both the variances  $u^2(A_0)$  and  $u^2(\lambda)$  as well as covariance term  $cov(A_0, \lambda)$  from knowledge of  $u^2(a)$ ,  $u^2(b)$  and  $cov(a, b)$  as obtained from the WTLSC algorithm implemented with a Gnu Octave/Matlab program is presented that does not rely on additional Monte Carlo simulations or statistical post-processing.

## 2 Literature review

### 2.1 Physical theory of pressure balances

Pressure balance theory from Dadson [2] for a cross-floated pressure balance in equilibrium may be used to define the applied pressure  $P$  in terms of a generalized force  $F$  and an effective area  $S$  as  $P = \frac{F}{S}$  where  $F$  is a generalized force term to account for buoyancy effects, and  $S$  is a cross-sectional effective area that accounts for pressure induced deformation and temperature expansion/contraction effects such that  $F = m \left(1 - \frac{\rho_a}{\rho_m}\right) g$  where  $S = A_0(1 + \lambda P)\phi$ ,  $A_0$  is a effective area when extrapolated to a zero pressure,  $\lambda$  is a distortion coefficient based on linear elasticity theory, and  $\phi$  is an appropriate temperature compensation function that adjusts the area to an appropriate reference temperature.

In general since there is a pressure dependence within the effective area term  $S$  the equation  $P = \frac{F}{S}$  is usually rearranged as a homogeneous equation  $h = \frac{F}{S} - P$  and then solved for specified inputs to obtain a value of  $P$  such that  $h=0$ . In the event that the pressure  $P$  is known and the area  $S$  must be determined the equation  $P = \frac{F}{S}$  is rearranged as  $S = \frac{F}{P}$  and again rearranged as another homogeneous equation  $h = \frac{F}{P} - S$  in order to determine the value of  $S$  that solves the homogeneous equation for specified inputs.

In the above equation  $m$  is the mass on top of the piston,  $\rho_a$  is the ambient fluid density in contact with the piston to account for buoyancy forces acting on the piston,  $\rho_m$  is the mass density,  $g$  is the gravitational acceleration constant,  $A_0$  is the zero-pressure area,  $\lambda$  is the distortion coefficient,  $P$  is the applied pressure, and  $\phi = 1 + \alpha(t - t_0)$  is a temperature compensation function to adjust the cross-sectional area to a reference temperature  $t_0$  where  $\alpha = \alpha_p + \alpha_c$  is the linear thermal expansion coefficient of the combination of the piston and cylinder where  $\alpha_p$  and  $\alpha_c$  are the corresponding linear thermal expansion coefficients of the piston and cylinder respectively.

When the above system of equations is implemented the equation for the cross-floated area  $S$  may then be simply modelled as

$$S = A_0(1 + \lambda P) \quad (3)$$

In the above equation the inputs are the expected values and standard uncertainties of the applied pressures ( $\mu(P_i) \pm u(P_i)$ ), the cross-floated areas ( $\mu(S_i) \pm u(S_i)$ ), and estimates of the covariances  $cov(P_i, S_i)$  for a sequence of measurements  $i=1, \dots, n$  where  $n$  is the number of measurements, whilst the unknowns are considered to be the expected values of the zero-pressure area  $A_0$ , the distortion coefficient  $\lambda$ , and the covariance  $cov(A_0, \lambda)$ . This modelling approach is equivalent to fitting a straight line of the form  $y = ax + b$  from experimental data, and utilizing a least squares regression to quantify the values and uncertainties of the parameters  $a$  and  $b$ .

### 2.2 Historical and mathematical context

The fitting of straight lines is a common numerical task to fields such as science, engineering, and economics that require experimental or computational data to be modelled and fitted. Usually a least-squares algorithm is implemented to minimize a merit function as discussed by Saunders [3] where in general a non-linear calibration equation  $y = y(x; a_1, \dots, a_N)$  is desired to be fitted from measurement pairs of controlled values  $x_i$  and observed values  $y_i$ . In the case where the number of measurements is  $M$  and the number of parameters to determine is  $N$  then a least squares solution is mathematically essential in the general case where  $M > N$ . For this situation a merit function is constructed as

$$\chi^2 = \sum_{i=1}^M w_i [y_i - y(x_i | a_1, \dots, a_N)]^2 \quad (4)$$

and optimized in terms of the parameters  $a_1, \dots, a_N$  by simultaneously solving  $\partial \chi^2 / \partial a_i = 0$  for  $i = 1, \dots, N$  where  $w_i$  is a suitable and appropriate weighting function to account for the underlying measurement accuracy and available information.

The simplest case involves an optimization where all of the data points have an assumed constant uncertainty/accuracy such that  $u(x_i) = u(y_i) = \sigma$  for  $i = 1, \dots, M$  measurements, which corresponds to the situation where no detailed or specific measurement uncertainty

information is available, and as a result the optimization of the merit function does not require any specific knowledge of  $\sigma$  as this can be conveniently factored out when optimizing  $\chi^2$ . This particular problem thus corresponds to the well known historical total least squares problem as investigated by Van Huffel and Vandervalle [4].

At a next conceptual level the problem of fitting a straight line with errors in both coordinates was originally formulated by York [5] and subsequently involved either attempts to directly solve the original problem posed by York or by various approximation methods in order to infer the variances of the fitting parameters by researchers working in the area of regression analysis of experimental measurements. For this problem the approach of Press et al. [6] has achieved the most notable recognition amongst many non-mathematicians for the fitting of a straight line where both  $x$  and  $y$  coordinates possess some underlying known statistical uncertainty.

In this particular case the model  $y(x) = a + bx$  involves a  $\chi^2$  merit function optimization of the form

$$\min \chi^2(a, b) = \sum_{i=1}^n \frac{(y_i - a - bx_i)^2}{(u(y_i))^2 + b^2(u(x_i))^2} \quad (5)$$

where  $u(x_i)$  and  $u(y_i)$  are the standard deviations of the  $x_i$  and  $y_i$  data points that are allowed to vary. Although conceptually straightforward the main challenge with directly solving this problem is that the presence of the parameter  $b$  in the denominator renders the partial derivative equation  $\partial\chi^2/\partial b = 0$  a non-linear equation, and as a result this problem is considerably more difficult to solve than the earlier total least squares problem which is amenable to an analytical solution.

Consequently, the optimization problem for straight lines with errors in both coordinates is typically solved either with classical optimization techniques such as the Levenberg-Marquardt method or with the Monte Carlo bootstrap technique as outlined by Press et al. [6], and as a result this solution approach is not generally directly accessible for many metrologists working in industrial laboratories due to the mathematical complexity of the method.

It should be noted that a conventional numerical solution of this non-linear optimization problem using either classical techniques such as Newton's, steepest descent and homotopy methods as discussed by Burden and Faires [7] or newer methods such as the particle swarm optimization (PSO) approach would still suffer from two limitations, namely that it would firstly not be able to compute the parameter variances, and secondly that it would not be able to provide any insights into the parameter variances and covariances.

The statistical effects from the choice of OLS, WLS and GLS regressions was subsequently investigated by Duer et al. [8] who determined that whilst the specific choice of OLS, WLS or GLS regressions did not generally produce inconsistent estimates for the intercept  $b$  and slope  $a$  parameter values, that the different regression schemes nevertheless did in fact produce considerable variations in the corresponding estimates of the uncertainties  $u(a)$

and  $u(b)$  of the parameters  $a$  and  $b$ . As a result, Duer et al. concluded that a simplified OLS regression could only be safely utilized in the very specific and special case of where the uncertainties  $u(x_k)$  in the  $x_k$  data points was negligible.

In the particular case of scientific metrology work an inconsistent or incorrect estimate of a parameter value and variance/covariance could then have a catastrophic effect on high level laboratory inter-comparison work or in safety critical experimental measurements as parameter uncertainties of equipment and instruments are typically utilized in many quality engineering systems to specify tolerances and safety limits for equipment and personnel.

Due to these specific limitations particularly for scientific measurement work at national laboratories the total least squares problem was subsequently revisited by Krystek and Anton [9] who simplified the original two-dimensional problem into an equivalent one-dimensional optimization which is amenable with a set of simplified mathematical tools. This initial approach was then later generalized to solve the full problem of errors in both coordinates with correlation effects in a later paper by Krystek and Anton [10]. This approach is now termed as a weighted total least squares with correlation (WTLSC) analysis approach and it makes an assumption that the covariance matrix  $\mathbf{U}_k$  for individual experimental data points for  $k = 1, \dots, n$  are available prior to optimizing the merit function which is explicitly specified as

$$\begin{aligned} \chi^2 &= \sum_{k=1}^n \begin{pmatrix} X_k - x_k \\ Y_k - y_k \end{pmatrix} \mathbf{U}_k^{-1} \begin{pmatrix} X_k - x_k \\ Y_k - y_k \end{pmatrix} \\ &= \sum_{k=1}^n \frac{1}{\rho_k^2} \left[ \frac{(x_k - X_k)^2}{u_{x,k}^2} \right. \\ &\quad \left. - 2\rho_k \frac{(x_k - X_k)(y_k - Y_k)}{u_{x,k}u_{y,k}} + \frac{(y_k - Y_k)^2}{u_{y,k}^2} \right] \quad (6) \end{aligned}$$

Part of the reasoning as to why the WTLSC approach does not appear to have been more widely adopted is that in the specific area of scientific metrology that the calculation of the covariance matrix is extremely mathematically complicated when using standard calculus based approaches, and is therefore only tractable from a full Monte Carlo pure numerical simulation approach as discussed by Harris [11]. As a result, if the covariance matrix is unavailable then the WTLSC approach becomes problematic in which case the covariance must either be calculated or estimated, or alternately modified by setting the correlations to zero such that the algorithm implementation then simplifies to the earlier weighted total least squares WLS algorithm.

Whilst the absence of a covariance matrix may be viewed as technical limitation when implementing a GLS regression where the standard statistical formulation comprises of determining the value of the regression parameter  $\beta$  which optimizes the residuals  $\varepsilon$  in a linear

regression model  $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ , it has subsequently been determined by researchers that this is not an insurmountable challenge as mechanisms exist to infer a likely covariance matrix. For this problem the covariance matrix  $\text{Cov}[\boldsymbol{\varepsilon}|\mathbf{X}] = \boldsymbol{\Omega}$ , is known to take the general form

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T \boldsymbol{\Omega}^{-1} \mathbf{X})^{-1} \mathbf{X}^T \boldsymbol{\Omega}^{-1} \mathbf{y} \quad (7)$$

In order to account for the unavailability of the actual covariance matrix  $\boldsymbol{\Omega}$  which may be unknown, or impractical to estimate, one may nevertheless get a consistent estimate of  $\boldsymbol{\Omega}$  usually written as  $\hat{\boldsymbol{\Omega}}$  that is termed the feasible generalized least squares (FGLS) estimator using either iteration based commercial routines with software such as Matlab [12], or with newer support vector machines (SVMs) approaches that are currently under development such as recently reported results from Miller and Startz [13].

### 2.3 Recent statistical regression developments

Recently work reported in the literature by Wuethrich and Souiyam [14] building on earlier work by Otal and Yardin [15] has investigated the use of Monte Carlo based simulations for Ordinary Least Squares (OLS), Weighted Least Squares (WLS) and Generalized Least Squares (GLS) regression schemes in order to determine the parameter values of a pressure balance model of the form  $S = bP + a$ , where  $P$  is an input of applied pressure,  $S$  is an output of cross-floated area, and  $a$  and  $b$  are model parameters whose values and uncertainties must be determined. Conventionally the pressure balance model has the form  $S = A_0(1 + \lambda P)$  where  $A_0$  is a zero-pressure area and  $\lambda$  is a distortion coefficient and the regression analysis fits the values for  $a = A_0$  and the product  $b = A_0\lambda$  for  $i = 1, \dots, n$  data-points such that

$$\begin{bmatrix} 1 & P_1 \\ \vdots & \vdots \\ 1 & P_n \end{bmatrix} \begin{bmatrix} A_0 \\ b \end{bmatrix} = \begin{bmatrix} S_1 \\ \vdots \\ S_n \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad (8)$$

$$\mathbf{X}\mathbf{C} = \mathbf{Y} + \boldsymbol{\varepsilon} \quad (9)$$

For the above regression scheme the distortion coefficient is recovered as  $\lambda = b/a$  after the residuals  $\varepsilon_i$  are minimized in the matrix equation  $\mathbf{X}\mathbf{C} = \mathbf{Y} + \boldsymbol{\varepsilon}$ . This minimization is implemented through the use of a coefficient matrix  $\mathbf{V}$  that is used to incorporate the known information such that

$$\mathbf{C} = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{W} \mathbf{Y}) \quad (10)$$

$$\mathbf{W} = \mathbf{V}^{-1} \quad (11)$$

In an OLS scheme no weighting is applied and a coefficient matrix  $\mathbf{V}$  consists of just a diagonal matrix with unity on each of the diagonal elements such that  $V_{i,i} = 1$ ,  $i = 1, \dots, n$  in a WLS scheme the coefficient matrix

consists of a diagonal matrix where the elements on each of the diagonal elements are proportional to the uncertainties of the areas i.e.  $V_{i,i} \propto u^2(S_i)$ , and finally in a GLS scheme the diagonal elements of  $\mathbf{V}$  equals the variances of the areas such that  $V_{i,i} = u^2(S_i)$  whilst the off-diagonal elements equal the covariances such that  $V_{i,j} = \text{cov}(S_i, S_j)$ ,  $i, j = 1, \dots, n$ ,  $i \neq j$ .

The general approach of Wuethrich and Souiyam [14] is to solve equation (9) with nominal values of the input quantities at specified pressures  $P_i$  for the equivalent homogeneous equation  $h = \frac{F_i}{P_i} - S_i$  in order to determine corresponding values of  $S_i$  for a sequence of pressures  $P_1, \dots, P_n$  with  $i = 1, \dots, n$ , and to then use this information to determine the regression parameter  $\mathbf{C}_j = [(A_0)_j, b_j]^T$  from the matrix equation with an appropriate choice of weighting matrix  $\mathbf{V}$ . This process is repeated  $M = 10000$  times with slightly perturbed values of the inputs consistent with the underlying uncertainties to sequentially generate samples  $\mathbf{C}_j$  for  $j = 1, \dots, M$  that then results in a sequence  $\mathbf{C}_1, \dots, \mathbf{C}_M$  of regression parameters, which are then post-processed to calculate expected values, variances, and the covariance of the regression parameters  $A_0$  and  $b = A_0\lambda$  where the distortion coefficient is extracted as  $\lambda_j = b_j / (A_0)_j$ . The pair of values  $((A_0)_j, \lambda_j)$  is then further post-processed according to the GUM [1] for two sampled random variables  $q_j$  and  $r_j$  to calculate the covariance  $\text{cov}(q, r)$  that is estimated as

$$\text{cov}(\bar{q}, \bar{r}) = \frac{1}{M(M-1)} \sum_{k=1}^M (q_k - \bar{q})(r_k - \bar{r}) \quad (12)$$

$$\bar{q} = \frac{1}{M} \sum_{i=1}^M q_i \quad (13)$$

$$\bar{r} = \frac{1}{M} \sum_{i=1}^M r_i \quad (14)$$

Investigations performed by Wuethrich and Souiyam [14] has concluded that a GLS regression produces high accuracy results, followed by the WLS regression which produces medium accuracy results, and finally by the OLS regression which produces low accuracy results. As a result the GLS approach may be considered the best overall approach however the key difficulty with implementing this approach is in the estimates of the covariances  $\text{cov}(S_i, S_j)$  that are required, which are either mathematically complex to determine or which must be numerically estimated with a large number of Monte Carlo simulations.

The key differences in the newer approach of Wuethrich and Souiyam [14] and an earlier approach of Ramnath [16] is that in the latter work a multivariate GUM Supplement 2 approach [17] was utilized that modelled the various inputs  $\mathbf{x}$  by treating the two parameters  $A_0$  and  $\lambda$  as a single vector  $\mathbf{y} = [A_0, \lambda]^T$  with an equation  $\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{0}$  where a multivariate PDF  $g_{\mathbf{y}}(\boldsymbol{\eta}_{\mathbf{y}})$  of the output was to be determined from a multivariate input  $g_{\mathbf{x}}(\boldsymbol{\xi}_{\mathbf{x}})$  by numerically solving the equation  $\mathbf{h}(\boldsymbol{\eta}_{\mathbf{y}}, \boldsymbol{\xi}_{\mathbf{x}}) = \mathbf{0}$  for a large number  $M$  of Monte Carlo simulation events.

In this approach sampled multivariate values of the inputs consistent with the joint PDFs of the inputs  $\mathbf{x}$  were used to solve the homogeneous vector equation corresponding to  $h_j = \frac{F_j}{P_j} - S_j$  and a sequence of pressures  $P_1, \dots, P_n$  and areas  $S_1, \dots, S_n$  were generated that was then used with an OLS approach to calculate sampled values of  $((A_0)_j, \lambda_j)$  from the joint PDF  $g_{\mathbf{y}}(\boldsymbol{\eta}_{\mathbf{y}})$ . The sampled values were then used to construct a bivariate copula that modelled  $u^2(A_0)$ ,  $u^2(\lambda)$  and  $cov(A_0, \lambda)$ , however a limitation of this approach was that it required a large number of Monte Carlo simulations, was mathematically complex, and thus more suited to scientific metrology applications in national metrology institutes.

Differences in the two approaches is that firstly the method of Wuethrich and Souiyam [14] requires estimates for the uncertainties of the areas  $u^2(S_1), \dots, u^2(S_n)$  in the case of a WLS approach and additionally that of the covariances  $cov(S_i, S_j)$ ,  $1 \leq i, j \leq n$  for each Monte Carlo event  $j$ , and secondly makes an assumption of a Gaussian correlation between  $A_0$  and  $\lambda$ . On the other hand the method of Ramnath [16] firstly avoids the explicit need for calculating  $u^2(S_i)$  and  $cov(S_i, S_j)$  at each Monte Carlo event  $j$  as the data is considered as statistical samples, and secondly does not assume a Gaussian correlation between  $A_0$  and  $\lambda$  since bivariate copulas can model non-Gaussian correlation effects.

It should be noted that the calculation of the covariances  $cov(S_i, S_j)$  that are required in a GLS regression whilst theoretically possible requires an additional matrix formulation solution as outlined in the GUM Supplement 2 [17], pp. 15–17. For an explicit matrix equation  $\mathbf{y} = \mathbf{f}(\mathbf{x})$  the covariance matrix of the output is  $\mathbf{C}_{\mathbf{y}} = \mathbf{C}_{\mathbf{x}} \mathbf{U}_{\mathbf{x}} \mathbf{C}_{\mathbf{x}}^T$  and for an implicit matrix equation  $\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{0}$  the covariance matrix of the output is solved from the matrix equation  $\mathbf{C}_{\mathbf{y}} \mathbf{U}_{\mathbf{y}} \mathbf{C}_{\mathbf{y}}^T = \mathbf{C}_{\mathbf{x}} \mathbf{U}_{\mathbf{x}} \mathbf{C}_{\mathbf{x}}^T$ , where  $\mathbf{U}_{\mathbf{x}}$  is a covariance matrix of the input, and  $\mathbf{C}_{\mathbf{x}} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \mathbf{J}_{\mathbf{x}}$  and  $\mathbf{C}_{\mathbf{y}} = \frac{\partial \mathbf{h}}{\partial \mathbf{y}} = \mathbf{J}_{\mathbf{y}}$  are corresponding sensitivity or Jacobian matrices.

Due to the combination of algebraic and numerical ill-conditioning complexity that is present for determining a multivariate covariance  $\mathbf{U}_{\mathbf{y}}$ , this calculation is usually only tractable from a pure numerical based Monte Carlo simulation using an appropriate homogeneous equation  $h_i = \frac{F_i}{P_i} - S_i$  for  $i = 1, \dots, n$  as previously discussed.

### 3 Mathematical modelling

We consider the standard case of a straight line equation  $y = ax + b$  with inputs  $x_i$ , outputs  $y_i$ , variances  $u^2(x_i)$  and  $u^2(y_i)$ , and covariances  $cov(x_i, y_i)$  for measurements  $i = 1, \dots, n$ . This system was originally analysed by Krystek and Anton [10] who termed this system as a weighted total least squares with correlation system and developed a WTLSC algorithm that is able to calculate the variances  $u^2(a)$  and  $u^2(b)$  and the covariance  $cov(a, b)$ . An earlier paper by Ramnath [18] utilized this scheme to quantify the variances  $u^2(A_0)$  and  $u^2(\lambda)$  but did not explicitly report on the covariance  $cov(A_0, \lambda)$  term.

Considering the straight line equation  $S = aP + b = A_0(1 + \lambda)$  it immediately follows that

$$A_0 = b \tag{15}$$

$$\lambda = \frac{a}{b} \tag{16}$$

From the above set of equations it follows that the variances are

$$u^2(A_0) = u^2(b) \tag{17}$$

$$\begin{aligned} u^2(\lambda) &= \left(\frac{\partial \lambda}{\partial a}\right)^2 u^2(a) + \left(\frac{\partial \lambda}{\partial b}\right)^2 u^2(b) \\ &\quad + 2 \frac{\partial \lambda}{\partial a} \frac{\partial \lambda}{\partial b} cov(a, b) \\ &= \frac{1}{b^2} u^2(a) + \frac{a^2}{b^4} u^2(b) - \frac{2a}{b^3} cov(a, b) \end{aligned} \tag{18}$$

Using  $a = A_0 \lambda$  it follows that

$$\begin{aligned} u^2(a) &= \left(\frac{\partial a}{\partial A_0}\right)^2 u^2(A_0) + \left(\frac{\partial a}{\partial \lambda}\right)^2 u^2(\lambda) \\ &\quad + 2 \frac{\partial a}{\partial A_0} \frac{\partial a}{\partial \lambda} cov(A_0, \lambda) \end{aligned}$$

Expanding out the partial derivatives in the above equation and solving for  $cov(A_0, \lambda)$  then yields

$$\begin{aligned} cov(A_0, \lambda) &= \frac{1}{2\lambda A_0} u^2(a) - \frac{\lambda}{2A_0} u^2(A_0) - \frac{A_0}{2\lambda} u^2(\lambda) \\ &= \frac{1}{b} \left[ cov(a, b) - \frac{a}{b} u^2(b) \right] \end{aligned} \tag{19}$$

### 4 Numerical simulations

Numerical simulations are performed using the summary of statistical data reported in Table 1 which is based on an earlier full  $M = 1000$  Monte Carlo simulation reported in detail in Ramnath [16]. In this table the columns are the values of  $P_k$ ,  $u(P_k)$ ,  $S_k$ ,  $u(S_k)$  and  $\rho(P_k, S_k)$  for  $k = 1, \dots, 10$ , where  $\rho(P_k, S_k)$  is the dimensionless correlation that is defined in terms of the covariance  $cov(P_k, S_k)$  by the formula

$$\rho(P_k, S_k) = \frac{cov(P_k, S_k)}{\sqrt{u^2(P_k)u^2(S_k)}} \tag{20}$$

Results are obtained by running the Octave/Matlab implementation of the WTLSC algorithm reported in Figure 1 with the script in Figure 2 as summarized in

**Table 1.** Statistical data for pressure balance cross-floating measurements for pressures  $P_k$  and areas  $S_k$  in approximate equal steps with  $k = 1, \dots, 10$  from  $P_1 = 50$  MPa through to  $P_{10} = 500$  MPa. The above data is sufficient to use as an input for the WTLSC algorithm and produces essentially the same quality of results as a full GLS Monte Carlo simulation. If the dimensionless correlation term  $\rho(P_k, S_k)$  is unknown or cannot be otherwise estimated it may be set to zero and the WTLSC algorithm will equivalently reproduce the results from a WLS algorithm which provides conservative estimates for the corresponding parameter expected values and associated variances/covariance. It should be noted that a zero correlation of the input/output data will in general still produce a non-zero correlation in the parameters to account for and model the coupling effect between the parameter random variables.

$k$	$P_k$ /[MPa]	$u(P_k)$ /[MPa]	$S_k$ /[mm <sup>2</sup> ]	$u(S_k)$ /[mm <sup>2</sup> ]	$\rho(P_k, S_k)$
1	49.8949758400	0.0089296671	1.9614349040	0.0001012499	-0.2240451872
2	99.7893658900	0.0191007200	1.9614617540	0.0000990361	-0.2541394667
3	149.6833836000	0.0294253924	1.9614690240	0.0001027761	-0.2782037624
4	199.5765267000	0.0398616767	1.9614745360	0.0001110578	-0.2888083360
5	249.4722013000	0.0488290818	1.9614781070	0.0001217275	-0.3393470915
6	299.3636620000	0.0617999429	1.9614790080	0.0001766929	-0.4174833922
7	349.2555759000	0.0731588130	1.9614751790	0.0002210962	-0.4587383198
8	399.1522063000	0.0942270814	1.9614826810	0.0002740451	-0.5507275721
9	449.0475852000	0.1132755048	1.9614670620	0.0003424950	-0.6075615254
10	498.9361994000	0.1386940818	1.9614930390	0.0004087028	-0.6700571979

**Table 2.** All of the results reported in this table were directly obtained from a Octave/Matlab program with the exception of the dimensionless correlation coefficient  $\rho(A_0, \lambda)$  as the available number of Monte Carlo simulation events at  $M=1000$  was too small to obtain meaningful statistical results with the use of equation (11).

Instead the correlation was estimated by saving the processed data from the Octave/Matlab program as a txt file and fitting a Gaussian copula which is defined in terms of the covariance using the following RStudio computer code [19]:

```
library(copula)
library(VineCopula)
xydata <- read.table("C:/mydata.txt")
x1 <- xydata[, 1]
x2 <- xydata[, 2]
fncF1 <- ecdf(x1)
fncF2 <- ecdf(x2)
u1 <- fncF1(x1)
u2 <- fncF2(x2)
C21 <- BiCopSelect(u1, u2, familyset = c(1))
rho <- C21$par
```

This trick was utilized since a calculation of a covariance/correlation is mathematically equivalent to the assumption of a Gaussian bivariate copula to model the coupling effect between two random variables, and the VineCopula statistical package [20] uses more advanced optimization algorithms to estimate the parameters for a covariance/correlation. In general if a large number of Monte Carlo simulation events for example  $M=10000$  is available then equation (11) can be used directly, and this observation illustrates the essential need for a large number of simulations to adequately apply a GLS regression in order to obtain a statistically meaningful result particu-

larly when a relatively small number of scattered data-points from a small number of Monte Carlo simulations is available for data analysis as shown in Figure 3.

It may be observed that in the case of a bivariate PDF that the joint PDF is then a three-dimensional function and as a result it becomes difficult to visualize the three PDFs  $g_{OLS}(\mathbf{x})$ ,  $g_{WLS}(\mathbf{x})$  and  $g_{GLS}(\mathbf{x})$  corresponding to the OLS, WLS and GLS regressions as these function overlap and intersect with one another. As a result in order to simplify the interpretation of the numerical results level curves of the PDFs are extracted as shown in Figure 4 so that the relative shape and sharpness of the PDFs may be meaningfully compared later in the paper.

Due to the complexity of the underlying pressure balance equations a large number of Monte Carlo simulations such as say  $M=10000$  is only feasible if the equations are implemented in an appropriate computer programming environment such as a Visual Basic for Applications (VBA) script within a Excel spreadsheet or a Matlab script, and as a result may present onerous technical computing demands on a pressure metrologist in an industrial calibration laboratory.

The results from Table 2 are visualized in Figure 5 from which it is confirmed that a OLS regression gives the lowest accuracy and a WLS gives a medium accuracy. The relative errors for the zero-pressure area  $\varepsilon_{A_0}^{(r)}$ , distortion coefficient  $\varepsilon_{\lambda}^{(r)}$ , correlation  $\varepsilon_{\rho}^{(r)}$ , distortion coefficient uncertainty  $\varepsilon_{u(\lambda)}^{(r)}$ , and zero-pressure uncertainty  $\varepsilon_{u(A_0)}^{(r)}$ , may be defined as

$$\varepsilon_Q^r = \frac{x_Q^T - x_Q^{GLS}}{x_Q^{GLS}},$$

$$T \in \{OLS, WLS, WTLSC\},$$

$$Q \in [A_0, \lambda, u(A_0), u(\lambda), \rho_{A_0, \lambda}] \quad (21)$$

```

1 function Odata = FitWTLSC(x, y, ux, uy, rhoxy)
2 % (c) Unisa 2020: Vishal Ramnath (ramnav@unisa.ac.za) 24/08/2020
3 % INPUTS: x, y, u(x), u(y), rho(x, y)
4 % OUTPUTS: y = ax + b: fits a, b, u(a), u(b), rho(a, b)
5 n = length(x);
6 q = polyfit(x, y, 1);
7 a0 = q(1);
8 b0 = q(2);
9 alpha0 = atan(a0);
10 function meritvalue = chisquared(theta)
11 g = (ux.^2)*sin(theta)^2 ...
12 - 2*rhoxy.*ux.*uy*sin(theta)*cos(theta)...
13 + (uy.^2)*cos(theta)^2;
14 u = ((1/n)*sum(1./g))^(1/2);
15 w = (u^2)./g;
16 xbar = (1/n)*sum(w.*x);
17 ybar = (1/n)*sum(w.*y);
18 p = ybar*cos(theta) - xbar*sin(theta);
19 v = y*cos(theta) - x*sin(theta) - p;
20 meritvalue = sum((v.^2)./g);
21 endfunction
22 foptimization = @(theta) chisquared(theta);
23 optimset('TolX', 1E-7);
24 [theta_min, f_min] = fminsearch(foptimization, alpha0);
25 alpha_optimum = theta_min;
26 theta = alpha_optimum;
27 g = (ux.^2)*sin(theta)^2 ...
28 - 2*rhoxy.*ux.*uy*sin(theta)*cos(theta)...
29 + (uy.^2)*cos(theta)^2;
30 u = ((1/n)*sum(1./g))^(1/2);
31 w = (u^2)./g;
32 xbar = (1/n)*sum(w.*x);
33 ybar = (1/n)*sum(w.*y);
34 p = ybar*cos(theta) - xbar*sin(theta);
35 v = y*cos(theta) - x*sin(theta) - p;
36 p_optimum = p;
37 v_alpha = -(y*sin(alpha_optimum) + x*cos(alpha_optimum));
38 v_alphaalpha = -(v + p_optimum);
39 g = 0.5*(ux.^2 + uy.^2) ...
40 - 0.5*(ux.^2 - uy.^2)*cos(2*alpha_optimum) ...
41 - rhoxy.*ux.*uy*sin(2*alpha_optimum);
42 g_alpha = (ux.^2 - uy.^2)*sin(2*alpha_optimum) ...
43 - 2*rhoxy.*ux.*uy*cos(2*alpha_optimum);
44 g_alphaalpha = 2*(ux.^2 + uy.^2) - 4*g;
45 chisquared_alphaalpha = sum(2*(v_alpha.^2 + v.*v_alphaalpha)./g ...
46 - (4*v.*v_alpha.*g_alpha + (v.^2).*g_alphaalpha).(g.^2) ...
47 + (2*(v.^2).*(g_alpha.^2))./(g.^3));
48 chisquared_pp = (2*n)/(u^2);
49 chisquared_palpha = -2*sum(v_alpha./g ...
50 - (v.*g_alpha).(g.^2));
51 c11 = chisquared_alphaalpha;
52 c12 = chisquared_palpha;
53 c21 = chisquared_palpha;
54 c22 = chisquared_pp;
55 C = (2/(c22*c11 - (c12)^2))*[c11, -c12; -c21, c22];
56 Var_p = C(1, 1);
57 Cov_palpha = C(1, 2);
58 Var_alpha = C(2, 2);
59 a = tan(alpha_optimum);
60 b = p_optimum/cos(alpha_optimum);
61 Var_a = Var_alpha/cos(alpha_optimum)^4;
62 Var_b = (Var_alpha*p_optimum^2*sin(alpha_optimum)^2 ...
63 + Var_p*cos(alpha_optimum)^2 ...
64 + Cov_palpha*p_optimum*sin(2*alpha_optimum))/(cos(alpha_optimum))^4;
65 Cov_ab = (Var_alpha*p_optimum*sin(alpha_optimum) ...
66 + Cov_palpha*cos(alpha_optimum))/(cos(alpha_optimum))^4;
67 Odata(1, 1) = a;
68 Odata(2, 1) = b;
69 Odata(3, 1) = sqrt(Var_a);
70 Odata(4, 1) = sqrt(Var_b);
71 Odata(5, 1) = Cov_ab/sqrt(Var_a*Var_b);
72 %Odata(1) = alpha_optimum;
73 %Odata(2) = p_optimum;
74 endfunction

```

**Fig. 1.** Gnu Octave/Matlab function file to implement weighted total least squares with correlation algorithm. The  $x$  input and its standard uncertainty  $u(x)$  have the same physical units, and similarly the  $y$  output and its standard uncertainty  $u(y)$  also have the same physical units, whilst the correlation  $\rho(x, y) = cov(x, y)/(u(x) \times u(y))$  is dimensionless.

```

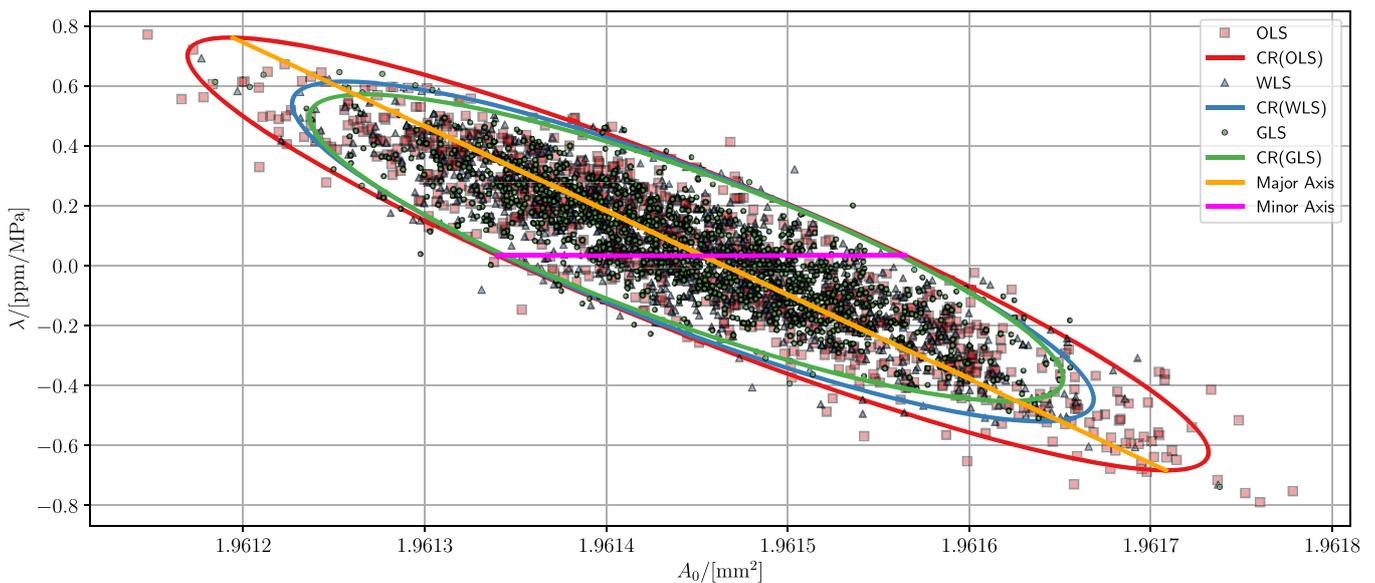
1 % WTLS: Weighted Total Least Squares
2 PB = load('pressurebalancedata.txt');
3 x = PB(:, 1); % the measured x-data
4 y = PB(:, 2); % the observed y-data
5 ux = PB(:, 3); % the measured uncertainty u(x)
6 uy = PB(:, 4); % the observed uncertainty u(y)
7 rhoxy = PB(:, 5); % the estimated dimensionless correlation rho(x, y)
8 Z = FitWTLS(x, y, ux, uy, rhoxy); % calls the program in Figure 1
9 a = Z(1);
10 b = Z(2);
11 ua = Z(3);
12 ub = Z(4);
13 rhoab = Z(5);
14 covab = rhoab*ua*ub;
15 muA0 = b;
16 muLambda = a/b;
17 sigmaA0 = ub;
18 sigmaLambda = sqrt((1/b^2)*ua^2 + (a^2/b^4)*ub^2 - (2*a/b^3)*covab);
19 covA0Lambda = (1/b)*(covab - (a/b)*ub^2);
20 rhoA0Lambda = covA0Lambda/(sigmaA0*sigmaLambda);

```

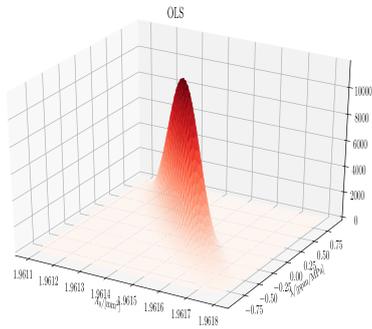
**Fig. 2.** Gnu Octave/Matlab script file using the function file FitWTLS.m to calculate the pressure balance parameter values and associated variance and covariance information in a typical pressure laboratory calibration certificate for a client pressure balance instrument. The data in the loaded file is from Table 1 and may conveniently be extracted from an Excel spreadsheet of measurement results and as saved a txt file.

**Table 2.** Final results and comparison of numerical experiments using the WTLS, OLS, WLS, and GLS methods.

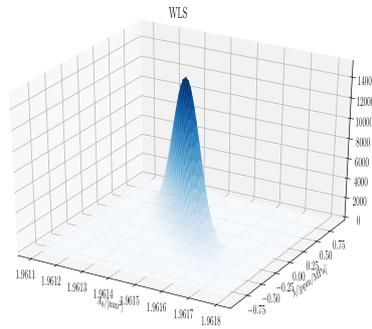
Quantity/Method	OLS	WLS	GLS	WTLS
$A_0$ /[mm <sup>2</sup> ]	1.9614505700	1.9614477500	1.9614441300	1.9614442200
$\lambda$ /[ppm/MPa]	0.0389765930	0.0474102619	0.0597407160	0.0599991615
$u(A_0)$ /[mm <sup>2</sup> ]	0.0001147822	0.0000901168	0.0000847245	0.0000853730
$u(\lambda)$ /[ppm/MPa]	0.2946518870	0.2313653420	0.2088520040	0.2146586350
$\rho(A_0, \lambda)$	-0.901647502	-0.855854363	-0.834854596	-0.8588924321



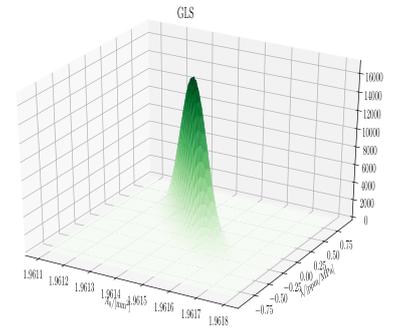
**Fig. 3.** Comparison of ordinary least squares (OLS), weighted least squares (WLS) and generalized least squares regression analysis from Monte Carlo data. Hyper-ellipsoids are generated to indicate the corresponding 95% confidence regions (CRs) and reveals that the highest accuracy scheme is that of a GLS which has the smallest CR which corresponds to a high degree of certainty of the range of possible values, followed by a WLS which has a slightly lower accuracy compared to a GLS, and finally a OLS which has the lowest accuracy as the CR is unduly large.



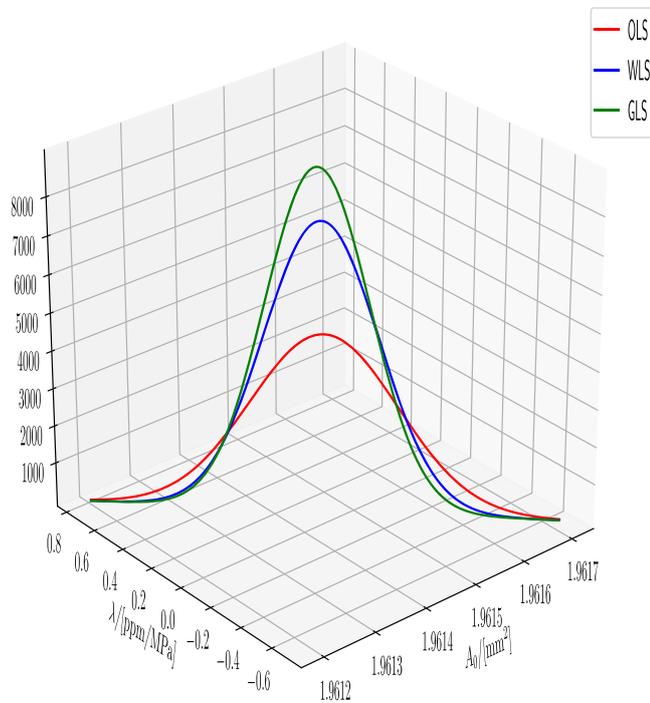
(a) OLS joint PDF



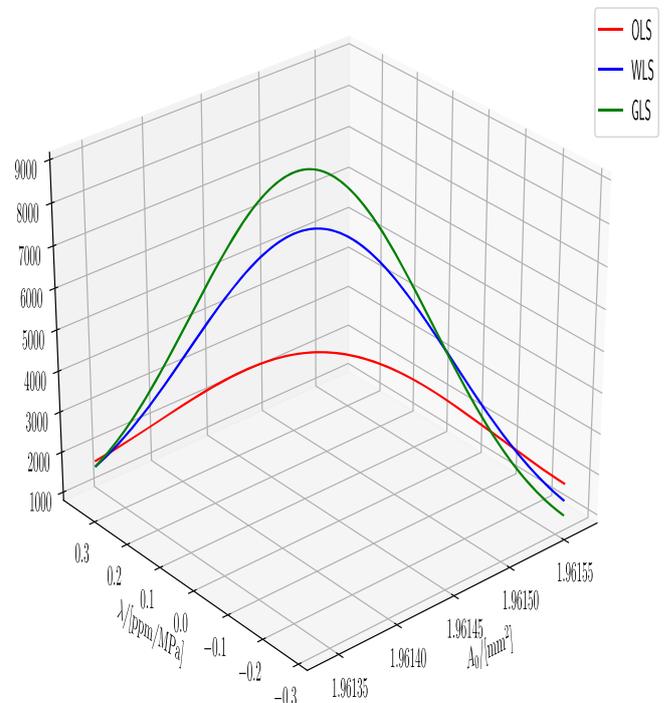
(b) WLS joint PDF



(c) GLS joint PDF

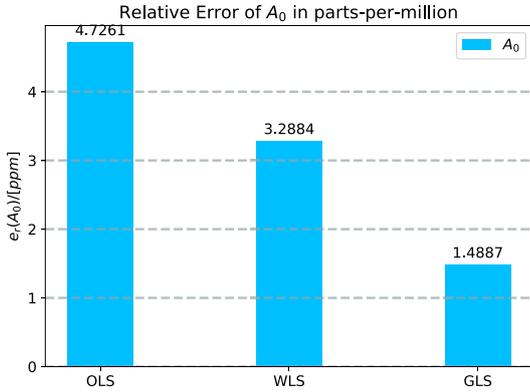


(d) Major axis level curve

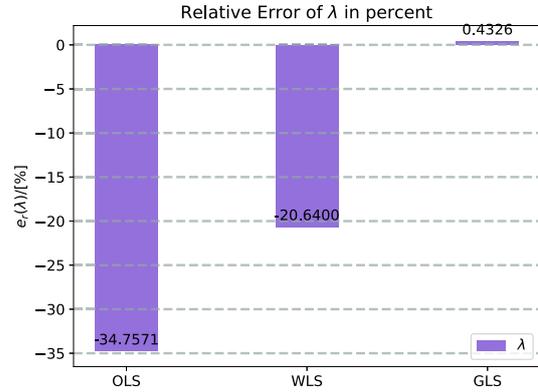


(e) Minor axis level curve

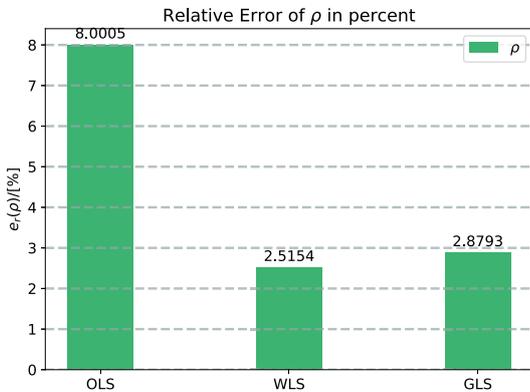
**Fig. 4.** Comparison of joint PDFs from OLS, WLS and GLS regressions. The confidence region may be visualized as a three dimensional volume underneath the envelope covered by the joint PDF and bounded by the hyper-ellipsoids, and indicates that the joint PDFs may fit inside one another. Due to the difficulties with visualizing three different joint PDFs intersecting within each other simultaneously, cuts along the major axis (orange line in previous figure) and minor axis (magenta line in previous figure) of the ellipsoids confidence regions are taken which produce level curves to more easily visualize the shape, orientation and size of the different OLS, WLS and GLS regression schemes. From these level curves it is observed that for level curves along the ellipsoid major axis that the GLS regression (green curve) has the narrowest/sharpest curve which corresponds to the highest confidence or accuracy in the estimates of the parameters, then the WLS regression (blue curve) which has a medium spread/sharpness, and finally the OLS regression (red curve) which is relative wide and flat which corresponds to less certainty/accuracy of the parameters in the regression analysis. A broadly similar behaviour is observed for level curves cut along the minor ellipsoid axis, and as a result it is observed and concluded a OLS regression has the least accuracy, a WLS regression has a medium accuracy, and a GLS regression has the highest accuracy. From the above graphs it may be concluded that a straight line regression for pressure metrology work should ideally utilize a GLS regression (green line) however if no covariance information is available the a WLS regression (blue line) can give a reasonable estimate particularly for industrial pressure metrology laboratories.



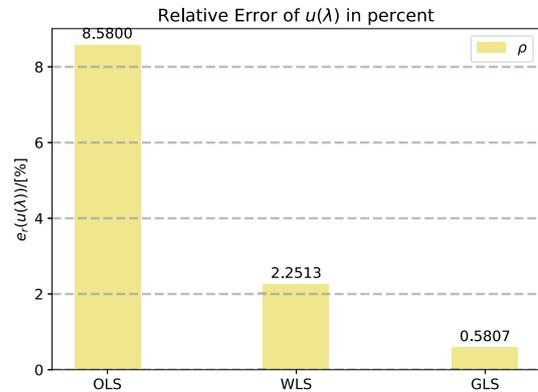
(a) Relative error of zero-pressure area



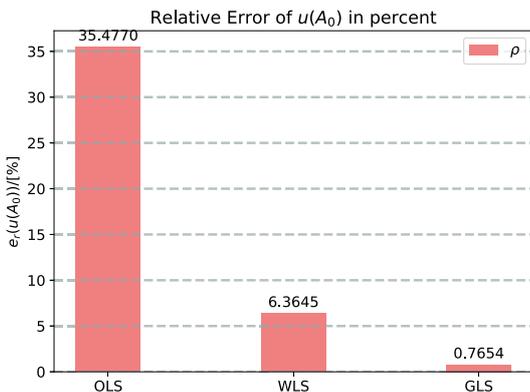
(b) Relative error of distortion coefficient



(c) Relative error of correlation



(d) Relative error of distortion coefficient uncertainty



(e) Relative error of zero-pressure uncertainty

**Fig. 5.** Comparison of relative errors of the quantities predicted by OLS, WLS and WTLSC regression approaches relative to predictions from a GLS regression approach. These graphs indicate that the relative error between a WTLSC regression and a GLS regression is relatively insignificant, and as a result a pressure metrologist instead of applying a more complicated GLS regression can simply apply a WTLSC regression with the aid of the supplied methodology developed in this paper.

where  $Q$  represents a selection of a quantity and  $T$  denotes a selection of a technique.

Considering the relative errors for the WTLSC regression when compared to predictions from the GLS which is considered as the most accurate technique, it is observed that the relative errors are  $\epsilon_{A_0}^{(r)} = 0.04610$  ppm,  $\epsilon_{\lambda}^{(r)} = 0.433\%$ ,  $\epsilon_{\rho}^{(r)} = 2.879\%$ ,  $\epsilon_{u(\lambda)}^{(r)} = 2.780\%$ , and

$\epsilon_{u(A_0)}^{(r)} = 0.765\%$  respectively for the representative pressure balance data that was analysed. As a result it may be concluded that a WTLSC algorithm gives essentially the same quality of a results as full GLS Monte Carlo simulation if there are available estimates for the correlation terms  $cov(P_i, S_i)$  for the data points  $i = 1, \dots, n$ .

The key benefits of utilizing a WTLSC algorithm is that it is a relatively straightforward implementation of using a Octave/Matlab function file FitWTLSC with about 20 lines of computer code as outlined in Figure 2, and does not require extensive or advanced programming skills for a pressure metrologist to implement. Limitations of the method are nevertheless present in that estimates for the correlation terms  $\rho(P_i, S_i)$ ,  $i=1, \dots, n$  are technically necessary to utilize the WTLSC algorithm.

This issue may be addressed in one of two practical ways, namely (i) setting zero correlations for all of the cross-float data-points such that  $\rho(P_i, S_i) = 0$  for  $i=1, \dots, n$  for which the WTLSC algorithm will reproduce results for a WLS algorithm as a conservative compromise, or (ii) performing a small number of say 50 or 100 repeat measurements in an Excel spreadsheet which is feasible without the need for any specialist VBA scripting programming within a spreadsheet with appropriate expert physical judgement to generate  $(P_i, S_i)$  data to fit a Gaussian copula with the 11 lines of RStudio code previously discussed to obtain an approximate estimate for the correlation.

Referring to the results in Figure 3 it is observed that the PDFs of the OLS, WLS and GLS regressions may be compared by examining the behaviour of the PDFs along the semi-major and semi-minor axes of the hyper-ellipsoid for the confidence region. When this analysis was performed as visualized in Figure 4 it was observed that the WLS regression performed relatively well when compared to the GLS regression. As a result it may be concluded that the limitations of the potential absence of a covariance matrix is not excessively detrimental.

## 5 Conclusions

Based on the research reported in this paper the following conclusions were determined:

- Numerical simulations have been performed that demonstrate that a weighted total least squares with correlation (WTLSC) algorithm gives essentially equivalent results for straight line parameters as a more advanced generalized least squares (GLS) algorithm with additional Monte Carlo simulations for parameter uncertainty estimation
- A method has been developed for the WTLSC algorithm that is able to analytically extract the variance/covariance information of the straight line parameters without the need for additional and more advanced Monte Carlo simulations

## 6 Influences and implications

Based on the research reported in this paper the following influences and implications are:

- Pressure metrologists working in industrial calibration laboratories may now utilize the WTLSC algorithm to conveniently and easily quantify the expected values and

variances/covariance of the zero-pressure area  $A_0$  and distortion coefficient  $\lambda$

- A computer program FitWTLSC.m is now freely and publicly available to conveniently implement the WTLSC algorithm in Gnu Octave or Matlab for calculating the straight line parameters and their associated variance/covariance information and may now also be used by other metrologists from different fields

This work was performed with funds provided by the Department of Higher Education, Science and Technology (DHEST) on behalf of the South African government for research by public universities.

## References

1. BIPM, IEC, IFCC, ILAC, ISO, IUPAP, and OIML, Evaluation of measurement data – Guide to the expression of uncertainty in measurement, tech. rep., JCGM/WG1 GUM, 2008. Revised 1st edition – <https://www.bipm.org/en/publications/guides/>
2. R.S. Dadson, S.L. Lewis, G.N. Peggs, *The Pressure Balance: Theory and Practice* (HMSO, London, 1982) ISBN 0114800480.
3. P. Saunders, Propagation of uncertainty for non-linear calibration equations with an application in radiation thermometry, *Metrologia* **40**, 93–101 (2003)
4. S. Vanhuffel, J. Vandewalle, *The Total Least Squares Problem: Computational Aspects and Analysis* (SIAM, 1987)
5. D. York, Least squares fitting of a straight line with correlated errors, *Earth Planet. Sci. Lett.* **5**, 320–324 (1968)
6. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes: The Art of Scientific Computing* (Cambridge University Press, 2007), 3rd edn
7. R.L. Burden, J.D. Faires, *Numerical Analysis* (Brookes/Cole, 2001), 7th edn
8. W.C. Duer, P.J. Ogren, A. Meetze, C.J. Kitchen, R.V. Lindern, D.C. Yaworsky, C. Boden, J.A. Gayer, Comparison of ordinary, weighted, and generalized least-squares straight-line calibrations for LC-MS=MS, GC-MS, HPLC, GC, and enzymatic assay, *J. Anal. Toxicol.* **32**, 329–338 (2008)
9. M. Krystek, M. Anton, A weighted total least-squares algorithm for fitting a straight line, *Measur. Sci. Technol.* **18**, 3438–3442 (2007)
10. M. Krystek, M. Anton, A least-squares algorithm for fitting data points with mutually correlated coordinates to a straight line, *Measur. Sci. Technol.* **22**, 035101 (2011)
11. P.M. Harris, C.E. Matthews, M.G. Cox, Summarizing the output of a Monte Carlo method for uncertainty evaluation, *Metrologia* **51**, 243–252 (2014)
12. Mathworks, fgls – Feasible generalized least squares, 2020. <https://uk.mathworks.com/help/econ/fgls.html>
13. S. Miller, R. Startz, Feasible generalized least squares using support vector machines, *Econ. Lett.* **175**, 28–31 (2019)
14. C. Wuethrich, S. Souiyam, Monte Carlo determination of the uncertainty of effective area and deformation coefficient for a piston cylinder unit, in *24th IMEKO TC-3, 14th TC-5, 6th TC-16 and 5th TC-22 International Conference*, edited by A. Salceanu, D. Agrez, J. Saliga, M. Savino (Cavtat-Dubrovnik, Croatia: IMEKO, 2020), pp. 1–6

15. P. Otal, C. Yardin, Modelling methods for pressure balance calibration, Measur. Sci. Technol. **31**, 034004 (2019)
16. V. Ramnath, Numerical analysis of the accuracy of bivariate quantile distributions utilizing copulas compared to the GUM supplement 2 for oil pressure balance uncertainties, Int. J. Metrol. Qual. Eng. **8**, 29 (2017)
17. BIPM, IEC, IFCC, ILAC, ISO, IUPAP, and OIML, Evaluation of measurement data – Supplement 2 to the Guide to the expression of uncertainty in measurement – Propagation of distributions using a Monte Carlo method, tech. rep., JCGM/WG1 GUM Supplement 2, 2011. 1st edition – <https://www.bipm.org/en/publications/guides/>
18. V. Ramnath, Determination of pressure balance distortion coefficient and zero-pressure effective area uncertainties, Int. J. Metrol. Qual. Eng. **2**, 101–119 (2011)
19. CRAN, The R Project for Statistical Computing, 2020. <https://www.r-project.org/about.html>
20. CRAN, VineCopula: Statistical Inference of Vine Copulas, 2020. <https://cran.r-project.org/web/packages/VineCopula/index.html>
21. V. Ramnath, Analysis and comparison of hyper-ellipsoidal and smallest coverage regions for multivariate Monte Carlo measurement uncertainty analysis simulation datasets, MAPAN-J. Metrol. Soc. India 1–16 (2019)
22. S.-N. Lee, M.-H. Shih, A volume problem for an  $n$ -dimensional ellipsoid intersecting with a hyperplane, Linear Algebra Appl. **132**, 90–102 (1990)
23. E.W. Weisstein, Ellipse – a Wolfram web resource (2020). <https://mathworld.wolfram.com/Ellipse.html>

**Cite this article as:** Vishal Ramnath, Comparison of straight line curve fit approaches for determining parameter variances and covariances, Int. J. Metrol. Qual. Eng. **11**, 14 (2020)

### Appendix A: Hyper-ellipsoidal bivariate confidence region

Let random variables  $\mathbf{x}_1, \dots, \mathbf{x}_M \in \mathbb{R}^N$  be grouped as  $\mathbf{\Omega} = [\mathbf{x}_1 \dots \mathbf{x}_M]$  where  $\mathbf{x}_j, j = 1, \dots, M$  are  $N \times 1$  vectors,  $\mathbf{\Omega}$  is a  $N \times M$  matrix, and  $M$  is a suitably large positive integer number of Monte Carlo simulation events. Then the expected value  $\boldsymbol{\mu}$  a  $N \times 1$  vector and the covariance matrix  $\mathbf{U}$  a  $N \times N$  matrix may be estimated with the GUM Supplement 2 [17] as  $\boldsymbol{\mu}$  and  $\mathbf{U}$ . Under these circumstances the joint PDF, unless higher order statistics information over and above the expected and covariance values are provided, is then assumed to follow a multivariate Gaussian distribution  $g(\mathbf{x}) \sim (\boldsymbol{\mu}, \mathbf{U})$ .

The corresponding joint PDF is then  $g(\mathbf{x})$  and consists of hyper-ellipsoids within a  $\mathbb{R}^N$  dimensional space that are centred at  $\boldsymbol{\mu}$  and have an orientation determined from characteristics of  $\mathbf{U}$  such that

$$\boldsymbol{\mu} = \frac{1}{M} \sum_{j=1}^M \mathbf{x}_j \tag{A1}$$

$$\mathbf{U} = \frac{1}{M-1} \sum_{j=1}^M (\mathbf{x}_j - \boldsymbol{\mu}) \cdot (\mathbf{x}_j - \boldsymbol{\mu})^T \tag{A2}$$

$$g(\mathbf{x}) = \left[ (2\pi)^N \det(\mathbf{U}) \right]^{-1/2} \times \exp \left( -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \mathbf{U}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right) \tag{A3}$$

In the special case of bivariate data where  $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ ,  $\boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$  and  $\mathbf{U} = \begin{bmatrix} U_{11} & U_{12} \\ U_{12} & U_{22} \end{bmatrix}$ , the hyper-ellipsoid will lie in the  $x_1x_2$ -plane within a 2 dimensional space, and the joint PDF will be

$$g(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \times \exp \left( -\frac{1}{2(1-\rho^2)} \times \left[ \frac{(x_1 - \mu_1)^2}{\sigma_1^2} + \frac{(x_2 - \mu_2)^2}{\sigma_2^2} - \frac{2\rho(x_1 - \mu_1)(x_2 - \mu_2)}{\sigma_1\sigma_2} \right] \right) \tag{A4}$$

For the above equation the terms are  $\sigma_1 = \sqrt{U_{11}}$ ,  $\sigma_2 = \sqrt{U_{22}}$ , and  $\rho = U_{12}/\sqrt{U_{11}U_{22}}$  since the covariance matrix may equivalently be written as  $\mathbf{U} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$ . According to the GUM Supplement 2 [17] the 100p% coverage region is a hyper-ellipsoid

of the form

$$(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{U}^{-1} (\mathbf{x} - \boldsymbol{\mu}) = k_p^2 \tag{A5}$$

An earlier analysis by Ramnath [21] determined that the ellipsoid semi-axis values  $a_1$  and  $a_2$  could be approximately determined by forming the matrix

$$\mathbf{Q} \approx \frac{1}{k_p^2} \mathbf{U}^{-1} \tag{A6}$$

Once this matrix was formed the two eigenvectors  $\mathbf{v}_1$  and  $\mathbf{v}_2$  along with the corresponding eigenvalues  $\lambda_1$  and  $\lambda_2$  were calculated from the equation

$$\mathbf{Q}\mathbf{v} = \lambda\mathbf{v} \tag{A7}$$

The ellipse semi-axis values were then calculated as

$$\lambda_1 = \frac{1}{a_1^2} \tag{A8}$$

$$\lambda_2 = \frac{1}{a_2^2} \tag{A9}$$

and the eigenvectors  $v_1$  and  $v_2$  then define the orientations of the ellipsoid principal axes following the analysis by Lee [22]. The disadvantage of this approach is that an appropriate software routine for determining eigenvectors and eigenvalues is necessary, however an advantage of this approach is that it is generalisable to higher dimensions.

An alternative direct algebraic approach is to directly substitute the statistical information  $\boldsymbol{\mu}$  and  $\mathbf{U}$  into the formula equation (A5) and expand out so that

$$\begin{aligned} &\sigma_2^2(x_1 - \mu_1)^2 + \sigma_1^2(x_2 - \mu_2)^2 \\ &- 2\rho\sigma_1\sigma_2(x_1 - \mu_1)(x_2 - \mu_2) \\ &- k_p^2\sigma_1^2\sigma_2^2(1 - \rho^2) = 0 \end{aligned} \tag{A10}$$

The above equation is in the form of the generalized ellipse equation as discussed by Weisstein [23] that is centred at  $x_o$  and  $y_o$ , with semi-major and semi-minor axes  $a$  and  $b$  and rotated by an angle  $\theta$  of the form

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \tag{A11}$$

The algebraic coefficients may now be expressed in terms of the statistical terms as

$$A = \sigma_2^2 \tag{A12}$$

$$B = -2\rho\sigma_1\sigma_2 \tag{A13}$$

$$C = \sigma_1^2 \tag{A14}$$

$$D = -2\sigma_2^2\mu_1 + 2\rho\sigma_1\sigma_2\mu_2 \tag{A15}$$

$$E = -2\sigma_1^2\mu_2 + 2\rho\sigma_1\sigma_2\mu_1 \tag{A16}$$

$$F = \sigma_2^2\mu_1^2 + \sigma_1^2\mu_2^2 - 2\rho\sigma_1\sigma_2\mu_1\mu_2 - k_p^2\sigma_1^2\sigma_2^2(1 - \rho^2) \tag{A17}$$

The corresponding ellipse geometry parameters may now be algebraically calculated as

$$a, b = \frac{-1}{B^2 - 4AC} \left[ 2(AE^2 + CD^2 - BDE) + (B^2 - 4AC)F \right] \times \left[ (A + C) \pm \sqrt{(A - C)^2 + B^2} \right]^{1/2} \tag{A18}$$

$$x_o = \frac{2CD - BE}{B^2 - 4AC} \tag{A19}$$

$$y_o = \frac{2AE - BD}{B^2 - 4AC} \tag{A20}$$

and the ellipse rotation is given as

$$\theta = \begin{cases} \arctan\left(\frac{C - A - \sqrt{(A - C)^2 + B^2}}{B}\right) & \text{if } B \neq 0 \\ 0 & \text{if } B = 0, A < C \\ \frac{\pi}{2} & \text{if } B = 0, A > C \end{cases} \tag{A21}$$

The hyper-ellipsoid may then be conveniently plotted using the parametrization  $0 \leq t \leq 2\pi$  so that the initial boundary points  $x_i$  and  $y_i$  are

$$x_i = a \cos(t) \tag{A22}$$

$$y_i = b \sin(t) \tag{A23}$$

Next the boundary points are rotated by the angle  $\theta$  by using a rotation matrix to give  $x_r$  and  $y_r$  such that

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{A24}$$

Finally the points are translated as

$$x_f = x_r + \mu_1 \tag{A25}$$

$$y_f = y_r + \mu_2 \tag{A26}$$

This algorithm is used to produce the hyper-ellipsoids in Figure 3 and may be used to determine the points along the ellipse major and minor axes. As a result if these lines are used a ‘‘cut’’ through the joint PDF may be used to

study how the level curves of the different OLS, WLS, and GLS regressions behave as it is too difficult to easily visualize the three dimensional joint PDFs in Figure 4 even with transparency effects. Once the lines are determine the  $x_1$  and  $x_2$  points are known from a simple straight line fit so the  $x_1$  and  $x_2$  points may then be substituted in the bivariate joint PDF in equation (A4) which will produce a density value. The set of points  $(x_1, x_2, g(x_1, x_2))$  may then be plotted as a three-dimensional curve in order to visualize the shape and sharpness of the corresponding level curve of the PDF. This algebraic approach is only suitable for bivariate simulations and becomes impractical in higher dimensions, for which the eigenvector/eigenvalue approach is then necessary.

### Appendix B: Simplified worked example

In order to illustrate the method suppose that the experimental data has  $n = 5$  points of generated pressures from 100 MPa through to 500 MPa in 100 MPa steps and cross-floated areas, which will give an experimental control variable of  $\mathbf{x} = [x_1, x_2, x_3, x_4, x_5]$  and an experimental observed variable of  $\mathbf{y} = [y_1, y_2, y_3, y_4, y_5]$ . Writing down the basis equation with the experimental data (for one particular Monte Carlo event) as follows.

$$\begin{bmatrix} 1 & 99.789 \\ 1 & 199.576 \\ 1 & 299.363 \\ 1 & 399.152 \\ 1 & 498.936 \end{bmatrix} \begin{bmatrix} A_0 \\ b \end{bmatrix} = \begin{bmatrix} 1.9614617540 \\ 1.9614745360 \\ 1.9614790080 \\ 1.9614826810 \\ 1.9614930390 \end{bmatrix}$$

Let

$$\mathbf{X} = \begin{bmatrix} 1 & 99.789 \\ 1 & 199.576 \\ 1 & 299.363 \\ 1 & 399.152 \\ 1 & 498.936 \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} 1.9614617540 \\ 1.9614745360 \\ 1.9614790080 \\ 1.9614826810 \\ 1.9614930390 \end{bmatrix}$$

Considering an ordinary least squares (OLS) regression the estimate for the matrix  $\mathbf{V}$  is just

$$\mathbf{V}_{OLS} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

where  $\mathbf{W} = \mathbf{V}^{-1}$ .

```

1 clear all
2 clc
3 format long e
4 % y = ax + b, S = aP + b ==> A0 = b, Lambda = a/b
5 disp('Fitting of Pressure/Area Data: A0(sq-mm) and Lambda(ppm/MPa)')
6 disp('A0, Lambda, u(A0), u(Lambda), rho(A0,Lambda)')
7 PB = load('pressurebalancedata.txt');
8 % matrix of n x 6 points Pi,Si,u(Pi),u(Si),rho(Pi,Si),1<=i<=n
9 Pdata = load('MCSP.txt'); % matrix 1000 x 10 points of P1,...,Pn
10 Sdata = load('MCSA.txt'); % matrix 1000 x 10 points of S1,...,Sn
11 n = 10; % number of cross-floats (P_1,S_1)...,(P_n,S_n)
12 M = 1000; % the number of Monte Carlo simulation events
13 X = zeros(n, 2); X(:, 1) = 1; Y = zeros(n, 1);
14 % OLS: Ordinary Least Squares
15 V = eye(n); W = inv(V);
16 for j = 1:M
17     X(:, 2) = Pdata(j, :)';
18     Y(:, 1) = Sdata(j, :)';
19     C = inv(X'*W*X)*(X'*W*Y);
20     D(j, :) = C';
21 end
22 R = D; R(:, 2) = R(:, 2)./R(:, 1); R(:, 2) = 1E6*R(:, 2);
23 ROLS = R;
24 save([pwd, '\ROLS.txt'], 'R', '-ascii')
25 A0 = R(:, 1); Lambda = R(:, 2); muA0 = (1/M)*sum(A0); muLambda = (1/M)*sum(
    Lambda);
26 sigmaA0 = std(A0); sigmaLambda = std(Lambda);
27 covA0Lambda = (1/(M*(M-1)))*sum((A0 - muA0).*(Lambda - muLambda))
28 rhoA0Lambda = covA0Lambda/(sigmaA0*sigmaLambda)
29 ResultsOLS = [muA0, muLambda, sigmaA0, sigmaLambda, rhoA0Lambda]';
30 ZOLS = ResultsOLS;
31 % WLS: Weighted Least Squares
32 V = eye(n); uy = PB(:, 4);
33 for i = 1:n
34     V(i, i) = uy(i);
35 end
36 W = inv(V);
37 for j = 1:M
38     X(:, 2) = Pdata(j, :)';
39     Y(:, 1) = Sdata(j, :)';
40     C = inv(X'*W*X)*(X'*W*Y);
41     D(j, :) = C';
42 end
43 R = D; R(:, 2) = R(:, 2)./R(:, 1); R(:, 2) = 1E6*R(:, 2); RWLS = R;
44 save([pwd, '\RWLS.txt'], 'R', '-ascii')
45 A0 = R(:, 1); Lambda = R(:, 2); muA0 = (1/M)*sum(A0);
46 muLambda = (1/M)*sum(Lambda); sigmaA0 = std(A0);
47 sigmaLambda = std(Lambda);
48 covA0Lambda = (1/(M*(M-1)))*sum((A0 - muA0).*(Lambda - muLambda));
49 rhoA0Lambda = covA0Lambda/(sigmaA0*sigmaLambda);
50 ResultsWLS = [muA0, muLambda, sigmaA0, sigmaLambda, rhoA0Lambda]';
51 ZWLS = ResultsWLS;
52 % GLS: Generalized Least Squares
53 V = zeros(n, n);
54 for i = 1:n
55     for j = 1:n
56         Si = Sdata(:, i);
57         Sj = Sdata(:, j);
58         Sibar = (1/M)*sum(Si);
59         Sjbar = (1/M)*sum(Sj);
60         V(i, j) = (1/(M*(M-1)))*sum((Si - Sibar).*(Sj - Sjbar));
61     end
62 end
63 W = inv(V);
64 for j = 1:M
65     X(:, 2) = Pdata(j, :)';
66     Y(:, 1) = Sdata(j, :)';
67     C = inv(X'*W*X)*(X'*W*Y);
68     D(j, :) = C';
69 end
70 R = D; R(:, 2) = R(:, 2)./R(:, 1); R(:, 2) = 1E6*R(:, 2);
71 RGLS = R; save([pwd, '\RGLS.txt'], 'R', '-ascii')
72 A0 = R(:, 1); Lambda = R(:, 2); muA0 = (1/M)*sum(A0);
73 muLambda = (1/M)*sum(Lambda); sigmaA0 = std(A0);
74 sigmaLambda = std(Lambda);
75 covA0Lambda = (1/(M*(M-1)))*sum((A0 - muA0).*(Lambda - muLambda));
76 rhoA0Lambda = covA0Lambda/(sigmaA0*sigmaLambda);
77 ResultsGLS = [muA0, muLambda, sigmaA0, sigmaLambda, rhoA0Lambda]';
78 ZGLS = ResultsGLS;
79 % WTLSC: Weighted Total Least Squares
80 x = PB(:, 1); % the measured x-data
81 y = PB(:, 2); % the observed y-data
82 ux = PB(:, 3); % the measured uncertainty u(x)
83 uy = PB(:, 4); % the observed uncertainty u(y)
84 rhoxy = PB(:, 5); % the estimated dimensionless correlation rho(x, y)
85 Z = FitWTLSC(x, y, ux, uy, rhoxy); % calls the program in Figure 1
86 a = Z(1); b = Z(2); ua = Z(3); ub = Z(4); rhoab = Z(5);
87 covab = rhoab*ua*ub; muA0 = b; muLambda = a/b;
88 sigmaA0 = ub; sigmaLambda = sqrt((1/b^2)*ua^2 + (a^2/b^4)*ub^2 - (2*a/b^3)*covab
    );
89 covA0Lambda = (1/b)*(covab - (a/b)*ub^2);
90 rhoA0Lambda = covA0Lambda/(sigmaA0*sigmaLambda);
91 ResultsWTLSC = [muA0, 1E6*muLambda, sigmaA0, 1E6*sigmaLambda, rhoA0Lambda]';
92 ZWTLSC = ResultsWTLSC; Z = [ZOLS, ZWLS, ZGLS, ZWTLS]';
93 save([pwd, '\CompareResults.txt'], 'Z', '-ascii')

```

Fig. B1. Gnu Octave/Matlab function file to implement OLS, WLS and GLS regression analysis.

Implementing the formula in Matlab or GNU Octave with the above information gives

$$\begin{aligned} \mathbf{C}_{OLS} &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{W} \mathbf{Y}) \\ &= \begin{bmatrix} 1.96145698896178e + 00 \\ 7.08658854028932e - 08 \end{bmatrix} \\ &= \begin{bmatrix} A_0 \\ b \end{bmatrix} \end{aligned}$$

Next considering a WLS regression it is known that the  $V_{i,i} \propto u^2(S_i)$  so using the data from Table 1 let  $u(S_1) = 0.0000990361$ ,  $u(S_2) = 0.0001110578$ ,  $u(S_3) = 0.0001766929$ ,  $u(S_4) = 0.0002740451$  and  $u(S_5) = 0.0004087028$ , and then simply let

*See equation below.*

Using Matlab/Octave to implement the matrix operations then produces

$$\mathbf{C}_{WLS} = \begin{bmatrix} 1.96145549101770e + 00 \\ 7.96180234449717e - 08 \end{bmatrix}$$

The above process is repeated for all of the experimental data points and then the results can be post-processed in order to obtain the statistical characteristics for the OLS, WLS and GLS regressions as shown in Figure 3.

Actual physical data is first obtained from the experimental measurements in the laboratory and this data is used as inputs into the instrument models to produce the data  $\mathbf{X}$  and  $\mathbf{Y}$  in this example where the computer program outlines the entire logical calculation process. As an example the piston temperature may have been measured as say 20.4°C with a standard uncertainty of  $\pm 0.1^\circ\text{C}$  with an assumed Gaussian distribution, then random temperatures consistent with the physical experimental uncertainty are sampled and used as inputs into the physical model. This is due to the fact that modern

measurement science has generally adopted a Bayesian statistics framework for measurement uncertainty and as a result no physical measurement is deterministic since all physical measurements, with the partial exception of certain physical constants, have an underlying probability distribution i.e. any physical measurement is considered as a statistical sample from that random variables probability density function. Once sufficient samples have been generated and used as inputs/outputs into the measurement model then the OLS, WLS or GLS regressions may be implemented as outlined in Figure B1.

The general steps to perform the regression analysis are therefore as follows:

- Obtain laboratory nominal experimental measurement data  $\mathbf{x}$  that has an associated joint PDF  $g_{\mathbf{x}}(\mathbf{x})$
- Sample a random variable  $\boldsymbol{\xi}$  from the joint PDF  $g_{\mathbf{x}}(\mathbf{x})$
- Solve the associated measurement equation  $\mathbf{h}(\boldsymbol{\eta}, \boldsymbol{\xi}) = \mathbf{0}$  to obtain the random variable  $\boldsymbol{\eta}$  associated with  $\boldsymbol{\xi}$ , and repeat this several times to produce the Monte Carlo data  $\boldsymbol{\Omega}$
- Step through the data input  $\boldsymbol{\xi}$  and data output  $\boldsymbol{\eta}$  i.e. the inputs  $\mathbf{x}$  and outputs  $\mathbf{y}$  and substitute this in the regression equation  $\mathbf{C} = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{W} \mathbf{Y})$
- Solve the regression equation for  $\mathbf{C}$  and then post-process all the results to work out the statistical characteristics of  $A_0$  and  $b$  namely  $u^2(A_0)$ ,  $u^2(b)$  and  $cov(A_0, b)$
- Simply apply the new algebraic formula to work out  $u^2(A_0) = u^2(b)$ ,  $u^2(\lambda) = \frac{1}{b^2} u^2(a) + \frac{a^2}{b^4} u^2(b) - \frac{2a}{b^3} cov(a, b)$ , and  $cov(A_0, \lambda) = \frac{1}{b} [cov(a, b) - \frac{a}{b} u^2(b)]$

The above computer code routines for performing the statistical regressions may be observed to be relatively lengthy and complicated, and may be compared to the WTLSC implementation which is simply the 20 lines of Matlab/Octave code in Figure 2. In this approach there is no need to perform any Monte Carlo simulations and the actual physical experimental data, in this simplified problem the  $n=5$  data points, may be used directly as inputs into the Octave/Matlab program FitWTLSC.m to directly extract the uncertainty information of the parameters  $A_0$  and  $\lambda$  and obtain  $u^2(A_0)$ ,  $u(\lambda)$  and  $cov(A_0, \lambda)$ .

---


$$\mathbf{V}_{WLS} = \begin{bmatrix} (0.0000990361)^2 & 0 & 0 & 0 & 0 \\ 0 & (0.0001110578)^2 & 0 & 0 & 0 \\ 0.4 & 0 & (0.0001766929)^2 & 0 & 0 \\ 0 & 0 & 0 & (0.0002740451)^2 & 0 \\ 0 & 0 & 0 & 0 & (0.0004087028)^2 \end{bmatrix}$$