

GENERATION OF NUMERICAL ARTEFACTS FOR GEOMETRIC FORM AND TOLERANCE ASSESSMENT

A. B. Forbes and H. D. Minh

National Physical Laboratory, Teddington, UK, alistair.forbes@npl.co.uk

Abstract: This paper describes an approach to generating reference data sets to evaluate the performance of algorithms used in coordinate metrology for form and geometric tolerance assessment. The approach starts with the reference results, e.g., the solution feature parameter values, and then determines the coordinate data. In this way, the expensive development of reference software is avoided. In this paper we consider the generation of reference data associated with determining the best-fit surface according to the least squares (Gaussian) and Chebyshev criteria.

Keywords: data generator, form assessment, numerical artefact, optimality conditions.

1. INTRODUCTION

Much of coordinate metrology depends on complex computations involving the analysis of coordinate data, e.g., to determine the best-fit geometric feature to a set of coordinate data according to the least squares, Chebyshev, or other criteria [1,2,3]. In order that traceability of coordinate metrology is maintained it is necessary to examine the computational links in the traceability chain in order to make sure they are fit for purpose [4].

We suppose the computational task can be described in terms of a deterministic function $\mathbf{a} = C(\mathbf{x})$, where \mathbf{x} represents the input data and \mathbf{a} is the exact mathematical solution of the computational problem. A standard approach [5] for investigating the correctness of computational software is to generate reference input data and output data $\langle \mathbf{x}, \mathbf{a} \rangle$, sometimes known as a reference pair, where \mathbf{x} and \mathbf{a} obey the exact mathematical relationship to a high degree of accuracy. The algorithm/software A under test will then be applied to the data \mathbf{x} to calculate a solution estimate $\hat{\mathbf{a}} = A(\mathbf{x})$. The difference between the reference solution \mathbf{a} and the computed solution $\hat{\mathbf{a}}$ is a measure of the fitness for purpose of the algorithm A .

The central issue to address in using this approach is the generation of the reference pairs $\langle \mathbf{x}, \mathbf{a} \rangle$. The standard approach is to develop a reference algorithm/software implementation, R , in which there is a high degree of confidence and then use the results $R(\mathbf{x})$ calculated by the reference software to create the reference pair $\langle \mathbf{x}, R(\mathbf{x}) \rangle$. This approach tends to be very expensive in terms of the resources required. There also remains the question of how to assess the accuracy or fitness for purpose of the reference algorithm. The data generator approach [3,5,6] starts instead

with the solution \mathbf{a} and then generates input data \mathbf{x} such that $\mathbf{a} = C(\mathbf{x})$ holds to a high degree of accuracy. The data generation problem can be considered as an inverse problem in which the inputs are generated in order to achieve a known output. For many computational problems, solving the inverse problem is much simpler. Often there is the added advantage that the correctness of reference data and results can be validated from first principles.

2. WHY COMPUTATIONAL SOFTWARE WILL NOT DETERMINE MATHEMATICALLY EXACT ANSWERS

There are a number of reasons why computational software will not return the mathematically exact answer. One immediately thinks of bugs in the software or other software engineering faults. However, even software implementations that are bug-free will not return the exact answer. Often for complex calculations, an approximate method is implemented, based on a linearised model, for example. For some types of input data, these approximate methods may be sufficiently accurate, but provide poorer results for other types of input data.

Many computational problems in coordinate metrology involve the solution of a set of nonlinear equations to which iterative techniques are applied. These algorithms require convergence tolerances to be set in order to assess when the iterative algorithm has converged to the solution. The accuracy of the computed solution depends directly on the assigned tolerances. Again tolerances that are appropriate for one type of input data may be less appropriate for another.

Most iterative algorithms used to solve a nonlinear problem are only guaranteed to find a local optimum [7], which may or may not be the global optimum. In general it is very difficult to construct algorithms that can find the global optimum. This problem applies particularly to the case of geometric form and tolerance assessment according to the Chebyshev and related criteria, for it is known that there can be a number of local minima close to each other in parameter space. For example, in flatness assessment it is possible to generate (somewhat artificial) data sets with m data points for which there are of the order of m global minima.

Finally, computations are performed in finite precision and the input data and output results are represented in finite precision. For most computational problems, it is impossible

to represent a mathematically exact reference pair in finite precision.

3. NUMERICAL ACCURACY OF REFERENCE PAIRS

The accuracy of a reference pair (\mathbf{x}, \mathbf{a}) can be quantified in a number of ways. The forward accuracy, D_F , is such that $D_F^2 = \Delta \mathbf{a}^T M_F \Delta \mathbf{a}$, where $\Delta \mathbf{a} = \mathbf{a} - C(\mathbf{x})$, and M_F is a symmetric positive definite defining a norm on the solution space. This measure depends on the distance from the computed solution to the mathematically correct solution for the input data \mathbf{x} . A second measure is the inverse or backward accuracy D_I which is such that

$$D_I^2 = \min_{\Delta \mathbf{x}} \Delta \mathbf{x}^T M_I \Delta \mathbf{x}$$

subject to the constraint $\mathbf{a} = C(\mathbf{x} + \Delta \mathbf{x})$, where M_I is a symmetric positive definite matrix. This measure depends on the amount required to perturb the input data in order that the computed solution is the mathematically exact solution for the perturbed data. It is possible to combine these measures in a hybrid measure D_H given by

$$D_H^2 = \min_{\Delta \mathbf{x}, \Delta \mathbf{a}} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{a} \end{bmatrix}^T M_H \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{a} \end{bmatrix}$$

subject to the constraint $\mathbf{a} + \Delta \mathbf{a} = C(\mathbf{x} + \Delta \mathbf{x})$.

4. LEAST SQUARES ORTHOGONAL DISTANCE REGRESSION

Let $\mathbf{u} \mapsto \mathbf{f}(\mathbf{u}, \mathbf{b})$ define a parametric surface where $\mathbf{u} = (u, v)^T$ are the footpoint or patch parameters and $\mathbf{b} = (b_1, \dots, b_n)^T$ are the parameters defining the position and shape of the surface. Given a set of coordinate data $X = \{\mathbf{x}_i, i = 1, \dots, m\}$, let $d(\mathbf{x}_i, \mathbf{b}) = \min_{\mathbf{u}} \|\mathbf{x}_i - \mathbf{f}(\mathbf{u}, \mathbf{b})\|$ be the orthogonal distance from the point \mathbf{x}_i to the surface defined by parameters \mathbf{b} . If \mathbf{u}_i^* is the optimal set of footpoint parameters and \mathbf{n}_i is the normal vector to the surface at $\mathbf{f}(\mathbf{u}_i^*, \mathbf{b})$, then

$$d(\mathbf{x}_i, \mathbf{b}) = (\mathbf{x}_i - \mathbf{f}(\mathbf{u}_i^*, \mathbf{b}))^T \mathbf{n}_i$$

and

$$\frac{\partial d}{\partial b_j}(\mathbf{x}_i, \mathbf{b}) = - \left(\frac{\partial \mathbf{f}}{\partial b_j} \right)^T \mathbf{n}_i$$

evaluated at \mathbf{u}_i^* .

In least squares orthogonal distance regression (LSODR [3,7]), the best-fit surface to the data set X solves

$$\min_{\mathbf{b}} \sum_{i=1}^m d^2(\mathbf{x}_i, \mathbf{b}).$$

4.1 Data generation for LSODR

If J is the Jacobian matrix of partial derivatives $\partial d_i / \partial b_j$, then the first order optimality conditions for \mathbf{b} to be a solution of the LSODR problem are that $J^T \mathbf{d} = \mathbf{0}$, i.e.,

$\mathbf{d} = (d_1, \dots, d_m)^T$ is in the null space of J . Let

$$X^* = \{\mathbf{x}_i^*, i = 1, \dots, m\}$$

with $\mathbf{x}_i^* = \mathbf{f}(\mathbf{u}_i^*, \mathbf{b})$ lying exactly on the surface $\mathbf{u} \mapsto \mathbf{f}(\mathbf{u}, \mathbf{b})$, J the Jacobian matrix evaluated at X^* and \mathbf{n}_i^* the normal vector to the surface at \mathbf{x}_i^* . If

$$X = \{\mathbf{x}_i^* + e_i \mathbf{n}_i^*, i = 1, \dots, m\}$$

represents a perturbation of X^* normal to the surface then the Jacobian matrix associated with X is also J . Furthermore, $d(\mathbf{x}_i, \mathbf{b}) = e_i$, so \mathbf{b} will satisfy the optimality conditions for X so long as $J^T \mathbf{e} = \mathbf{0}$, $\mathbf{e} = (e_1, \dots, e_m)^T$. The problem of data generation to reduce to that of determining a vector in the null space of a matrix [3,6,9]. Standard techniques involving orthogonal factorisation [10] can be used to achieve this. If the Jacobian matrix is factored as

$$J = QR = [Q_1 \ Q_2] \begin{bmatrix} R_1 \\ \mathbf{0} \end{bmatrix} = Q_1 R_1, \quad (1)$$

where Q is an orthogonal matrix and R is an upper-triangular matrix, then for any $(m - n)$ vector $\tilde{\mathbf{e}}$, $\mathbf{e} = Q_2 \tilde{\mathbf{e}}$ is such that $J^T \mathbf{e} = \mathbf{0}$. Conversely, if $J^T \mathbf{e} = \mathbf{0}$, then $\mathbf{e} = Q_2 \tilde{\mathbf{e}}$, where $\tilde{\mathbf{e}} = Q_2^T \mathbf{e}$. For large data sets, it is necessary to represent the orthogonal matrices compactly in terms of Householder transformations, for example [10]. Note that for any \mathbf{r} , if \mathbf{p} solves the linear least squares system $J\mathbf{p} \approx \mathbf{r}$, then

$$\mathbf{e} = \mathbf{r} - J\mathbf{p}$$

is in the null space of J .

4.2 Data sets for fixed shapes

Many fitting problems involve fitting a fixed shape, defined by a CAD file, to point cloud data. In this case, the optimisation parameters only involve three translation parameters, three rotation parameters and (optionally) one scale parameter. If the shape has translational or rotational symmetry then one or more transformation parameters have to be fixed. For example, a cylinder has one translational and one rotational symmetry so that only four transformation parameters are required. For the general case, given \mathbf{x}_i^* lying exactly on the fixed parametric surface with associated normal vector \mathbf{n}_i^* , the corresponding row of the Jacobian matrix is

$$[-\mathbf{n}_i^* \quad \mathbf{x}_i^* \times \mathbf{n}_i^* \quad -(\mathbf{x}_i^*)^T \mathbf{n}_i^*].$$

This means that the data generation problem is defined by \mathbf{x}_i^* and \mathbf{n}_i^* alone and that no information about the surface, however complex it may be, is required.

Figure 1 shows a 2 dimensional example of this approach for the case of fitting a circle to data. In this case, only two translational parameters are involved. The data points, dots, are generated such that the least squares best-fit circle to the data is the circle centred at the origin with radius 100 mm, solid curve.

4.3 Customised data sets for LSODR

Given a perturbation vector \mathbf{d}_0 representing a particular type of form error, for example, we can find the vector in the null space closest to \mathbf{d}_0 by solving

$$\min_{\mathbf{d}} \|\mathbf{d} - \mathbf{d}_0\|^2$$

subject to $J^T \mathbf{d} = \mathbf{0}$, a linear least squares problem subject to linear equality constraints. Again this problem can be solved using standard orthogonal factorisation techniques. If J has factorisation as in expression (1), then the solution \mathbf{d} is given by $\mathbf{d} = Q_2 Q_2^T \mathbf{d}_0$, the projection of \mathbf{d}_0 onto the null space of J . Again, for large data sets it is important that the calculations involving the orthogonal matrix are performed using a compact representation of the matrix [10].

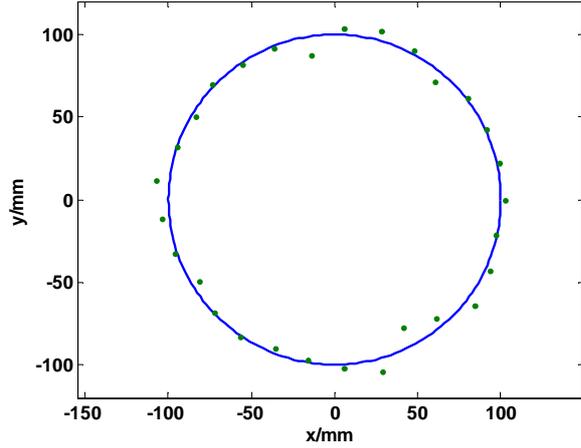


Figure 1. Data generated using the null space approach for a least squares circle fit to data.

Figure 2 shows a 2 dimensional example of this approach for the case of fitting a circle to data. The data points, dots, are generated such that the least squares best-fit circle to the data is the circle centred at the origin with radius 100 mm, solid curve, and mimic a three-lobed form error.

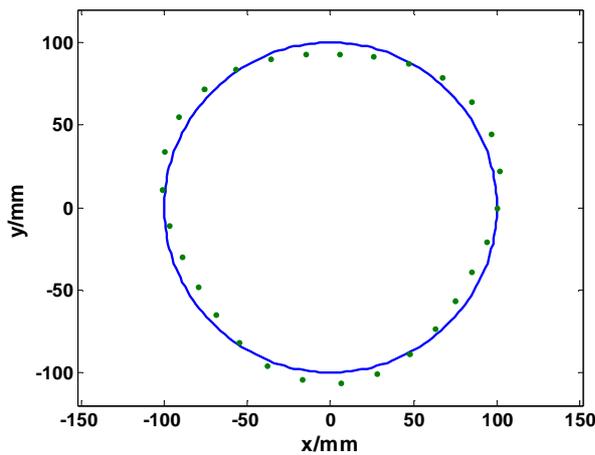


Figure 2. Null space data generation for a least squares circle fit, simulating a three-lobed form error.

4.4 Data sets modelling systematic effects associated with measurement systems

The data gathered by a coordinate measuring machine (CMM) can be modelled as

$$\mathbf{x} = \mathbf{x}^* + \mathbf{s}(\mathbf{x}^*, \mathbf{c}) + \mathbf{e},$$

where \mathbf{x} are the measured coordinates, \mathbf{x}^* , are the true coordinates, $\mathbf{s}(\mathbf{x}^*, \mathbf{c})$ models the systematic error behaviour of the coordinate measuring system, e.g., the kinematic errors of a CMM, and depends on parameters \mathbf{c} , and \mathbf{e} is a random effect. Similar to the case of form error, data sets can be generated that mimic systematic error behaviour associated with measurement systems. Figure 3 shows the null space data generation of a least squares circle fit mimicking a 2 dimensional coordinate measuring system with scale errors along each of the two axes and a squareness error [11].

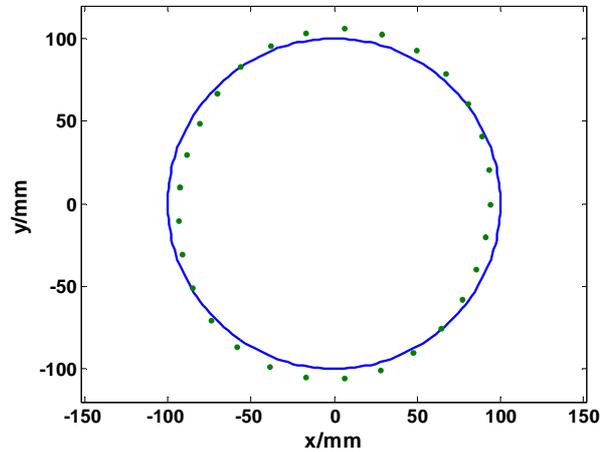


Figure 3. Null space data generation for a least squares circle fit, simulating scale and squareness errors associated with the coordinate measuring system.

4.5 Data sets with pre-assigned spatially correlated form error

Figures 2 and 3 concerning the generation of data mimicking form error or systematic behaviour associated with a coordinate measuring system are based on functional models in which the required deviation is given as a function of location on the surface. A more general approach is to specify the behaviour in terms of spatial correlation, reflecting the fact that nearby points on a surface will, in general, be associated with similar errors [12,13]. One approach is as follows. Let e and e' be the error at locations \mathbf{x} and \mathbf{x}' , respectively, that are correlated according to $\text{cov}(e, e') = k(\mathbf{x}, \mathbf{x}')$ where the kernel k is given by

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp \left\{ -\frac{1}{2\lambda^2} (\mathbf{x} - \mathbf{x}')^T (\mathbf{x} - \mathbf{x}') \right\}.$$

The parameter λ defines the length scale over which the errors are correlated while σ^2 is the variance associated with the errors at any point on the surface. Given a set of points

$X^* = \{\mathbf{x}_i^*, i = 1, \dots, m\}$ lying exactly on a surface, we evaluate the $m \times m$ variance matrix V with $V_{ij} = k(\mathbf{x}_i^*, \mathbf{x}_j^*)$. If V has Cholesky factorisation [10] $V = LL^T$, then if \mathbf{r} is an m -vector of random samples from a standard normal distribution, $\tilde{\mathbf{e}} = L\mathbf{r}$ is a random sample with variance matrix V . If the Jacobian matrix J is factored as in (1), then $\mathbf{e} = Q_2 Q_2^T \tilde{\mathbf{e}}$ is in the null space of J and represents errors reflecting the correlation structure.

Figure 4 shows the null space data generation for a least squares circle fit, with pre-assigned spatial correlation associated with the form error. Both figures 1 and figures 4 represent errors generated at random. However, the correlation structure associated with figure 4 ensures a smoother and more realistic departure from the ideal shape.

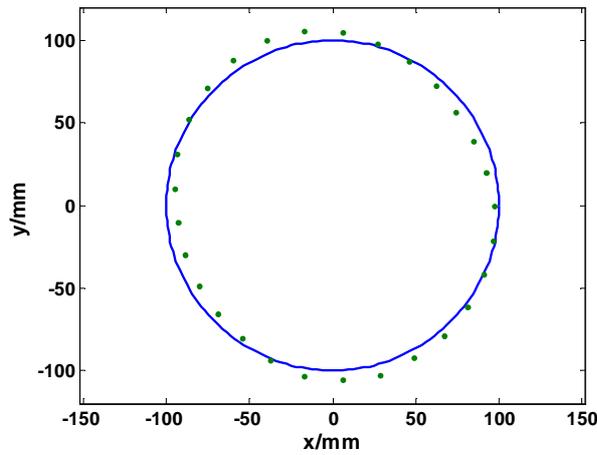


Figure 4. Null space data generation for a least squares circle fit, with pre-assigned space correlation associated with the form error.

In summary, the data generation problem for the nonlinear LSODR problem can be solved using simple linear computational techniques, and can be used flexibly to mimic different behaviours. We note that in figures 1 to 4, all data sets have the same least squares best-fit circle and all data sets give rise to the same value of the sum of squares of residual errors, i.e., represent deformations from the ideal geometry of the same magnitude.

5. CHEBYSHEV ORTHOGONAL DISTANCE REGRESSION

The Chebyshev orthogonal distance regression (ChODR) problem is also posed in terms of the distance function:

$$\min_{\mathbf{b}} \max_i |d(\mathbf{x}_i, \mathbf{b})|.$$

An equivalent formulation is

$$\min_{\mathbf{b}, e} e \text{ subject to } -e \leq d(\mathbf{x}_i, \mathbf{b}) \leq e, \forall i,$$

an optimisation problem involving a linear objective function and generally nonlinear inequality constraints. The parameter e represents the form error. As such, the ChODR problem is much more difficult to solve than its least squares counterpart. If there are n surface parameters \mathbf{b} , up to $n + 1$ constraints are active at the solution. In fact, there

can be more, but to first order they will be linearly dependent, a situation known as degeneracy, and one which has to be treated carefully by ChODR algorithms. Each inequality constraint defines a hypersurface in Euclidian $(n + 1)$ -space. If there are exactly $n + 1$ (non-degenerate) constraints active at the solution, the solution parameters are defined by first order conditions, the Kulesh-Kuhn-Tucker (KKT) conditions [3,7], involving the partial derivatives of the distance function. Geometrically, the solution is defined at the vertex of $n + 1$ intersecting hypersurfaces, and is referred to as a vertex solution. Otherwise, the optimality conditions for a solution depend additionally on second order information involving the second order partial derivatives of the distance functions. Such a solution is referred to as a non-vertex solution. Non-vertex solutions arise quite naturally. For example, the sphere of smallest radius enclosing a set of points can be defined by four points, three points (lying on a great circle) or two points (at the ends of a diameter).

5.1 Data generation for ChODR

The data generation problem for ChODR is also more complex than that for its least squares counterpart, and more research is required, in particular to guarantee a global minimum [3]. However, some progress has been made. Consider the problem of fitting a design surface (geometric element or CAD model) to a set of data points in which the only parameters to be optimised are six position parameters (three translation, three rotation) and one global scale parameter, seven in all. In this case, the first order optimality constraints for a non-degenerate vertex solution are as follows. There are eight constraints active at the solution. Let $I^+ = \{i: e = d(\mathbf{x}_i, \mathbf{b})\}$ and $I^- = \{i: -e = d(\mathbf{x}_i, \mathbf{b})\}$ and \mathbf{n}_i be the normal to the surface $\mathbf{u} \mapsto \mathbf{f}(\mathbf{u}, \mathbf{b})$ at \mathbf{x}_i . Then the KKT conditions are that there exist Lagrange multipliers $\lambda_i \geq 0$ such that

$$\sum_{i \in I^+ \cup I^-} \lambda_i = 1,$$

$$\sum_{i \in I^+} \lambda_i \mathbf{n}_i = \sum_{i \in I^-} \lambda_i \mathbf{n}_i,$$

$$\sum_{i \in I^+} \lambda_i \mathbf{x}_i \times \mathbf{n}_i = \sum_{i \in I^-} \lambda_i \mathbf{x}_i \times \mathbf{n}_i,$$

and

$$\sum_{i \in I^+} \lambda_i \mathbf{x}_i^T \mathbf{n}_i = \sum_{i \in I^-} \lambda_i \mathbf{x}_i^T \mathbf{n}_i.$$

These equations arise from considering the optimality conditions associated with the form parameter e , the translation parameters, the rotation parameters and the scale parameter, respectively. These conditions lead to the following simple scheme [2] for generating data with a known vertex (local) solution:

1. Determine eight points $\mathbf{x}_i^*, i \in I = \{1, \dots, 8\}$ lying exactly on the surface $\mathbf{u} \mapsto \mathbf{f}(\mathbf{u}, \mathbf{b})$ and corresponding normal vectors \mathbf{n}_i^* . Evaluate the 8×7 matrix J whose i th row is

$$[-\mathbf{x}_i^* \quad \mathbf{x}_i^* \times \mathbf{n}_i^* \quad -(\mathbf{x}_i^*)^T \mathbf{n}_i^*].$$

It is required that the points are chosen so that this matrix is full rank.

2. For all partitions $I = I^+ \cup I^-$, solve the KKT equations above for the λ . There will be two mutually anti-symmetric partitions for which the solution Lagrange multipliers are positive.
3. For such a partition $I = I^+ \cup I^-$ and $e > 0$, set $\mathbf{x}_i = \mathbf{x}_i^* + e\mathbf{n}_i^*$, $i \in I^+$, and $\mathbf{x}_i = \mathbf{x}_i^* - e\mathbf{n}_i^*$, $i \in I^-$.
4. Generate at random additional points \mathbf{x}_i^* on the surface and set $\mathbf{x}_i = \mathbf{x}_i^* + \delta_i \mathbf{n}_i^*$ where $\delta_i \in [-e, e]$ is chosen at random.

For sufficiently small e , \mathbf{b} defines a local solution for the ChODR problem.

The above scheme can be generalised to determining a vertex solution for the more general ChODR problem. For this case, the KKT first order conditions for a local minimum are given by $\lambda_i \geq 0$,

$$\sum_{i \in I^+ \cup I^-} \lambda_i = 1,$$

and

$$\sum_{i \in I^+} \lambda_i \nabla d(\mathbf{x}_i, \mathbf{b}) = \sum_{i \in I^-} \lambda_i \nabla d(\mathbf{x}_i, \mathbf{b}).$$

Given $(n + 1)$ points \mathbf{x}_i^* lying exactly on the surface, let J be the $(n + 1) \times n$ matrix of partial derivatives

$$J_{ij} = \frac{\partial d}{\partial b_j}(\mathbf{x}_i^*, \mathbf{b}).$$

Let P be an $(n + 1) \times (n + 1)$ diagonal matrix with ± 1 on the diagonal representing a partition of the points on to the outer and inner surfaces. We require a partition P such that the solution of $\begin{bmatrix} \mathbf{1}^T \\ J P \end{bmatrix} \lambda = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix}$ is such that $\lambda_i \geq 0$. Here, $\mathbf{1}^T$ is an $(n + 1)$ row vector of ones. Such a partition can be found by trying all $M = 2^n$ partitions. This task can be made more efficient by noticing that if Q is an orthogonal matrix such that $Q^T J$ is upper-triangular, then so is $Q^T J P$. Using this fact the required number of steps to search all partitions is $O(Mn^2)$ instead of $O(Mn^3)$.

5.2 Example: elliptic hyperboloid

An elliptic hyperboloid in standard position is given parametrically by

$$f(\mathbf{u}, \mathbf{s}) = \begin{bmatrix} s_1 \cos u \cosh v \\ s_2 \sin u \cosh v \\ s_3 \sinh v \end{bmatrix},$$

involving three shape/size parameters $\mathbf{s} = (s_1, s_2, s_3)^T$. Along with the six parameters determining the position of the surface, the surface is defined by nine parameters so that a vertex solution is defined by ten data points. Table 1 gives the coordinates of ten points that lie exactly on an elliptic hyperboloid with semi-axes $s_1 = 100$, $s_2 = 150$ and $s_3 = 200$. The final column shows which points are to be perturbed to lie on the outer or inner surfaces in order to define a vertex solution for the ChODR problem.

Table 1: Ten points that lie exactly on an elliptic hyperboloid that can be used to define a vertex solution for the ChODR problem.

x	y	z	+/-
95.5696	-98.8765	117.9606	1
11.0127	239.8539	250.5204	1
-111.4338	120.0531	187.8633	-1
-152.8815	-151.4192	307.0041	1
49.8298	-208.1489	216.6933	-1
145.6858	-85.9577	-240.9002	-1
72.1861	189.9424	-212.0901	-1
-182.4654	73.0065	-320.3903	1
-130.8710	-165.3982	-277.7460	-1
-26.4820	-221.0097	-222.8032	1

6. DISCUSSION

The null space approach for generating reference data for the LSODR problem is reasonably mature and effective. For the ChODR problem, only approaches to generating vertex solutions have been implemented. Some progress in generating non-vertex solutions has been made [3], but this problem still needs further research. A new European research project, Traceability of Computationally-Intensive Metrology, involving a number of European national metrology institutes and universities, has been funded to undertake this research [14].

7. ACKNOWLEDGEMENTS

This work has been supported by the UK's National Measurement Office Mathematics and Modelling for Metrology programme and by funding from the European Union through the European Metrology Research Programme (EMRP) Joint Research Project NEW06, Traceability in Computationally-Intensive Metrology. The EMRP is jointly funded by the EMRP participating countries within EURAMET and the European Union. We are grateful to colleague Dr Ian Smith for comments on drafts of this paper.

8. REFERENCES

- [1] K Carr and P Ferreira, Verification of form tolerances, part I: basic issues, flatness and straightness. *Prec. Eng.* 17(2), 131-143, 1995.
- [2] ISO 1101, Geometric Product Specifications (GPS) Geometric tolerancing - tolerances of form, orientation, location and run out. ISO, Geneva, 2005.
- [3] A B Forbes and H D Minh, Form assessment in coordinate metrology, in *Approximation Algorithms for Complex Systems*, E H Georgoulis et al., eds., Springer-Verlag, Heidelberg, 69-90, 2011.
- [4] ISO 10360-6, Geometric Product Specification (GPS), Estimation of error in computing Gaussian associated features. ISO, Geneva, 2001.
- [5] B P Butler et al., A methodology for testing the numerical correctness of approximation and optimisation software, in *The Quality of Numerical Software: Assessment and Enhancement*, 138-151, Chapman & Hall, London, 1997.
- [6] M G Cox and A B Forbes, Strategies for testing form assessment software, NPL Report DITC 211/92, National Physical Laboratory, Teddington, 1992.

- [7] P Gill, W Murray and M Wright, *Practical Optimization*, Academic Press, London, 1991.
- [8] P T Boggs et al., A stable and efficient algorithm for nonlinear orthogonal distance regression, *SIAM Sci. Comp.* 8(6), 1052-1078, 1987.
- [9] A B Forbes, P M Harris and I M Smith, Correctness of free form surface fitting software, in *Laser Metrology and Machine Performance VI*, D G Ford, ed., 263-272, WIT Press, Southampton, 2003.
- [10] G H Golub and C F Van Loan. *Matrix Computations*, 3rd edn., John Hopkins University Press, Baltimore, 1996.
- [11] A B Forbes, Surface fitting taking into account uncertainty structure in coordinate data, *Meas. Sci. Tech.*, 17:553-558, 2006.
- [12] C E Rasmussen and C K I Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, 2006.
- [13] P J Scott and A B Forbes, Mathematics for modern precision engineering, *Phil. Trans. Roy. Soc. A*, to appear, doi:10.1098/rsta.2011.0379, 2012.
- [14] http://www.euramet.org/index.php?id=emrp_call_2011