In-Situ Measurement and Distributed Computing for Adjustable CNC Machining

Allan D. Spence[1], D. Alan Sawula[2], James R. Stone[3] and Yu Pin Lin[4]

[1]McMaster University, adspence@mcmaster.ca
[2]McMaster University, sawulad@mcmaster.ca
[3]McMaster University, stonej4@mcmaster.ca
[4]McMaster University, linjy@mcmaster.ca

ABSTRACT

Machining is an established standard for producing the final shape of parts, molds, etc. The nominal CAD representation is used to create sophisticated 3- to 5-axis CNC tool paths. Lack of computational and in-situ measurement resources restricts at machine registration and other adjustments to correct for actual surface geometry. This paper describes progress towards provision of adequate touch trigger probe and laser scanner measurement resources at the machine tool, with supporting computational resources accessed via Ethernet. The emerging MTConnect standard provides executive level monitoring. Illustrative examples include fixture registration using datum spheres, aeroengine assembly weld machining, and associated data fitting and tool path adjustment. The goal is to develop an environment with a higher level of intelligence and decision making ability for machining under variable geometry conditions.

Keywords: geometrically adjustable machining, computer numerical control, laser scanning, reverse engineering, MTConnect.

1. INTRODUCTION

Computer Numerical Control (CNC) machining is the established standard for producing the final shape of cast or forged parts, injection mold cores and cavities, etc. The associated surfaces are complex, and require sophisticated cutter path planning and 3- to 5-axis machine tools to successfully produce. Under ideal conditions, the geometry presented for machining exactly matches the nominal CAD model that was used to plan the path. In practice, actual geometry varies due to inconsistent fixture clamping, casting distortion during cooling, or heat treatment of molds. Variations can also arise from inexact assembly component relative positions, and location and size of weld beads or additive manufacturing laser melt material deposits. In these cases, adjustment of the nominal machining path trajectory is essential to achieve in-tolerance final shape. Modification of feeds and speeds may also be desired. These situations clearly require in-situ measurement of the actual geometry, coupled with adequate computing resources to perform the cutter path adjustments, at a level that currently is not well integrated with CNC machine tool.

This paper describes progress towards integration of measurement resources at the machine tool, with supporting computational resources accessed via Ethernet. The emerging MTConnect standard provides executive level monitoring. Illustrative examples include fixture registration using datum spheres, aeroengine assembly weld machining, and associated data fitting and tool path adjustment. The remainder of the paper is organized as follows. Section 2 describes the proposed system architecture including touch trigger probes, laser scanner, machine tool interface, and geometric computing system. Section 3 describes a first example implemented for fixture registration using datum spheres and planes. Section 4 describes a second example implemented to adjust tool paths for a variable assembly and weld bead aeroengine application. Conclusions follow in Section 5.

2. PROPOSED SYSTEM ARCHITECTURE

The proposed in-situ measurement and distributed computing system architecture is shown in Fig. 1. An industry standard machine tool with motion controller is assumed. Support for a conventional spindle
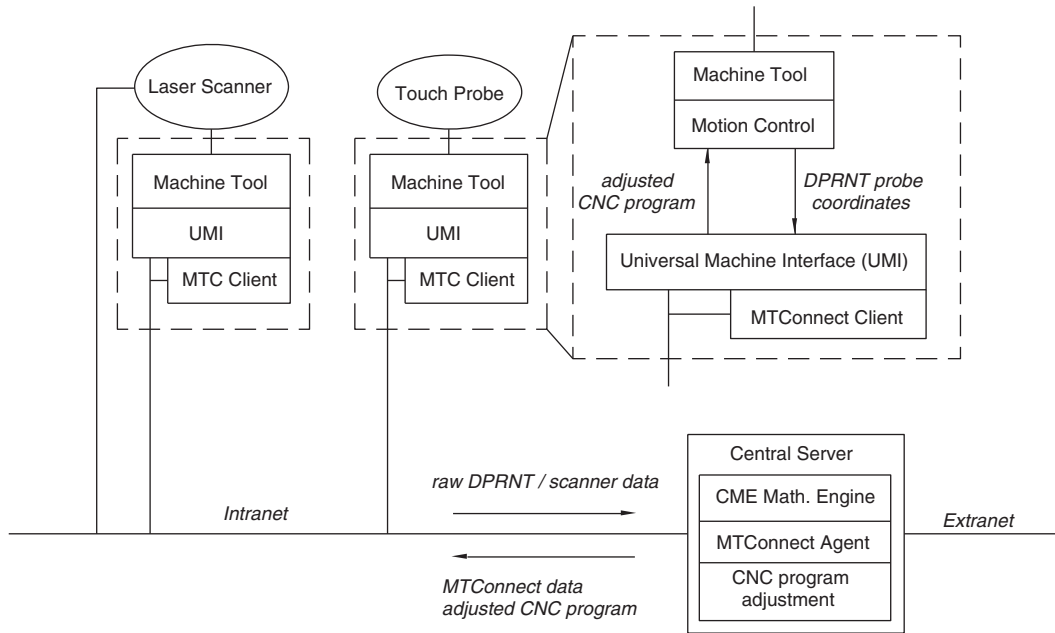
Taylor & Francis
Taylor & Francis Group

Fig. 1:   In-situ measurement and distributed computing system architecture.

mounted touch trigger probe is included. At McMaster a Renishaw [10] MP12 probe was used with a Fanuc [4] 15iM motion control. The G31 skip command was used to move the probe towards the datum surface, and the Macro B DPRNT command was used to transmit the actual touch coordinates via RS232 serial line into the Memex Automation [6] Ax9150 Universal Machine Interface (UMI). The UMI is used as a communications bridge to repeat the touch coordinates over Ethernet for processing using the CheckMate geometry Mathematics Engine (CME) [8]. The CME provides, on a central computer, sophisticated algorithms that cannot be implemented using the limited capability of the local motion control. These algorithms are used for Orthogonal Least Squares (OLS) data fitting, measurement to CAD registration, and transformation of the nominal CNC tool paths to suit actual measured geometry.

The emerging MTConnect standard [7] is available to display summarized results at the central computer, at the machine tool (hosted on either the UMI or a separate Windows based netbook computer), or via extranet to any HTTP capable client browser. In-situ laser scanner support is currently being implemented, and will be reported in a future publication.

3.   FIXTURE REGISTRATION

A classic requirement in machining is fixture setup, establishment of datums, and conversion from the part nominal coordinate system to actual machine tool coordinates. Only under these conditions can in-tolerance production of the part surface geometry

be expected. In cases where repeatability can be established, such as with solidly locating pallets, adjustments can be established with the initial setup and subsequently be reused without further calculation. Initially, or for fixtures that do not repeatably locate, measurement, datum calculation, and coordinate system adjustment must be performed for each setup.

Consider, for example, the setup shown in Fig. 2(a). A rough, partial cylinder shape is created using a band saw, and screw fastener holes are added using a drill press. The part is fastened to the nominally flat plate using two socket head cap screws. Three datum spheres are added to the fixture to permit registration and tool path adjustment between the part and machine tool coordinate system. In the initial / reference (Case 0) (Fig. 3) setup, the fixture is manually aligned so that the part vertical flat sides are aligned with the CNC machine tool axis, and then secured (using cap screws and T-nuts) to the table (Fig 2(b)). The datum spheres are then measured, and their nominal center points are located using the CME Math. Engine. The CAD nominal tool path (Fig. 2(c)) is then executed (Fig. 3(a)).

The cap screws / T-nuts are then loosened, the fixture plate slightly displaced, and the screws tightened again. This is repeated four times (Case 1-4). In each case, after tightening, the datum spheres are remeasured and the OLS centers calculated. A basic check for consistent tightening is made by checking that, after registration, the center location deviations are near zero. Results are calculated and made available, using an MTConnect client display (Fig. 2(d)), to the operator at the machine. The results
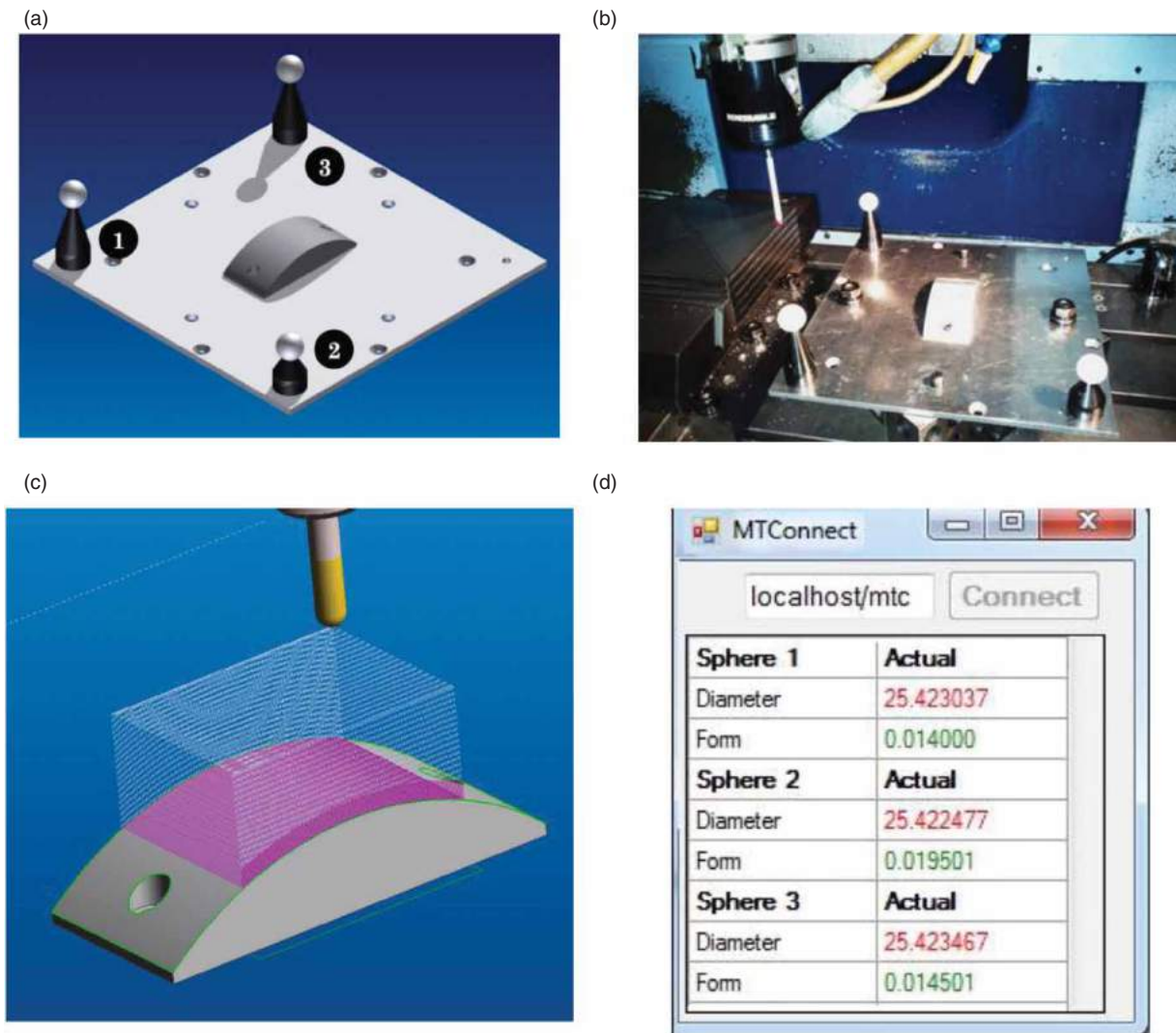
Fig. 2: (a) CAD representation of fixture plate, partial cylinder part, and labeled spheres; (b) mounted on CNC machine tool table with Renishaw MP12 touch trigger probe; (c) CNC tool path; (d) MTConnect Client status display.

can also be displayed using either a local or extranet MTConnect client, and can be recorded to archival storage.

The Case 1-4 sphere center deviations (with respect to Case 0), and homogeneous transformation matrices (HTMs) are summarized in Tab. 1. The maximum sphere center deviation, found in Case 3, is 0.018 mm. The machining setup for Case 0 is shown in Fig. 3(a). To readily support 6 degree of freedom transformation, a 7.9375 mm diameter ball nose mill was used. For Case 1-4, the Case 0 tool path positions were transformed using the listed rotation $\mathbf{R}^k$ and translation $\mathbf{T}^k$ components. The finished part (Fig. 3(b)) was laser scanned, and an OLS cylinder was fitted to the Case 0 points. The surface deviations for Case 1-4 are generally below 0.050 mm

(Fig. 3(c)) – consistent with levels encountered for typical aeroengine blade machining [13].

The primary advantage of using the CME Math. Engine is access to geometric feature fitting and registration algorithms that far exceed the capabilities of the CNC motion control. For example, CNC control based sphere fitting is limited to taking surface points on opposite sides and averaging to obtain the x and y center coordinates. The z coordinate is found by taking a point at the top and subtracting the probe tip and sphere radii (Fig. 4(a)). Using the distributed CME OLS algorithm, the center is determined by minimizing the quantity $S^2 = \sum_{j=1}^{M} d_j^2$ where M is the number of sample points and

$$d_j = \left[\left(s_{j,x} - s_x \right)^2 + \left(s_{j,y} - s_y \right)^2 + \left(s_{j,z} - s_z \right)^2 \right]^{1/2} - r, \quad \mathbf{s}_j$$
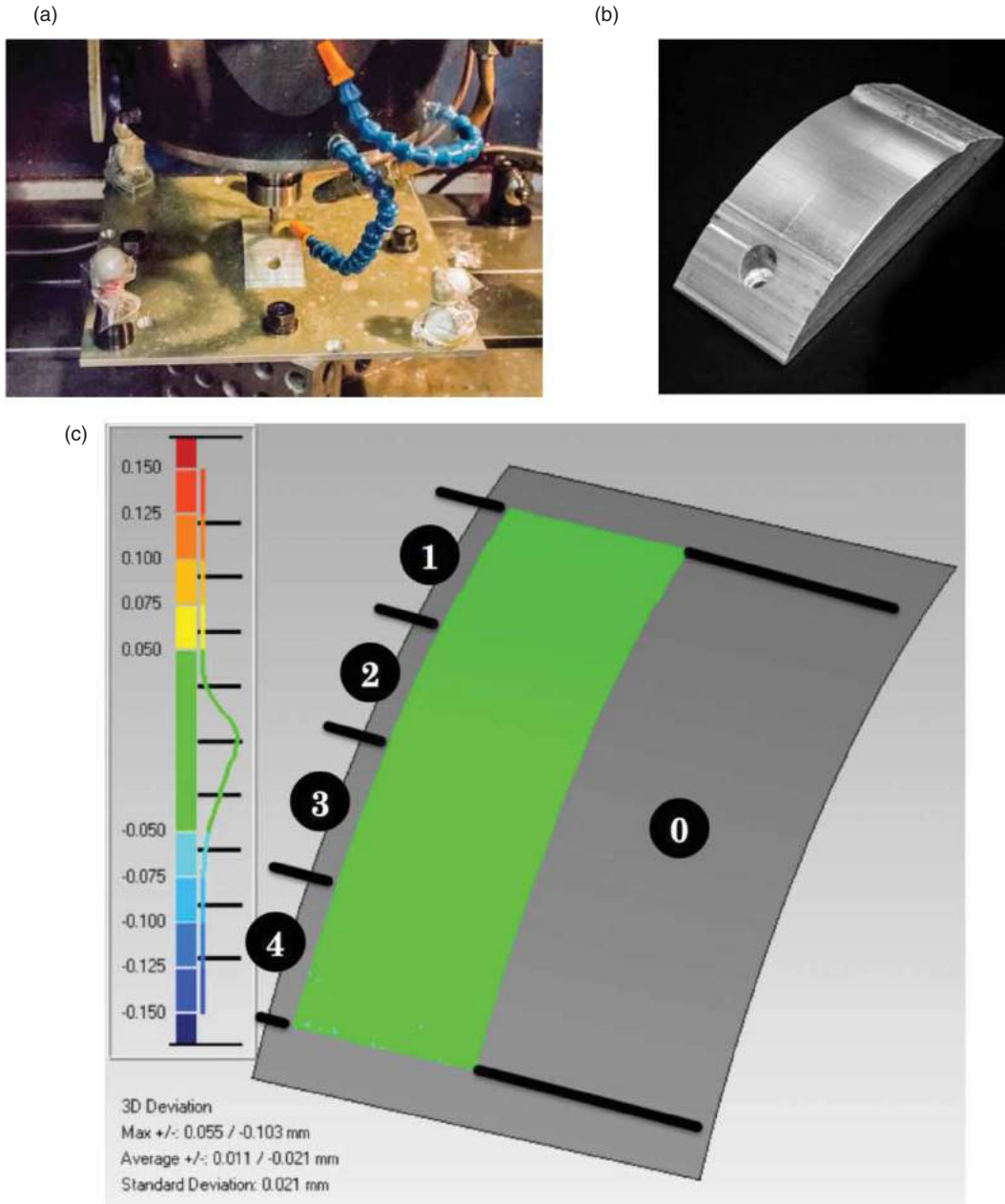
Fig. 3: (a) Case 0 machining; (b) finished part photo; (c) laser scan results.

$= (s_{j,x}, s_{j,y}, s_{j,z})$ is a sample point, $\mathbf{s} = (s_x, s_y, s_z)$ is the fitted center, and r is the fitted radius (Fig. 4(b)). Assuming that the sphere has low form error (is very nearly round), and that the touch probe lobing error is small, the well accepted OLS solution is to begin with the initial linear approximation [1]

$$\theta_j = [(s_{j,x} - s_x)^2 + (s_{j,y} - s_y)^2 + (s_{j,z} - s_z)^2] - r^2$$
$$= \{[(s_{j,x} - s_x)^2 + (s_{j,y} - s_y)^2 + (s_{j,z} - s_z)^2]^{1/2} - r\}$$

$$\cdot \{[(s_{j,x} - s_x)^2 + (s_{j,y} - s_y)^2 + (s_{j,z} - s_z)^2]^{1/2} + r\}$$
$$\approx d_j \cdot 2r$$

and then minimizing the sum $\Theta^2 = \sum_{j=1}^{M} \theta_j^2 = \sum_{j=1}^{M} \{g_j - h - 2[s_{j,x} s_x + s_{j,y} s_y + s_{j,z} s_z]\}^2$ where $g_j = s_{j,x}^2 + s_{j,y}^2 + s_{j,z}^2$ and $h = r^2 - (s_x^2 + s_y^2 + s_z^2)$. Simultaneously solving the partial derivative equations

$$\frac{\partial \Theta^2}{\partial s_x} = 0; \quad \frac{\partial \Theta^2}{\partial s_y} = 0; \quad \frac{\partial \Theta^2}{\partial s_z} = 0; \quad \frac{\partial \Theta^2}{\partial h} = 0$$

	Fitted Sphere Center Deviations			Transformation Components			
Case 1	x	y	z	$\mathbf{R}^1$			$\mathbf{T}^1$
Sphere 1	−0.0022	0.0148	0.0002	0.9999	−0.0173	−0.0002	−0.8700
Sphere 2	0.007	−0.0049	−0.0006	0.0173	0.9999	0.0000	−0.5866
Sphere 3	−0.0049	−0.0098	0.0005	0.0002	0.0000	1.0000	0.0060
Case 2	x	y	z	$\mathbf{R}^2$			$\mathbf{T}^2$
Sphere 1	0.0011	0.0053	−0.0001	0.9999	0.0157	−0.0001	16.6543
Sphere 2	0.0085	−0.0098	−0.0008	−0.0157	0.9999	0.0000	−0.6807
Sphere 3	−0.0096	0.0045	0.0009	0.0001	0.0000	1.0000	0.0117
Case 3	x	y	z	$\mathbf{R}^3$			$\mathbf{T}^3$
Sphere 1	−0.0114	0.0029	0.0011	1.0000	−0.0062	−0.0002	−48.4372
Sphere 2	0.0182	−0.0064	−0.0017	0.0062	1.0000	0.0000	−1.9574
Sphere 3	−0.0067	0.0036	0.0006	0.0002	0.0000	1.0000	0.0219
Case 4	x	y	z	$\mathbf{R}^4$			$\mathbf{T}^4$
Sphere 1	−0.0113	0.0046	0.0011	0.9993	0.0377	0.0000	−35.1846
Sphere 2	0.0087	0.0026	−0.0008	−0.0377	0.9993	0.0002	−0.2768
Sphere 3	0.0026	−0.0072	−0.0003	−0.0000	−0.0002	1.0000	−0.0090

Tab. 1: Sphere center deviations (relative to Case 0) and Transformation Components – rotation $\mathbf{R}^k$, translation $\mathbf{T}^k$.
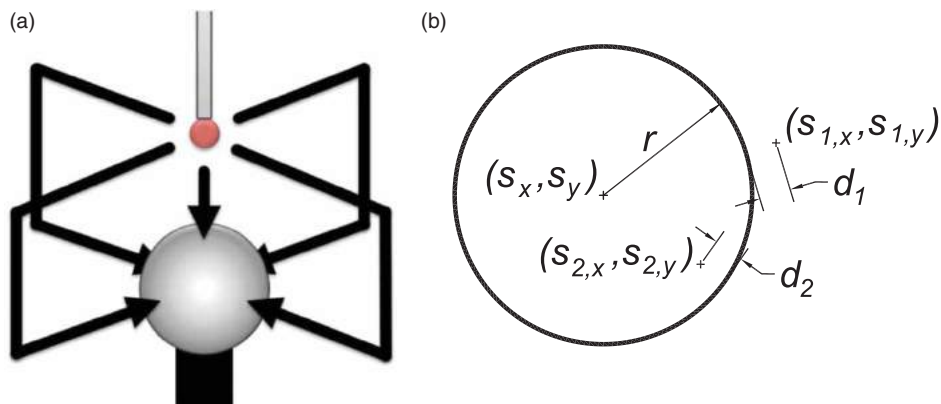

Fig. 4: (a) 5 point sphere probing; (b) OLS sphere data fitting.

leads to the linear system

$$\begin{bmatrix} \sum_{j=1}^{M} 1 & \sum_{j=1}^{M} s_{j,x} & \sum_{j=1}^{M} s_{j,y} & \sum_{j=1}^{M} s_{j,z} \\ \sum_{j=1}^{M} s_{j,x} & \sum_{j=1}^{M} s_{j,x}^2 & \sum_{j=1}^{M} s_{j,x}s_{j,y} & \sum_{j=1}^{M} s_{j,x}s_{j,z} \\ \sum_{j=1}^{M} s_{j,y} & \sum_{j=1}^{M} s_{j,x}s_{j,y} & \sum_{j=1}^{M} s_{j,y}^2 & \sum_{j=1}^{M} s_{j,y}s_{j,z} \\ \sum_{j=1}^{M} s_{j,z} & \sum_{j=1}^{M} s_{j,x}s_{j,z} & \sum_{j=1}^{M} s_{j,y}s_{j,z} & \sum_{j=1}^{M} s_{j,z}^2 \end{bmatrix}$$

$$\cdot \begin{bmatrix} h \\ 2s_x \\ 2s_y \\ 2s_z \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{M} g_j \\ \sum_{j=1}^{M} s_{j,x}g_j \\ \sum_{j=1}^{M} s_{j,y}g_j \\ \sum_{j=1}^{M} s_{j,z}g_j \end{bmatrix}$$

that can be solved for initial values for h, s_x, s_y, and s_z. Iterative improvement is then used to minimize S^2

in the original exact problem by solving for

$$\begin{bmatrix} \frac{\partial S^2}{\partial s_x} \\ \frac{\partial S^2}{\partial s_y} \\ \frac{\partial S^2}{\partial s_z} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 S^2}{\partial s_x^2} & \frac{\partial^2 S^2}{\partial s_x \partial s_y} & \frac{\partial^2 S^2}{\partial s_x \partial s_z} \\ \frac{\partial^2 S^2}{\partial s_x \partial s_y} & \frac{\partial^2 S^2}{\partial s_y^2} & \frac{\partial^2 S^2}{\partial s_y \partial s_z} \\ \frac{\partial^2 S^2}{\partial s_x \partial s_z} & \frac{\partial^2 S^2}{\partial s_y \partial s_z} & \frac{\partial^2 S^2}{\partial s_z^2} \end{bmatrix} \cdot \begin{bmatrix} \delta s_x \\ \delta s_y \\ \delta s_z \end{bmatrix}$$

and then updating the sphere center estimate using $s_x = s_x - \delta s_x$, $s_y = s_y - \delta s_y$, and $s_z = s_z - \delta s_z$. The improved radius estimate is $r = \frac{1}{M} \sum_{j=1}^{M} [(s_{j,x} - s_x)^2 + (s_{j,y} - s_y)^2 + (s_{j,z} - s_z)^2]^{1/2}$. Iterations continue until the estimates for $\mathbf{s} = (s_x, s_y, s_z)$ and r converge. There is no direct need to include the touch probe radius, or take points on diametrically opposite sides of the sphere.
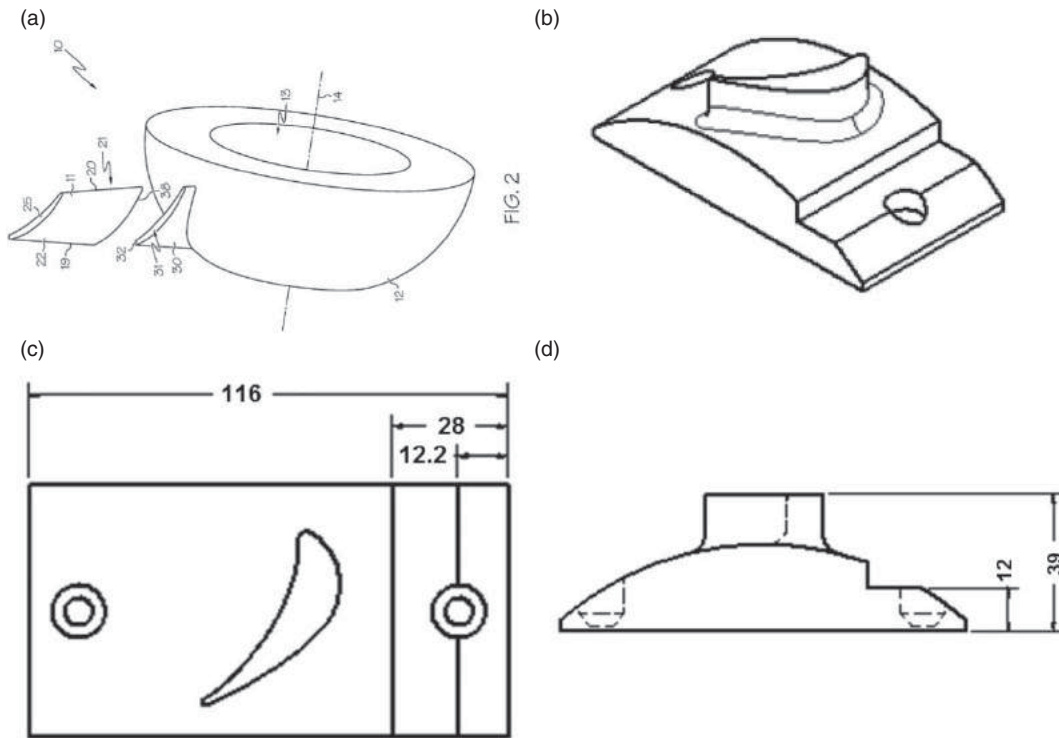
Fig. 5: (a) Bladed Disk assembly by welding [3]; (b,c,d) single blade demonstration part. Dimensions in mm.

Registration of sphere centers between Cases is performed using the algorithm reported in [2], as summarized next. For each Case k, denote sphere center i as $\mathbf{s}_i^k = \left(s_{i,x}^k, s_{i,y}^k, s_{i,z}^k\right)$. Calculate the mean values $\bar{\mathbf{s}}^k = \frac{1}{3}\sum_{i=1}^3 \mathbf{s}_i^k$ and denote the mean translated centers as $\bar{\mathbf{s}}_i^k = \mathbf{s}^k - \mathbf{s}_i^k$. To transform from Case 0 to Case k, form the matrix $\mathbf{A} = \sum_{i=1}^3 \mathbf{s}_i^0 \cdot \left(\mathbf{s}_i^k\right)^T$ and calculate the Singular Value Decomposition (SVD) $\mathbf{U}\sum\mathbf{V}^T = \mathbf{A}$. Calculate $\mathbf{W} = \mathbf{V}\mathbf{U}^T$. If $\det(\mathbf{W}) > 0$, then the rotation matrix is $\mathbf{R}^k = \mathbf{W}$. If $\det(\mathbf{W}) < 0$, then form $\mathbf{V}' = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & -\mathbf{v}_3 \end{bmatrix}$ where $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are the column vectors of $\mathbf{V}$, and calculate $\mathbf{R}^k = \mathbf{V}'\mathbf{U}^T$. The translation vector is $\mathbf{T}^k = \bar{\mathbf{s}}^k - \bar{\mathbf{s}}^0$.

These sophisticated calculations are well beyond the mathematical capability of the CNC motion control, but are well established algorithms that can use any reasonable number of sample points per sphere, and any reasonable number of sphere centers for least squares fit registration. The connectivity between the UMI, MTConnect, etc. provides access to centralized computing in an automated manner.

4. VARIABLE ASSEMBLY AND WELD BEAD AEROENGINE EXAMPLE

More complex circumstances arise when there is uncertainty in casting stock allowance, due to pattern shape deviations from nominal, cooling distortion,

etc. If a correct machining setup, that distributes stock removal over the whole casting, is not performed, wall thicknesses and hole positions will be out of tolerance, resulting in a scrapped part [9]. Similar situations can arise with forgings, or heat treat distortion before finish machining of injection molds. In the case of very large components that are difficult to machine from a single blank, such as aeroengine fan assemblies, assembly and welding of the blades to a central hub is an attractive alternative. A representative photo, showing a hub and blades roots, is shown in Fig. 5(a). For demonstration herein, a single blade demonstration part was developed (Fig. 5(b,c,d)). The step area was added to permit local registration without the need for tooling spheres. In practice, the whole hub would be available to establish datum registration. Slight deviations from nominal relative hub / blade position and orientation were introduced for testing.

The nominal CAD/CAM tool paths are designed using a helical style motion that advances along the fillet center, with extensions beyond the fillet to ensure that all excess weld material is machined away (Fig. 6). This custom approach provides two key advantages:

- the uncut chip thickness depends on the designed pitch, and is independent of the uncertain actual weld bead shape.
- the cutter contact / cutter center vectors remain normal to the nominal surface, simplifying
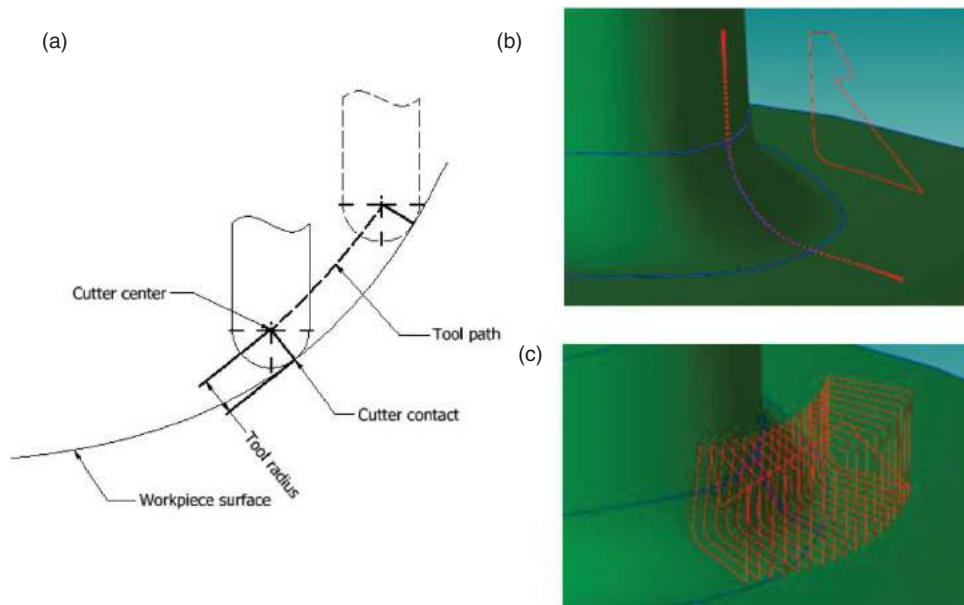
Fig. 6: Helical custom tool path design: (a) cutter point definitions; (b) cutter contact points and single helical style loop extended beyond fillet onto contiguous surfaces; (c) sequence of helical loops used for machining experiments.
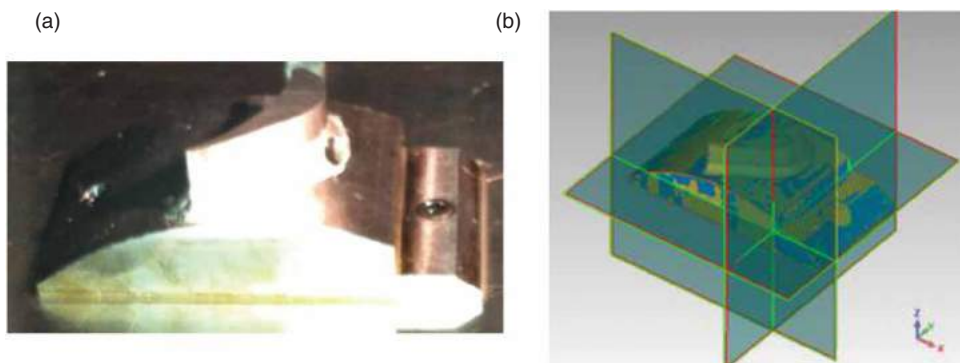

Fig. 7: (a) demonstration part; (b) laser scanner point cloud alignment to CAD.
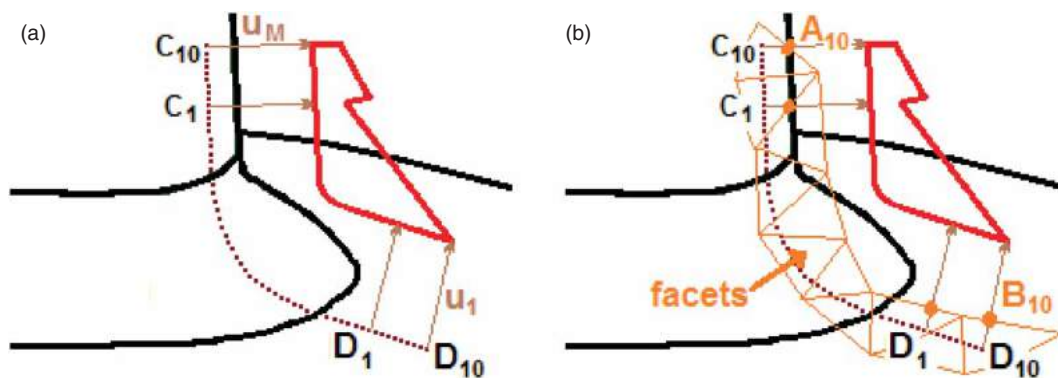

Fig. 8: Tool path customization: (a) cutter contact to cutter center vectors; (b) actual facet intersection points.

correction for actual measured geometry to off-setting along the normal vector.

The demonstration parts (Fig. 7(a)) were laser scanned using a Roland LPX-600DS. Approximately 120,000 points per part were collected. Registration of the point cloud to the CAD coordinate system was accomplished using a modified 3-2-1 method in which Orthogonal Least Squares planes were best fit to the points on the ZX side (fixing the Y origin and rotation about the X and Z axes), the XY side (fixing the Z origin and rotation about the Y axis), and the YZ side (fixing the X origin) (Fig. 7(b)). At this point, the nominal tool path can be customized to the actual measured geometry.

After the measured point cloud has been registered to the CAD coordinate system, the nominal tool path is adjusted to suit the actual geometry. This is accomplished by first creating triangular planar facets from the point cloud. Next, for each (both blade and hub) extended in-contact portion of a helical style loop, $N = 10$ cutter contact to cutter centre vectors are constructed (Fig. 8(a)). The intersection of the vector with the actual measurement facet is then computed (Fig. 8(b)). The blade intersection points are labeled $\mathbf{A}_1, \ldots, \mathbf{A}_{10}$ and the blade cutter contact points are labeled $\mathbf{C}_1, \ldots, \mathbf{C}_{10}$. The hub intersection points are labeled $\mathbf{B}_1, \ldots, \mathbf{B}_{10}$ and the hub cutter contact points are labeled $\mathbf{D}_1, \ldots, \mathbf{D}_{10}$.
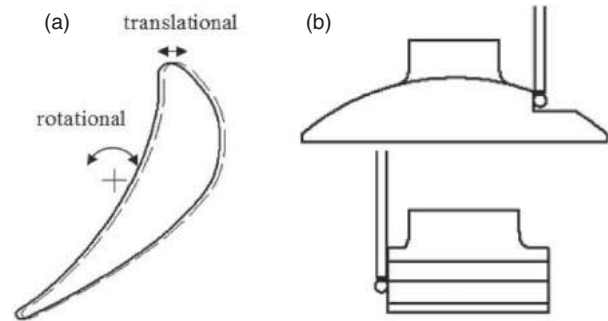

Fig. 9: (a) blade translation and rotation; (b) touch trigger probe datum plane registration.

The individual offsets between the actual facet intersection and the nominal CAD cutter contact points are defined as $d_{k,\mathbf{AC}} = \mathbf{A}_k - \mathbf{C}_k$ and $d_{k,\mathbf{BD}} = \mathbf{B}_k - \mathbf{D}_k$ and the average offsets are $\overline{d_{k,\mathbf{AC}}} = 0.1 \cdot \sum_{k=1}^{10} d_{k,\mathbf{AC}}$ and $\overline{d_{k,\mathbf{BD}}} = 0.1 \cdot \sum_{k=1}^{10} d_{k,\mathbf{BD}}$. Note that the averages may be either positive or negative. The sample size $N = 10$ was chosen based on the observed weld bead extent, and a reasonable spacing between tool path positions. Along the entire helix style loop path (including the fillet) with nominal cutter contact points $\mathbf{F}_1 = \mathbf{D}_{10}, \ldots, \mathbf{F}_{10} = \mathbf{D}_1, \ldots, \mathbf{F}_{M-10} = \mathbf{C}_1, \ldots, \mathbf{F}_M = \mathbf{C}_{10}$, define the adjusted contact points
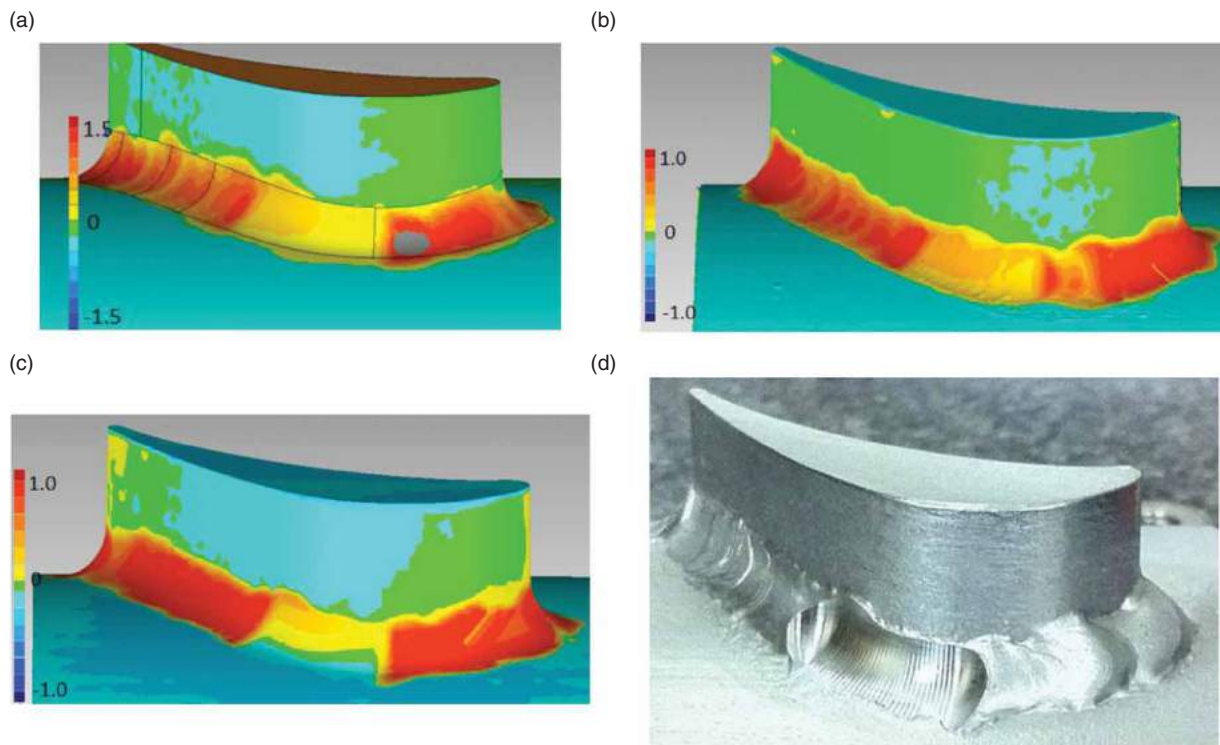

Fig. 10: Demonstration part geometric comparisons: (a) nominal - case i); (b) translated by −0.5 mm, rotated by 1 degree clockwise – case ii); (c) translated by +0.5 mm, rotated by 1 degree counterclockwise – case iii); photograph for case iii). Dimensions in mm.

as $\mathbf{E}_1 = \mathbf{B}_{10}, \ldots, \mathbf{E}_{10} = \mathbf{B}_1, \mathbf{E}_{11}, \ldots, \mathbf{E}_{M-11}, \mathbf{E}_{M-10} = \mathbf{A}_1,$ $\ldots, \mathbf{E}_1 = \mathbf{A}_{10}$ where the interior extended fillet points are $\mathbf{E}_{m=11,\ldots,M-11} = \mathbf{F}_m + (\overline{d_{\mathbf{BD}}} + (m-10) \cdot (\overline{d_{\mathbf{AC}}} - \overline{d_{\mathbf{BD}}})$ $/(M-22)) \cdot (\mathbf{u}_m/\|\mathbf{u}_m\|)$ and $\mathbf{u}_m$ is the vector directed from the nominal cutter contact point $\mathbf{F}_m$ towards its corresponding cutter center point. The sequence of adjusted points $\mathbf{E}_m$ is then offset in the direction $\mathbf{u}_m$ by the ball nose cutting tool radius (in this case 5 mm) to obtain the cutter center points. The cutter center points are connected together to create the adjusted helical style loop tool path, corrected for the actual measured geometry. As noted earlier, this approach innovatively captures the knowledge contained in the nominal CAD geometry, including the cutter contact to cutter center surface normal vectors.

Geomagic [5], and later the Origin CME, were used for the 3-2-1 registration, to create the facets from the point cloud, and for geometric comparison. Tech Soft 3D HOOPS [12] and Spatial Corp. ACIS [11] were used for the nominal CAD tool path creation and adjustment. The procedure was tested for actual assemblies with: i) nominal blade geometry; ii) blade translation of -0.5 mm and rotation of 1 degree clockwise (Fig. 9(a)); iii) blade translation of $+0.5$ mm and rotation of 1 degree counterclockwise. During setup on the CNC machine tool, the demonstration part location was registered using the touch trigger probe and OLS plane / line fitting (Appendix A) (Fig. 9(b)).

Geometric comparisons are shown in Fig. 10. Match to approximately 0.25 mm along the fillet area is easily achieved. For demonstration, an increased step over distance between subsequent helical style loops was used, and hence machining cusps can be seen. In actual practice a smaller step would reduce the cusp height to any specified value. Since only a portion of overall tool path was executed, the thickness of the weld bead material can also be seen. As stated earlier, the helical style loops allow step over distance adjustment to control the milling feed per tooth and cusp height. Compared to the tool radius, minor changes in weld bead thickness are insignificant. For hard aerospace material alloys, this machining strategy is particularly attractive.

5. CONCLUSIONS

This paper has described progress towards providing geometric data fitting and registration computational resources, via Ethernet and MTConnect standards, to support touch trigger probe and laser scanner measurement for CNC machining. Illustrative examples included fixture registration using datum spheres, aeroengine assembly weld machining, and associated data fitting and tool path adjustment. Ongoing research laboratory development includes integration with a CNC mounted laser scanner and additional development of the CME Math. Engine for registration and facet creation.

REFERENCES

[1]   Anthony, G.T.; Cox, M.G.: Reliable Algorithms for Roundness Assessment to BS3730, in Software for Coordinate Measuring Machines, Cox, M.G. and Peggs, G.N., Ed. Teddington, UK: National Physical Laboratory, 1986, 30–37.

[2]   Arun, K.S.; Huang, T.S.; Blostein, S.D.: Least-Squares Fitting of Two 3-D Point Sets, IEEE Trans. PAMI, 9(5), 1987, 698–700.

[3]   Chung, V.; Swanson, J.R.; Howard, W.; Bazzell, B; Berry, B.; Natter, C.; Ryan, D., Linear Friction Welded BLISK and Method of Fabrication, U.S. Patent Application 20090185908, 2009.

[4]   FANUC FA America, Hoffman Estates, IL, USA, www.fanucfa.com.

[5]   Geomagic, Inc., Morrisville, NC, USA, www.geomagic.com.

[6]   Memex Automation, Burlington, ON, Canada, www.memex.ca.

[7]   MTConnect Institute, Association for Manufacturing Technology, McLean, VA, USA, www.mtconnect.org.

[8]   Origin International, Markham, ON, Canada, www.originintl.com.

[9]   Origin Int'l, Balancing Stock on Castings: A CheckMate Adaptive Manufacturing Application, http://info.originintl.com/Portals/37807/docs/solidiformcastingr2withcopyright.pdf.

[10]   Renishaw plc., Wotton-under-Edge, UK, www.renishaw.com.

[11]   Spatial Corp., Broomfield, CO, USA, www.spatial.com.

[12]   TechSoft3D, Berkeley, CA, USA, www.techsoft3d.com.

[13]   Wu, C.Y.: Arbitrary Surface Flank Milling and Flank SAM in the Design and Manufactoruing of Jet Engine and Compressor Coils, in ASME Turbo Expo 2012, Copenhagen, 2012.

APPENDIX A – OLS PLANE / LINE FITTING

Orthogonal Least Squares (OLS) plane and line fitting is implemented using the following well known approach. Given a set of M sample points $\mathbf{s}_m = (x_m, y_m, z_m)$, choose a point $\mathbf{p}$ and direction $\mathbf{v}$ of the best fit line such that the sum $S^2 = \sum_{m=1}^{M} \|d_m\|^2$ is minimized, where $\|d_m\|$ is the orthogonal distance from point $\mathbf{s}_m$ to the line (Fig. A1(a)). This problem can be generalized to the OLS best fit plane case as well, in which case the sum to be minimized is
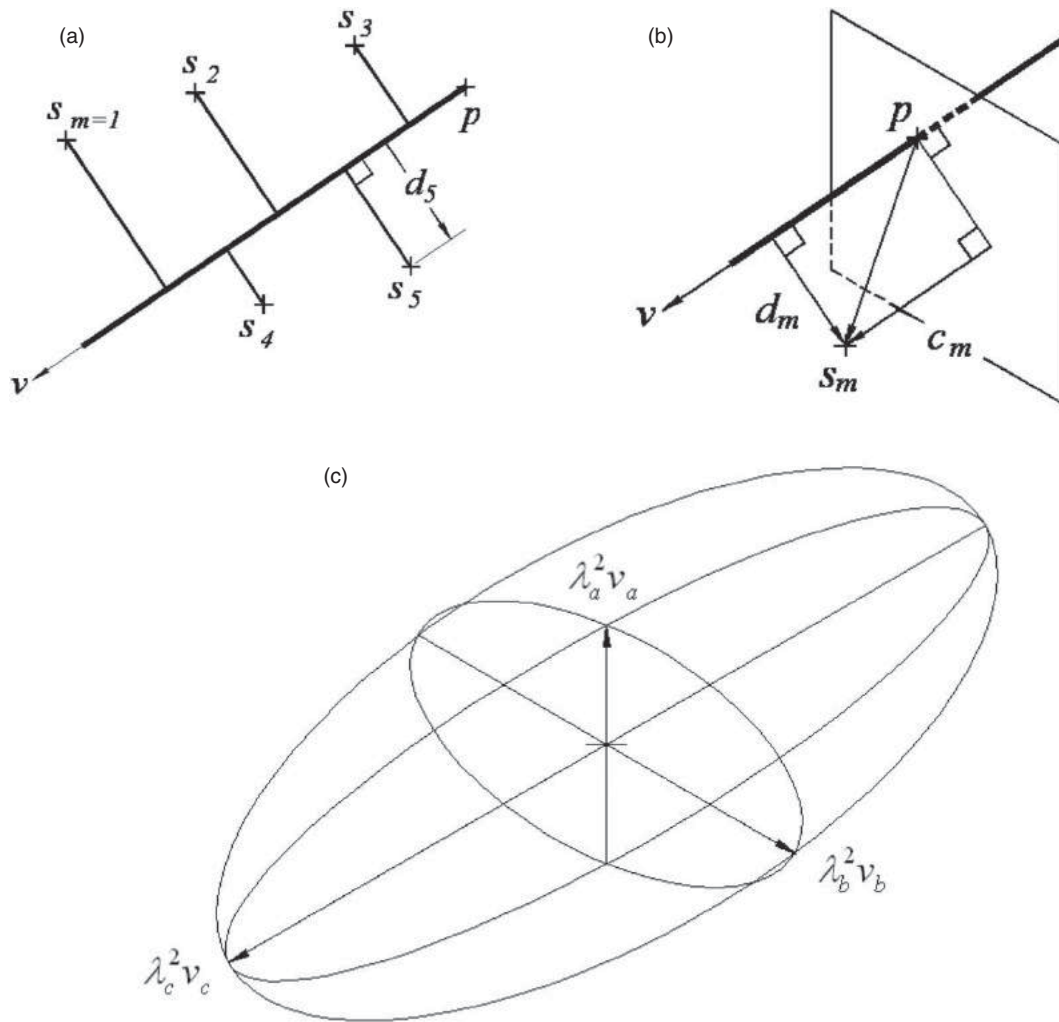
Fig. A1: (a) OLS line fit definitions; (b) OLS plane fit definitions; (c) quadratic form ellipse illustration.

$S^2 = \sum_{m=1}^{M} \|c_m\|^2$ where $\|c_m\|$ is the orthogonal distance from point $\mathbf{s}_m$ to the plane (Fig. A1(b)). Without loss of generality, it can be assumed that the direction vector has unit length. That is $\|\mathbf{v}\| = 1$. Solving this problem consists of separate steps to first determine $\mathbf{p}$, and then $\mathbf{v}$.

For planes, $\|c_m\| = (\mathbf{s}_m - \mathbf{p}) \cdot \mathbf{v}$ and hence the sum to be minimized may be written as

$$S^2 = \sum_{m=1}^{M} [(\mathbf{s}_m - \mathbf{p}) \cdot \mathbf{v}]^2$$

$$= \sum_{m=1}^{M} [(s_{m,x} - p_x)v_x + (s_{m,y} - p_y)v_y + (s_{m,z} - p_z)v_z]^2$$

Taking partial derivatives with respect to the components of $\mathbf{p}$, and setting them equal to zero yields

$$0 = \frac{\partial S}{\partial p_x} = -2 \sum_{m=1}^{M} [(s_{m,x} - p_x)v_x^2 + (s_{m,y} - p_y)v_x v_y + (s_{m,z} - p_z)v_x v_z]$$

$$0 = \frac{\partial S}{\partial p_y} = -2 \sum_{m=1}^{M} [(s_{m,x} - p_x)v_x v_y + (s_{m,y} - p_y)v_y^2 + (s_{m,z} - p_z)v_y v_z]$$

$$0 = \frac{\partial S}{\partial p_z} = -2 \sum_{m=1}^{M} [(s_{m,x} - p_x)v_x v_z + (s_{m,y} - p_y)v_y v_z + (s_{m,z} - p_z)v_z^2]$$

Without loss of generality, the coordinate system can freely be chosen so that $\mathbf{v}$ is aligned with one of the x, y, or z axes. For example, if $\mathbf{v}$ is aligned with the x axis, then $v_x = 1$ and it must hold that $0 = \sum_{m=1}^{M} (s_{m,x} - p_x)$ or $p_x = \frac{1}{M}\sum_{m=1}^{M} s_{m,x}$. Similarly if $\mathbf{v}$ is aligned with the y axis, then $v_y = 1$ and it must hold that $0 = \sum_{m=1}^{M} (s_{m,y} - p_y)$ or $p_y = \frac{1}{M}\sum_{m=1}^{M} s_{m,y}$. If $\mathbf{v}$ is aligned with the z axis, then $v_z = 1$ and it must hold that $0 = \sum_{m=1}^{M} (s_{m,z} - p_z)$ or $p_z = \frac{1}{M}\sum_{m=1}^{M} s_{m,z}$. This result is also valid for lines.

Again, in the plane case, for convenience, define $\mathbf{a}_m = \mathbf{s}_m - \mathbf{p}$. The sum to be minimized may be written as

$$S^2 = \sum_{m=1}^{M} [\mathbf{a}_m \cdot \mathbf{v}]^2 = \sum_{m=1}^{M} [a_{m,x}v_x + a_{m,y}v_y + a_{m,z}v_z]^2$$
$$= \mathbf{v}^T \mathbf{A}^T \mathbf{A} \mathbf{v}$$

where

$$\mathbf{A} = \begin{bmatrix} a_{1,x} & a_{1,y} & a_{1,z} \\ \vdots & \vdots & \vdots \\ a_{M,x} & a_{M,y} & a_{M,z} \end{bmatrix}$$

This is an ellipsoidal quadratic form in $\mathbf{v}$, and hence the sum has extreme values when $\mathbf{v}$ is aligned with one of the principle directions (eigenvectors) of $\mathbf{A}^T \mathbf{A}$. Note that $\mathbf{A}$ and $\mathbf{A}^T \mathbf{A}$ have the same (unit length) eigenvectors, denoted $\mathbf{v}_a$, $\mathbf{v}_b$, and $\mathbf{v}_c$. If $\mathbf{A}^T \mathbf{A}$ has corresponding eigenvalues λ_a^2, λ_b^2, and λ_c^2 then $\mathbf{A}$ will have eigenvalues $\lambda_a < \lambda_b < \lambda_c$. These are the semi-lengths of the axes of the ellipsoidal quadratic form $\mathbf{v}^T \mathbf{A}^T \mathbf{A} \mathbf{v}$ (Fig. A1(c)), and hence the extreme values of the sum of squares S^2. The minimum value is $S_a^2 = \lambda_a^2$ and $\mathbf{v} = \mathbf{v}_a$ is the corresponding normal vector for the OLS plane.

For lines, $\|d_m\|^2 = \|(\mathbf{s}_m - \mathbf{p}) \times \mathbf{v}\|^2 = \|\mathbf{a}_m \times \mathbf{v}\|^2$, and the sum is $S^2 = \sum_{m=1}^{M} \|d_m\|^2$. Using the same definition of $\mathbf{A}$ that was used for planes, the solution occurs when $v = v_c$.

Eigenvalue/eigenvector problems such as this are most frequently solved using the Singular Value Decomposition (SVD) matrix factorization. Using the SVD representation, $\mathbf{A} = \mathbf{U}\mathbf{W}\mathbf{V}^T$. The diagonal of $\mathbf{W}$ contains the eigenvalues, and the columns of $\mathbf{V}$ are the corresponding eigenvectors. The auxiliary matrix $\mathbf{U}$ satisfies $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Hence $\mathbf{A}^T \mathbf{A} = (\mathbf{U}\mathbf{W}\mathbf{V}^T)\mathbf{U}\mathbf{W}\mathbf{V}^T = \mathbf{V}\mathbf{W}\mathbf{U}^T\mathbf{U}\mathbf{W}\mathbf{V}^T = \mathbf{V}\mathbf{W}^2\mathbf{V}^T$.