Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 00 (2016) 000–000

www.elsevier.com/locate/procedia

14th CIRP Conference on Computer Aided Tolerancing (CAT)

Theory and algorithm for planar datum establishment using constrained total least-squares

Craig M. Shakarji[a], Vijay Srinivasan[b]

[a]*Physical Measurement Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, craig.shakarji@nist.gov*
[b]*Engineering Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, vijay.srinivasan@nist.gov*

Abstract

First, we present an efficient algorithm for establishing planar datums that is based on a constrained minimization search based on the L2 norm after forming a convex surface from sampled points. Visualized by Gauss maps, we prove that the problem reduces to a minimization search where the global minimum is localized about the minimizing facet. Second, we highlight advantages of this planar datum, including the major advantage that the datum planes have full mechanical contact with the datum features in stable cases yet are automatically balanced for rocking conditions. These advantages make this definition appealing for standardization.

Peer-review under responsibility of the organizing committee of the 14th CIRP Conference on Computer Aided Tolerancing.

Keywords: constrained least squares; constrained optimization; convex hull; datum; Gauss map; least squares; fitting; optimization; planar datum; singular value decomposition; total least squares

1. Introduction

In the world of Geometric Dimensioning and Tolerancing (GD&T), datums are used extensively to locate and orient tolerance zones [1-7]. Datum planes in particular are common and are established by mating planes to imperfect datum features on parts during inspection [3] (see Fig. 1). Distances and orientations on drawings and three-dimensional models are established from these datum planes, relative to which tolerance zones are located and oriented. Additional details of the importance and prevalence of datum planes in specifications are given in [8] and will not be revisited in this paper.

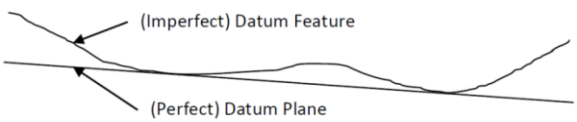

Fig. 1. Deriving a datum plane from a datum feature.

Given that datum planes are ubiquitous, it might be surprising that—short of standardization—there are several different yet reasonable approaches by which a datum plane can be established from a datum feature [9]. Furthermore, the International Organization for Standardization (ISO) and the American Society for Mechanical Engineering (ASME) are actively working to establish default datum plane definitions. In [10] we introduced a definition for a planar datum that naturally combines a correspondence to physical, surface plate mating (i.e., "high points") but with automatic balancing in the case of unstable, rocking conditions. The datum plane definition is based on a constrained total least-squares criterion (abbreviated here as L2C), which is explored in this paper. This should not be confused with an unconstrained total least-squares fit that is shifted out of the material.

Given a set of points sampled on a datum feature, the two major steps in establishing the L2C datum plane are as follows:

1) Compute the "lower" convex envelope of those points. This is the portion of the convex hull that lies on the nonmaterial side of the datum feature. In 3D, this convex envelope consists of a union of non-overlapping triangles, while in 2D it is a union of line segments creating a piecewise linear curve.
2) Find the plane, constrained to lie on the nonmaterial side of the computed convex surface that minimizes the integral of squared distances from that surface, namely $\int_S d^2(\boldsymbol{p}, P)ds$, where S is the convex surface and d is the distance from a point $\boldsymbol{p}$ on the surface to the

Peer-review under responsibility of the organizing committee of the 14th CIRP Conference on Computer Aided Tolerancing.

plane, P. If P contains $\boldsymbol{x}$ and has normal $\boldsymbol{a}$, then $d = \boldsymbol{a} \cdot (\boldsymbol{p} - \boldsymbol{x})$.

Concentrating on the second step, we find the need to integrate over a set of triangles (or line segments in 2D). For each triangle (or line segment) this integral can be replaced by the Simpson's rule approximation (see Fig. 2) [11] (which we will see is actually exact in our case).

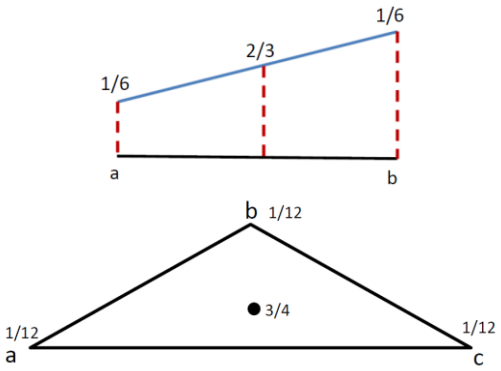

Fig. 2. The locations and weights for function evaluations for numerical integration using Simpson's rule over an interval and triangle.

Simpson's rule for integrating over an interval (or triangle for the 3D case) depends only on the weighted values of the function at the endpoints (or vertices in 3D) and at the centroid. Over an interval, Simpson's rule is given by:

$$\int_a^b f(x)dx \approx (b-a)\left(\frac{1}{6}f(a) + \frac{2}{3}f\left(\frac{a+b}{2}\right) + \frac{1}{6}f(b)\right),$$

and for integrating over a triangle, T, as shown in Fig. 2,

$$\int_T f(\boldsymbol{s})dT \approx$$

$$\text{Area}(T)\left(\frac{1}{12}f(a) + \frac{1}{12}f(b) + \frac{1}{12}f(c) + \frac{3}{4}f\left(\frac{a+b+c}{3}\right)\right).$$

Because Simpson's rule [11] is exact for functions of degree 2 (our case), we note that in the two formulas just above, these are exact calculations of the integrals and not mere approximations. The framing of this problem as a weighted sum-of-squares now allows us to solve the objective function as a singular value decomposition (SVD) problem. See [12] for a general treatment of using the SVD as a method for minimizing the total least-squares problem, and [13] for an application of it applied to planar fitting with weighted points (essential to be physically correct), which is our case here.

For the 3D case, let a S be a lower convex surface be made up of N triangles, T_1, T_2, …,T_N, where T_i has vertices (x_{iA}, y_{iA}, z_{iA}), (x_{iB}, y_{iB}, z_{iB}), and (x_{iC}, y_{iC}, z_{iC}) and where each triangle has centroid $(\bar{x}_i, \bar{y}_i, \bar{z}_i)$ and area A_i. If P is a candidate plane and, for each triangle, d_{iA}, d_{iB}, d_{iC} are the distances between P and the vertices and $\bar{d}_i$ is the distance from P to the triangle's centroid. Then, the L2C objective function to be minimized is:

$$\sum_{i=1}^{N} A_i \left(\frac{d_{iA}^2}{12} + \frac{d_{iB}^2}{12} + \frac{d_{iC}^2}{12} + \frac{3\bar{d}_i^2}{4}\right). \quad (1)$$

For the 2D case, where the convex surface is comprised of $N-1$ line segments, each having length L_i , endpoints (x_i, y_i), and (x_{i+1}, y_{i+1}), d_i being the distance from P to (x_i, y_i), and $\bar{d}_i$ is the distance from P to the line segment's midpoint, we then have the objective function being

$$\sum_{i=1}^{N-1} L_i \left(\frac{d_i^2}{6} + \frac{d_{i+1}^2}{6} + \frac{2\bar{d}_i^2}{3}\right). \quad (2)$$

In [10] we proved that the (2D) objective function for any candidate plane P is given by the elegant, efficient formula:

$$\sigma_1^2 \text{Cos}^2\theta + \sigma_2^2 \text{Sin}^2\theta + L d_c^2, \quad (3a)$$

or equivalently

$$\sigma_1^2 a^2 + \sigma_2^2 b^2 + L d_c^2, \quad (3b)$$

where (see Fig. 3) d_c is the distance from the plane P to the centroid, σ_1 and σ_2 are the singular values from the SVD of the matrix $\boldsymbol{M}$ below, and θ represents the angle P makes with the singular vector corresponding to the smallest singular value, σ_1 . Eqs. (3a) and (3b) are equivalent, where $(a, b) = (\text{Cos}\theta, \text{Sin}\theta)$ is the unit normal to the candidate plane when expressed as the dot product of that normal with each of the two singular vectors (e.g., a is the dot product of the unit normal to the plane with the first singular vector). The $3N \times 2$ matrix, $\boldsymbol{M}$, that is used in the SVD comes from the elements the Simpson's rule approximation (see [10] for more detail), repeated for each of the N line segments:

$$\boldsymbol{M} = \sqrt{\frac{1}{6}} \begin{bmatrix} \sqrt{L_1}(x_1) & \sqrt{L_1}(y_1) \\ 2\sqrt{L_1}\left(\frac{x_1+x_2}{2}\right) & 2\sqrt{L_1}\left(\frac{y_1+y_2}{2}\right) \\ \sqrt{L_1}(x_2) & \sqrt{L_1}(y_2) \\ \vdots & \vdots \\ \sqrt{L_N}(x_N) & \sqrt{L_N}(y_N) \\ 2\sqrt{L_N}\left(\frac{x_N+x_{N+1}}{2}\right) & 2\sqrt{L_N}\left(\frac{y_N+y_{N+1}}{2}\right) \\ \sqrt{L_N}(x_{N+1}) & \sqrt{L_N}(y_{N+1}) \end{bmatrix}$$

(The construction of $\boldsymbol{M}$ is done with the data translated so the centroid is at the origin. This translation is not shown explicitly in the matrix for reasons of space.)

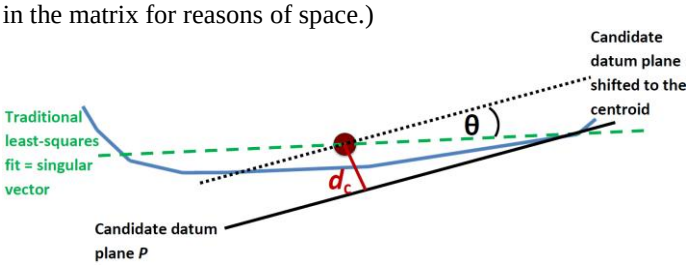

Fig. 3. The objective function for any candidate datum can be found simply by knowing the angle θ and distance d_c and using Eq. (3).

Using Eq. (3) to compute the objective function means that the SVD has to be computed only once, and its result can be applied to any given candidate datum plane. This makes for a much more efficient minimization algorithm.

What is fascinating about Eq. (3) is that the two terms on the left are exactly the objective function used in a traditional least-squares minimization while the term on the right is the objective function in a constrained L_1 fit [14, 15]. We will see that the objective function indeed does manifest itself as having the balancing property of the unconstrained least-squares and the full mechanical contact of the constrained L_1 definition, which is what is desired.

This can extend to 3D as well, since we showed that there is an extension of Simpson's rule that applies to integration over a triangular region. For the 3D case, the objective function for any candidate plane P is given by the efficient formula:

$$\sigma_1^2 a^2 + \sigma_2^2 b^2 + \sigma_3^2 c^2 + A d_c^2, \quad (4)$$

where d_c is the distance from the plane P to the centroid, σ_1, σ_2 and σ_3 are the singular values from the SVD of the matrix $\boldsymbol{M}$ below, and (a, b, c) is the unit normal to the candidate plane P when expressed as the dot product of that normal with each of the three singular vectors (e.g., a is the dot product of the unit normal to the plane with the first singular vector). Applying Simpson's rule for each of the N triangles, the $4N \times 3$ matrix $\boldsymbol{M}$ that is used in the SVD is:

$$\boldsymbol{M} = \sqrt{\frac{1}{12}} \begin{bmatrix} \sqrt{A_1}x_{1A} & \sqrt{A_1}y_{1A} & \sqrt{A_1}z_{1A} \\ \sqrt{A_1}x_{1B} & \sqrt{A_1}y_{1B} & \sqrt{A_1}z_{1B} \\ \sqrt{A_1}x_{1C} & \sqrt{A_1}y_{1C} & \sqrt{A_1}z_{1C} \\ 3\sqrt{A_1}\bar{x}_1 & 3\sqrt{A_1}\bar{y}_1 & 3\sqrt{A_1}\bar{z}_1 \\ \vdots & \vdots & \vdots \\ \sqrt{A_N}x_{NA} & \sqrt{A_N}y_{NA} & \sqrt{A_N}z_{NA} \\ \sqrt{A_N}x_{NB} & \sqrt{A_N}y_{NB} & \sqrt{A_N}z_{NB} \\ \sqrt{A_N}x_{NC} & \sqrt{A_N}y_{NC} & \sqrt{A_N}z_{NC} \\ 3\sqrt{A_N}\bar{x}_N & 3\sqrt{A_N}\bar{y}_N & 3\sqrt{A_N}\bar{z}_N \end{bmatrix}$$

(The construction of $\boldsymbol{M}$ is done with the data translated so the centroid is at the origin. This translation is not shown explicitly in the matrix for reasons of space.)

The notation used in showing $\boldsymbol{M}$ (just above) assumes the surface is comprised of N triangles T_i, each having area A_i and vertices (x_{iA}, y_{iA}, z_{iA}), (x_{iB}, y_{iB}, z_{iB}), and (x_{iC}, y_{iC}, z_{iC}), their average being $(\bar{x}_i, \bar{y}_i, \bar{z}_i)$.

We can summarize the 3D constrained L_2 algorithm as follows (the 2D case being similar): Given data points $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots, \boldsymbol{x}_M$, where each $\boldsymbol{x}_i = (x_i, y_i, z_i,)$, and a direction that indicates the direction into the material, then the datum plane is established using the following steps:

1) Compute the convex hull of the data points and represent it by the union of a set of triangles.
2) Select the N triangles (where $N < M$) that are exterior to the material.
3) Compute the centroid, $\bar{\boldsymbol{x}}$, of the convex surface of Step 2.
4) Construct the matrix $\boldsymbol{M}$ as defined above and compute its SVD to obtain the singular values σ_1, σ_2, and σ_3 and their corresponding singular vectors.
5) The objective function can now be constructed by Eq (4) and used to find the optimal plane.

2. Gauss maps and convexity

The procedure to accomplish the optimization in Step 5 (just above) is not obvious. This section will use Gauss maps to describe the nature of the objective function, which will drive our choice of method to search for the optimal plane. In 2D, the search is for the optimal line with only one degree of freedom, namely the angle of the line (Fig. 4).

Thus we can envision a candidate datum line rolling (with increasing angle, as pictured in Fig. 4) from the left to the right, contacting different points and edges along the piecewise-linear curve. We note that the "rolling candidate line" will contact each vertex of the curve for some finite time, and coincide with each edge for only an instant before the point of contact shifts to the next vertex. This can be viewed as a Gauss map as in Fig. 5.

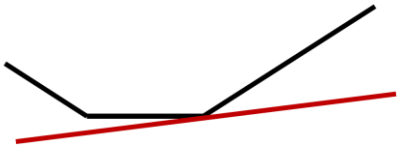

Fig. 4. A candidate datum is defined by its angle alone. Its location is automatically determined to just contact the curve.

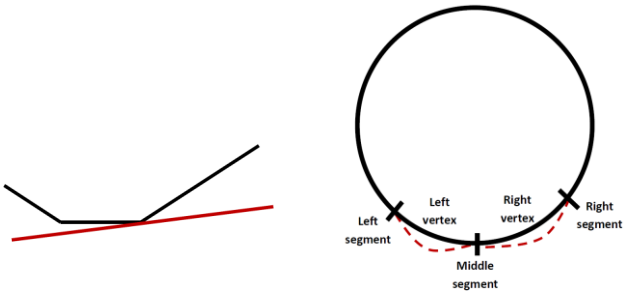

Fig. 5. (a) a rolling candidate line; (b) a 2D Gauss map showing a (dashed) example of a composite elliptical shape.

In this view, one can see that an edge on the curve corresponds to a point on the circle (Gauss map) and a vertex on the curve corresponds to an arc on the circle.

The objective function, when superimposed on the Gauss map, would be a composite elliptical shape. That is, the image of each arc on the circle would correspond to a part of an ellipse. (Note: the dashed curves in Fig. 5(b) show an example of a composite elliptical shape. It is not meant to correspond to the exact composite elliptical shape arising from the fig 5(a) to its left)

In 3D, there are two angular degrees of freedom in the optimization search. This can be visualized on a Gauss map using a sphere. The set of triangles that makeup the convex surface includes faces, edges, and vertices. The correspondences to the Gauss map are: triangular faces on the convex surface correspond to points on the sphere, edges on the convex surface correspond to edges on the sphere, and vertices on the convex surface correspond to (somewhat triangular) patches on the sphere. The objective function superimposed on the Gauss map forms a composite ellipsoidal shape, the 3D equivalent of the 2D case.

Realizing that the objective function has such a composite ellipsoidal/elliptical shape paves the way for the means to prove that the objective function is convex, a fact that is extremely helpful in creating an algorithm to efficiently find the global minimum.

3. The objective function is convex over the relevant search region.

We now give an outline of the proof that the objective function is convex with respect to any reasonable range of candidate orientations. Here, convexity is not a mere detail of technical interest but one we identify as a key accomplishment

of this paper. Assurance of convexity is powerful, in that it allows for a much faster (and, in our case, non-iterative) solution to find the datum plane. Convexity of the objective function is not obvious, since the term Ld_c^2 that appears on the right of Eq (3) is not convex with respect to θ. Here we understand the objective function to be dependent solely on the candidate plane's orientation—its location is understood to be always just-contacting the convex surface.

We will first consider the 2D case, along with the assumption that the datum feature, and the discrete points arising from it, are approximately planar. (And thus the convex surface arising from the points is approximately planar.) This assumption is reasonable, since planar datums are nominally planes that typically have form deviations that are orders of magnitude smaller than their size. Even if the form deviation were, say, 10 % or 20 % of the extent of the planar patch (which would be an extremely large relative form error) the proof still holds, which is outlined in the following two steps.

Step 1: For each fixed vertex of the convex surface, the objective function (which is solely a function of the orientation, θ) is convex over each interval of θ that represents the rolling of the contacting plane about that vertex.

One way to see this is that we know the Gauss map of the objective function over the circular arc is an ellipse. This is shown in [16], which states that any linear transformation applied to the unit circle yields an image that is an ellipse. The size, shape, and orientation of the ellipse can be seen by the observing the SVD of the linear transformation matrix. Since the surface is nominally planar, the shape of the ellipse is predictably oriented and elongated similar to that as shown in Fig. 6. (Typically the elongation will be much more extreme than that shown.) Since the curvature of the ellipse between between θ_1 and θ_2 is sufficiently small, it is clear that a plot of the radial value of the ellipse between θ_1 and θ_2 is convex, as depicted in Fig. 6. The objective function is the square of the function shown in Fig. 7, but we note that the square of any nonnegative, convex function is also convex.

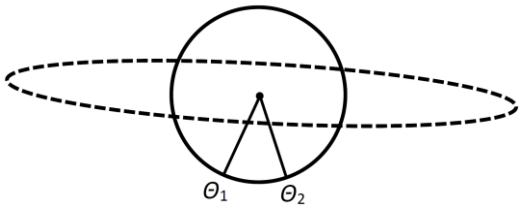

Fig. 6. A unit circle with its elliptical image.

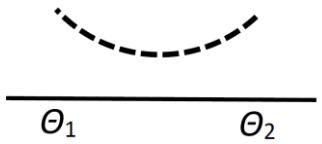

Fig. 7. The polar plot from θ_1 to θ_2 from Fig. 6 when expressed as a function of θ in a Cartesian graph.

We note that θ_1 and θ_2 would be limited, if needed, to conform to a reasonable range of candidate orientations.

Another way of demonstrating Step 1 is to observe that the objective function for a plane passing through a fixed vertex is

$$\sigma_1^2 \mathrm{Cos}^2\hat{\theta} + \sigma_2^2 \mathrm{Sin}^2\hat{\theta}, \quad (5)$$

where σ_1 and σ_2 are the singular values from the SVD of the matrix $\widehat{\boldsymbol{M}}$ (which is $\boldsymbol{M}$ defined above, but with the data points shifted so that the vertex under consideration is the origin). $\hat{\theta}$ represents the angle P makes with the singular vector corresponding to the smallest singular value, σ_1, which is also the direction of the least-squares line constrained to pass through the vertex (Fig. 8). Since the second derivative of the function is a constant times $\mathrm{Cos}(2\hat{\theta})$, and since a function is convex over the region that its second derivative is nonnegative, this function is convex for all angles, $\hat{\theta}$, between $-45°$ and $+45°$. This requirement is easily met under our assumption that the surface be somewhat planar.

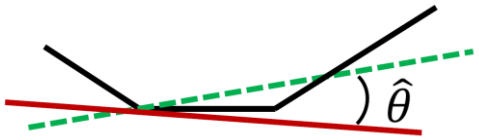

Fig. 8 The dashed line is the unconstrained least-squares fit to the piecewise linear curve, which is the basis from which the angle is measured for Eq. (5).

We note that it is indeed true that for the ends of the convex surface, a steep edge can exist, but we are only seeking to show the objective function is convex in the reasonable search range of values of θ, which excludes those extreme angles.

Step 2: The convexity from one piece of the graph to the next is preserved. Outline of proof of Step 2: The issue to be proven here is illustrated in Fig. 9. Depending on how the two functions come together determines whether convexity is preserved or broken.

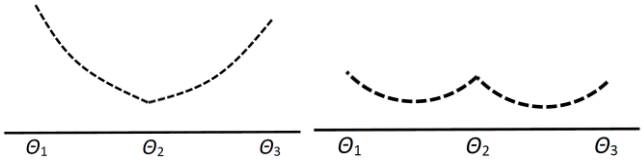

Fig. 9. (a) two convex functions over adjacent intervals result in a single convex function over the entire interval; (b) two convex functions over adjacent intervals result in a single nonconvex function over the interval.

Convexity can be proved by showing the first derivative is nondecreasing. Therefore we can prove that any two adjacent parts of the objective function come together in a manner like Fig. 9(a) rather than Fig 9(b) by comparing the derivative from the left with the derivative from the right. The derivative on the left equals the derivative on the right when observing the first two terms (on the left) in Eq (3a). However the third term (on the right) is different, since the vertex about which the candidate line rotates changes.

Three cases exist: In Case 1, as θ increases such that the candidate datum plane (line in 2D) rotates about vertex A, approaching the line segment, d_c (the distance to the centroid) is decreasing, see Fig. 10(a). Once θ increases past the line segment so that it is contacting and rotating about vertex B, d_c is increasing. Hence the derivative of the objective function is increasing through the transition from one piece to the next.

In Case 2, as θ increases such that the candidate datum plane (line in 2D) rotates about vertex A, approaching the line segment, d_c is decreasing, see Fig. 10(b). Once θ increases past the line segment so that it is contacting and rotating about

vertex B, d_c is still decreasing but at a slower rate (seen by the lower "leverage" due to a closer fulcrum). Hence the derivative of the objective function is increasing through the transition from one piece to the next.

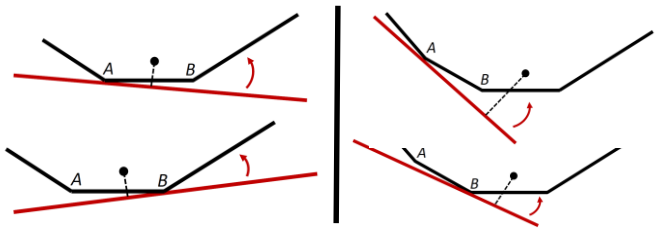

Fig. 10. As the point of contact shifts from A to B, (a) the distance to the centroid is decreasing then increasing, (b) the distance to the centroid is decreases more slowly (with respect to the angle).

Case 3 (not pictured, but somewhat like a mirror image of Fig. 11) is like case 2, but d_c is increasing in both cases, but increases at a faster rate after the transition from vertex A to vertex B.

While the 3D case is certainly more complicated and is not put in writing in this paper, there is nothing fundamentally different in extending Steps 1 and 2 to demonstrate convexity in that case as well with appropriate changes (e.g., 2D ellipses become 3D ellipsoids).

4. Convexity of the objective function leads to an efficient algorithm.

The fact that we can rely on the objective function to be convex has powerful implications for the efficiency of our fitting algorithm. In particular, we will show that convexity allows us to search for the minimizing facet and then simply test the boundaries of that minimizing facet for a global solution. Convexity assures that such a search captures the global minimum and does not miss some hidden minimum elsewhere.

Before giving an efficient, non-iterative algorithm, we note that the solution could be achieved by beginning with the orientation obtained using unconstrained total least-squares a starting orientation, and applying an iterative "downhill" minimization algorithm to the objective function as given in Eq. (1) (for 3D) or Eq. (2) (for 2D). For each candidate orientation, the candidate plane (or line in 2D) would be the one that just contacts the surface. Convexity assures that there is no risk of obtaining a local but not global minimum. While this may not be the most elegant approach, we mention it because it may be the simplest to code, which is desirable in some situations.

But convexity can also be used to create an efficient and elegant solution. One can perform an SVD to get the objective function in the form of Eq. (3) (for 2D) or Eq. (4) (for 3D) and use it to quickly compute the objective function for every triangular facet (or line segment in 2D).

In the 2D case, this step can be followed by checking the endpoints of the minimizing line segment to see if there exists a line passing through either endpoint that has a lower objective function and lies outside the material, see Fig. 11(a). This step can be achieved by—for each of the two endpoints—computing the SVD of the matrix matrix $\widehat{\boldsymbol{M}}$ (which is $\boldsymbol{M}$ defined above, but with the data points shifted so that the endpoint under consideration is the origin). Form the line passing through the endpoint and oriented in the direction of the singular vector corresponding to the smallest singular value. Convexity allows us to know that if that line lies outside the material, then it is the line that minimizes the objective function.

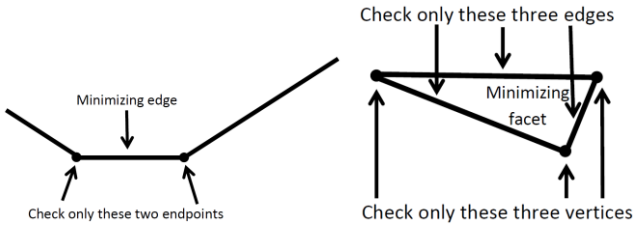

Fig. 11. (a) the minimizing line will coincide with the minimizing edge or will balance on one of the two adjacent vertices;. (b) the minimizing plane will be coincident with the minimizing facet or balance on one of the adjacent edges or vertices. The triangle shown is one of a collection of triangles (not shown) that make up the convex surface, but only this triangle's edges and vertices need to be checked.

In 3D one can compute the centroid, shift to it, and then compute the SVD to achieve the objective function formula (Eq. (4)) for any candidate plane. Then the objective function can be computed for each triangle of the convex surface. For each triangle, this task is only a matter of evaluating Eq (4). As in the 2D case, only one SVD needs to be performed in order to gain the objective function values for all the triangles. The triangle corresponding to the minimum objective function can then be identified. Then the vertices and edges of that triangle can be checked to see if a plane passing through any of them gives a lower objective function, see Fig. 11(b). The vertices can be checked using the 3D equivalent of the method described above in the 2D case. Each edge can be tested similarly, by rotating the data points such that the edge coincides with the Z-axis, and reducing the problem to a 2D one.

5. Advantages of the constrained least-squares datum

This L2C datum definition with the algorithm shown here has the following advantages:

- The most complex mathematical tools required are a convex hull algorithm and SVD. Both of these are well studied, reliable, and available.
- It can be performed efficiently. In fact, the limiting factor is the convex hull itself for which algorithms exist of time order $n\log(n)$, where n is the number of original points.
- It is not adversely affected by unevenly sampled data points as some datum definitions are.

But the most notable advantage is the remarkable ability of the L2C datum definition that the datum makes full contact with the datum feature when it is stable to do so, and balances rocking conditions when there is instability that requires it.

Figure 12 shows two typical cases where, on the left, one would seek to balance the rocking condition, and on the right, one would seek for the datum plane to be stably flush with the edge of the datum feature. This is what the L2C solution does automatically.

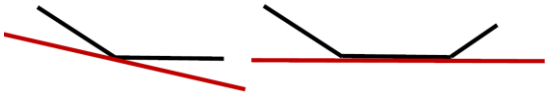

Fig. 12. Two typical cases of datum features with the associated L2C datums shown. The balanced rocking case is on the left and the stable, flush case is on the right.

For the rocker condition pictured on the left side of Fig. 12, if the line segment on the right were made longer, the L2C datum plane would roll to the right smoothly. For the stable case pictured on the right side of Fig. 12, if the line segment on the right were made somewhat longer, the L2C datum plane would not move from its stable state. It would remain flush with the edge of the datum feature until the line segment on the right grew long enough to make a rocker condition, at which point the L2C datum would smoothly begin to roll to the right to balance the rocker.

In contrast, the shifted least-squares solution would achieve a flush mating with the datum feature (as pictured on the right of Fig. 12) for only an instant. That is, as the line segment on the right began to be extended, there would only be one length that resulted in a flush mating. This contrast shows the fascinating feature of the L2C, which stays flush with the datum feature—even while the line segment extends—until it reaches such a length that a rocking condition exists, like shown in Fig. 13.

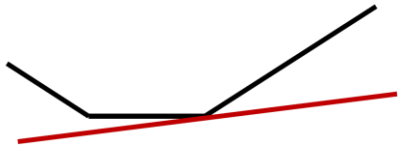

Fig. 13. The line segment on the right is long enough for the constrained L_2 datum to treat it as a rocking condition and separate from the flush contact it had in the right hand picture of Fig. 12.

6. Implementation

The L2C datum definition has been coded and run under various input data set scenarios. The results are that the theory does in fact hold. For 2D, this means that the datum line contacts two points in sufficiently stable cases, and contacts one point when there is a rocking condition, which it appropriately balances. In 3D, the datum plane does, in fact, contact three points in sufficiently stable cases, and contacts two points along an edge, when there is a rocking condition along that edge (which it balances), and contacts one point when there is a rocking condition on that point (which is balanced by this datum plane definition).

7. Conclusion

The L2C datum plane definition automatically shifts between a full-contact solution to stabilizing rocker conditions. Besides other advantages, the definition is robust and, because the nature of the objective function has been investigated in this paper (and in particular because it is convex over the region of interest) reliable, efficient algorithms are available. The mathematical tools required to carry out implementation are reliable and available. Based on all these, the L2C is an attractive choice for standardization of planar datums.

References

[1] Srinivasan, V., “Reflections on the role of science in the evolution of dimensioning and tolerancing standards,” Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, Vol. 227, No. 1, pp. 3-11, 2013. DOI: 10.1177/0954405412464012

[2] Tandler, W. “All Those Datum Things” Inside Metrology, Quality Digest, Quality Digest Magazine, 13 February 2008.

[3] Tandler, W. “Establishing Datum Reference Frames,” Inside Metrology, Quality Digest, 12 March 2008.

[4] ANSI/ASME Y14.5.1M-2009 “Dimensioning and Tolerancing,” The American Society of Mechanical Engineers, New York.

[5] ANSI/ASME Y14.5.1M-1994 “Dimensioning and Tolerancing,” The American Society of Mechanical Engineers, New York.

[6] ISO 5459:2011. “Geometrical product specifications (GPS)—geometrical tolerancing—datums and datum systems.” Geneva: International Organization for Standardization, 2011.

[7] Zhang, Xuzeng, and Roy, Utpal “Criteria for establishing datums in manufactured parts” Journal of Manufacturing Systems, 12(1), pp 36–50, 1993.

[8] Shakarji, C. M., and Srinivasan V., “Theory and Algorithms for L1 Fitting Used for Planar Datum Establishment in Support of Tolerancing Standards,” DETC2013-12372, Proceedings, ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Portland, OR, 2013.

[9] Hopp, T. H., 1990, “The Mathematics of Datums,” ASPE Newsletter, September 1990, American Society for Precision Engineering, Raleigh, NC

[10] Shakarji, C. M., and Srinivasan V., “A Constrained L2 Based Algorithm for Standardized Planar Datum Establishment,” IMECE2015-51199, Proceedings, ASME 2015 International Mechanical Engineering Congress & Exposition, Houston, TX, 2015.

[11] Horowitz, A., "A version of Simpson’s rule for multiple integrals," Journal of Computational and Applied Mathematics 134 (2001) 1–11.

[12] VanHuffel, S., and Vandervalle, J., 1991 The Total Least Squares Problem: Computational Aspects and Analysis, SIAM, Philadelphia, PA.

[13] Shakarji, C. M., and Srinivasan, V., “Theory and Algorithms for Weighted Total Least-Squares Fitting of Lines, Planes, and Parallel Planes to Support Tolerancing Standards,” ASME Journal of Computing and Information Science in Engineering, 13(3), 2013.

[14] Shakarji, C. M., and Srinivasan, V., “Datum Planes Based on a Constrained L1 Norm,” ASME Journal of Computing and Information Science in Engineering, 15(4), 2015.

[15] Shakarji, C. M., and Srinivasan V., “An improved L1 based algorithm for standardized planar datum establishment,” DETC2014-35461, Proceedings, ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Buffalo, NY, 2014.

[16] Trefethen, Lloyd N., and David Bau III. Numerical linear algebra. Vol. 50. Siam 1997. p. 25-31.