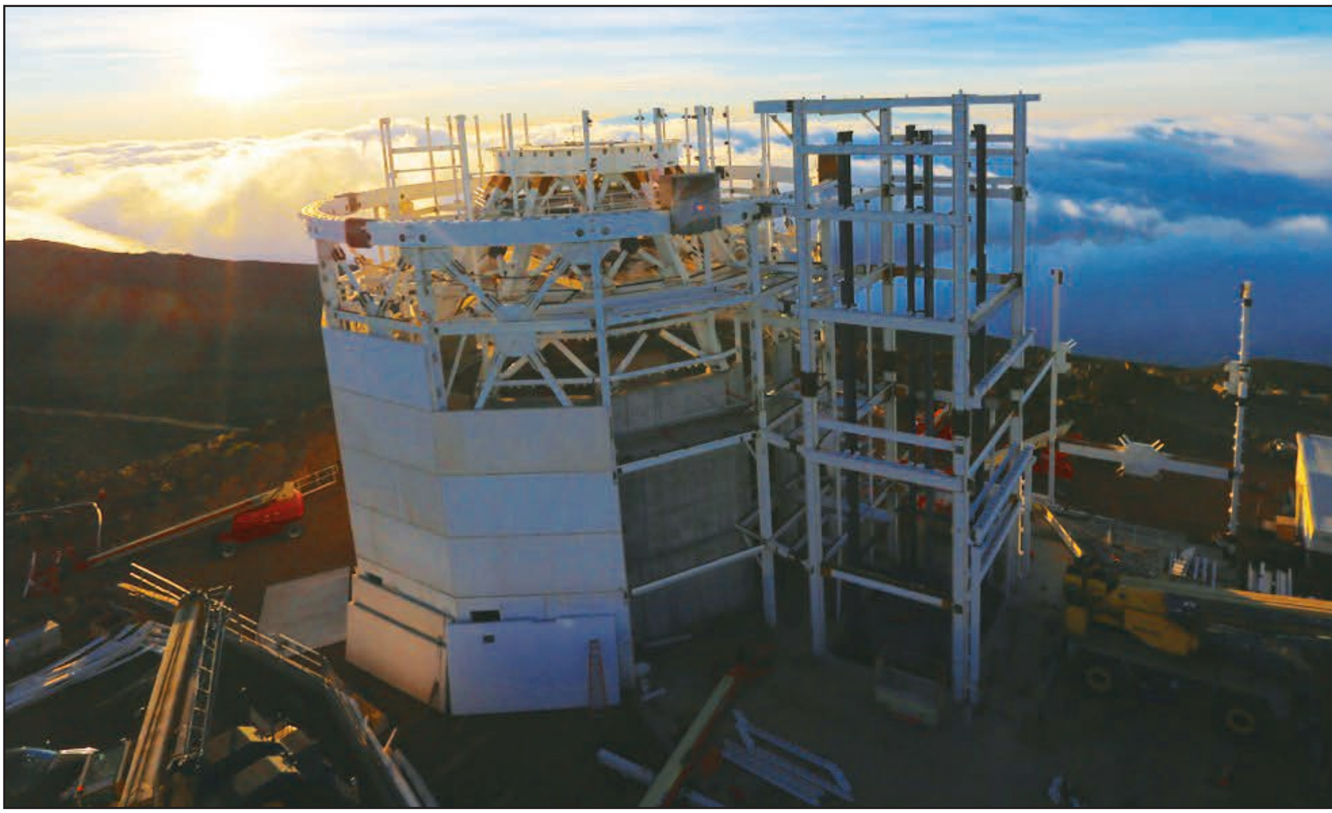
Vol. 10, No. 1 ◆ Spring 2015

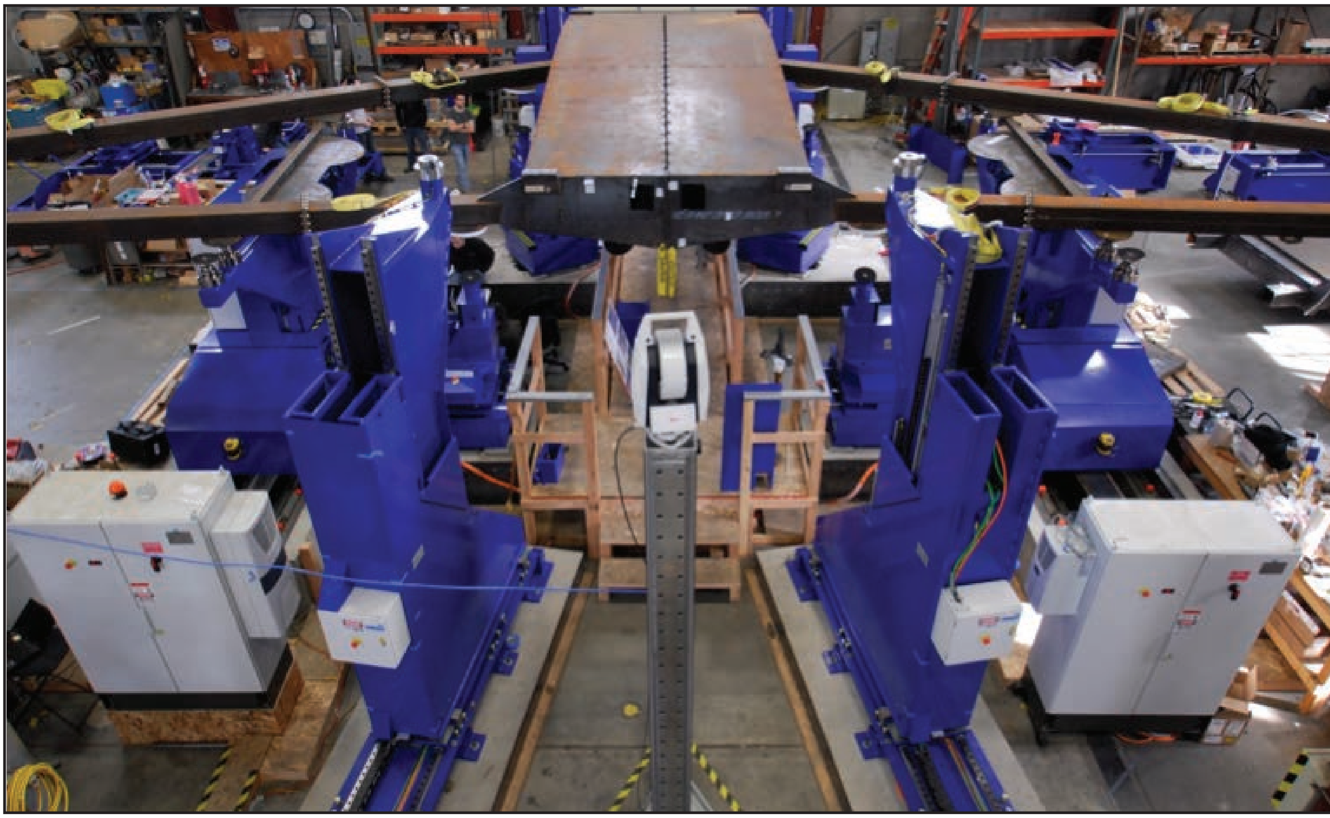
The Journal of the CMSC

THE PUBLICATION FOR 3D MEASUREMENT TECHNOLOGY



IN THIS ISSUE:

WWW.CMSC.ORG



Automated Metrology in a Business Jet Final Assembly Line

by Robert Flynn and Schuyler Horky, Electroimpact Inc.

An aircraft final assembly line (FAL) offers many opportunities for improved assembly via metrology. This article describes an implementation of an FAL with automated positioners and a metrology system. The aircraft in question is a business jet with an approximate 105-ft length and 105-ft wingspan. Automated metrology solutions reduce assembly time, improve quality, increase repeatability, and deskill the operation so that those who are not engineers can carry out a more rapid and accurate assembly process. A novel human-machine interface (HMI) gives a common look and feel throughout all operations in the multiple work cells, provides user instructions at the task-by-task level, and places a list with task checkoff functionality on the screen. The HMI uses SpatialAnalyzer (SA) from New River Kinematics to control laser tracker operations, record data, and communicate with a programmable logic controller (PLC) to command machine actions, yet all functionality is programmable via a Microsoft Excel spreadsheet for easy modification of user instructions, graphics, and automation.

NOMENCLATURE

- **SpatialAnalyzer (SA):** A metrology software product developed by New River Kinematics. SpatialAnalyzer supports a diverse range of metrology instruments and is capable of many geometric operations and computations.
- **Least-squares best fit or best fit:** An algorithm that minimizes the sum of all the deltas between a desired position and current position.
- **Transformation matrix:** A rectangular array of numbers defining the orientation and position of an object in 3D space. Transformation matrixes are usually calculated during a best fit.
- **Rigid body move:** The result of an object moving from point A to point B without deflection; usually the desired motion for objects being manipulated by positioners. Such motion is achieved by correct simultaneous movement of multiple positioners, which is sometimes called "cam" motion.
- **Programmable logic controller (PLC):** A low-level, embedded system that is used to control machine action in industrial applications.
- **Sphere-mounted retroreflector (SMR):** An array of three mutually perpendicular mirrors set in a tooling ball; the target that a laser tracker measures.
- **Foundation reference system (FRS):** A collection of very carefully measured permanent targets distributed through the work area, mounted in the floor or on very rigid walls.

- **Vector bar (aka hidden point rod):** A bar with three aligned points at precise distances from each other. Two points on the vector bar are measured by an instrument, and the location of the third point can be calculated through SpatialAnalyzer.
- **Human-machine interface (HMI):** A computer program that presents a graphical user interface used to monitor and control machine action.
- **Work center:** A working cell in which two sections of an aircraft are joined. This area includes laser trackers, parts of the factory FRS, positioners, a work center PC, and PLC connected together through a common local area network.

SYSTEM OVERVIEW

Components

The FAL is composed of five components joined together in three work centers, as seen in figure 1. First, two wing halves are joined. Next, the completed wing is joined to the center fuselage. Finally, the forward and aft fuselage sections are joined to the center fuselage in one work center during two separate operations. At all work centers, components are held by precision positioners, most of which have three axes. Components all carry targets for laser trackers and are located by trackers at each station.

Adaptive tooling

There are several distinct approaches to tooling design. Traditional rigid tooling consists of fixed elements and/or elements that bolt together with a high degree of repeatability. Flexible tooling enables the use of a single fixture for assembly of two or more part variants (for example, a spar-assembly fixture that can be used for the front, center, or rear spar assembly). Adaptive tooling recognizes the unique as-built dimensions of each part and makes slight adjustments accordingly to enable a more accurate assembly.

Architecture

In this case study, the FAL process relies on automated metrology for part data and automation to accurately move the tooling via machine-integrated control loops of laser trackers and PLC-controlled subassembly positioners together under the control of each work center PC. This tooling is both flexible (it accommodates two aircraft variants) and adaptive. All join procedures are managed by the work center PCs, which run an HMI with an operation-specific instruction set called a task script. For each join, the HMI prompts the operator to load part-specific files. Each file contains unique data about the corresponding subassembly used during the measurement of each part. This allows the FAL solution to adapt to subtle part-to-part variances to achieve more accurate and rapid joins.

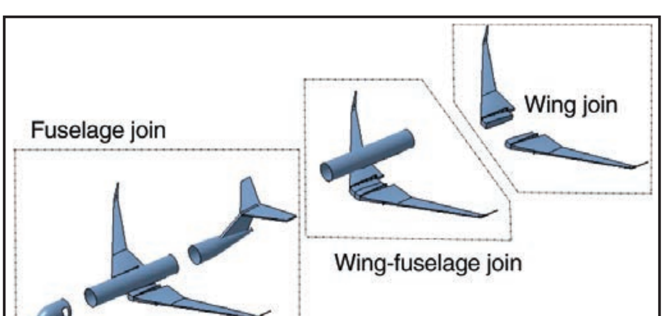


Figure 1. The FAL work centers

WING-WING JOIN PROCESSES IN DETAIL

Setup

The wing-wing work center brings two wing halves together for a join. To start, discrete wing halves are loaded by crane onto a set of three precision multi-axis positioners, as seen in figure 2. The operator starts the HMI and follows the instructions on the screen. Serialized data packages are loaded into the HMI that allow part-specific computations to be performed, and thus enable the positioners to adapt to the wings. The HMI also imports an operation-specific file containing the approximate positions of all laser trackers to be used, nominal positions for both wings, and approximate initial positions of the wings for the work center, after which one tracker on a "pop up" tower (forward of the wing) is raised to operating height. The HMI automatically connects and initializes all utilized laser trackers through an application programming interface to SpatialAnalyzer. The trackers are then fit into the established FRS to accurately define each tracker position. Targets are placed at the defined locations on the wing and automatically measured by a tracker using target searching functions to acquire a lock on each target.

Closing to final join

Once each wing has been measured, a calculation is invoked in SpatialAnalyzer by the HMI to compute the best-fit transformation of each measured wing half to 5 in. away from the nominal join position. This is so that the final rigid body move will only require adjustment in one dimension for the two halves to meet, simplifying the final approach. If the best fits for each wing pass a tolerance check, the resultant six-degree-of-freedom (6DOF) transformation matrixes are sent to the work center PLC for a rigid body move of both the right wing and the left wing separately. If either of the best fits does not pass, a tracker reshoots the wing and tries the best fit again. Once the matrix has been sent to the PLC, the operator uses a wireless touchscreen module called a pendant to control the PLC during the move process, as seen in figure 3. The operator can oversee the entire rigid body move from the pendant, control the speed, and monitor forces on each positioner.

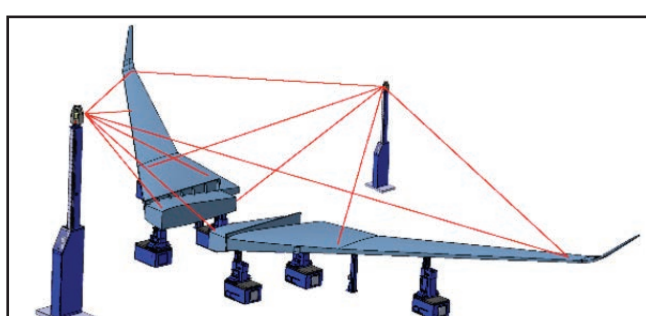


Figure 2. The wing-wing join cell

Figure 3. A wireless PLC pendant

Moving to final join

Once the wings have completed the move to 5 in. from the final join position, the HMI invokes laser trackers to remeasure the wings and calculate new best fits to the final join locations. The HMI reviews and sends the matrixes as before, moving the right wing and then the left. When the join is complete, a tracker measures the final positions of the wings and a tolerance check is computed. If the check fails, the operator must follow additional steps to readjust the wings and try the check again. A typical aircraft assembly process requires drilling and then separation for deburring the holes and cleaning. Accordingly, if the final positions pass the check, the join position is recorded, the positioners separate the wing halves by several feet (for cleaning), and then the halves are returned to join position to be fastened together.

Moving to next work center

Once the wing is fastened together, the entire assembly is transported by a specialized twin tower lifting rig (called an ATLAS), which is used to move the aircraft assemblies between work centers, as seen in figure 4. The ATLAS transports the assembled wing to the wing-fuse work center and unloads it onto the work center-specific positioners (as seen in figure 5) for the next operation, concluding the wing-wing join operation. The ATLAS can be shared between work centers, ferrying assemblies from one to the next.

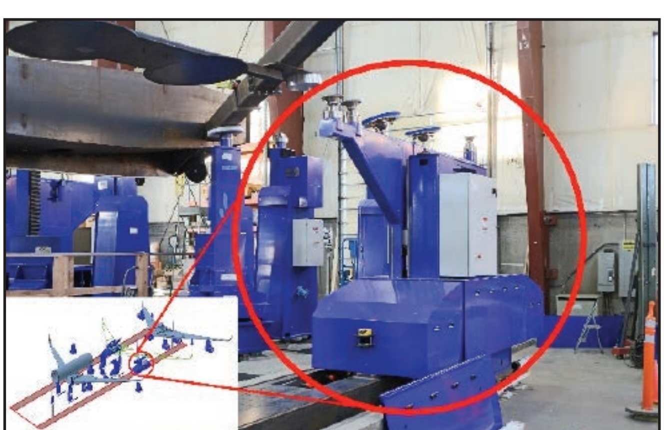


Figure 4. One side of the ATLAS

Figure 5. A wing positioner

DEVELOPMENT

Overview

Although the basic process for each work center remained the same, certain work centers needed additional innovation to meet the assembly requirements. The typical process for a join is summarized here, and additional details of interest are discussed for some of the assemblies.

Transformation matrixes to machine movement

The interpretation and execution of the rigid body move by the PLC is common to each part move. For each move, the work center PC running the HMI sends the PLC a 4 × 4 matrix with position and orientation data for the move. The PLC converts these data into machine movement to drive each utilized positioner.

An unexpected problem arose when it was discovered that the number of PLCs supported significant figures and was insufficient for the desired quality. This became a problem for rotation because values close to 0 or 1 round off, such that subsequent trigonometric functions are inaccurate. The solution was to evaluate parts of the matrix prior to transmission by using the HMI to calculate the necessary angle and send the calculated angle to the PLC along with the transformation matrix.

Iron wing/fuselage

A two-piece iron wing was manufactured to permit complete simulation of all the wing-wing join-alignment processes. The iron wing, as seen in figure 6, was a key development tool

Figure 6. *The iron wing after a join*

because the gap between an untested set of instructions and a fully debugged process can be quite significant. Repeating the join process also enabled us to validate the join math and establish accuracy expectations for the join. Among other features, the iron wing had a butt-line zero-mating face (parting plane) and four precision through-holes that passed through both halves of the wing.

These holes were pinned off, and the configuration of the wing was measured to establish the ideal join condition. An iron fuselage was also fabricated to aid in software development and prove the accuracy of the join. The iron fuselage featured matching machined mating surfaces, allowing for an accurate join to the wing.

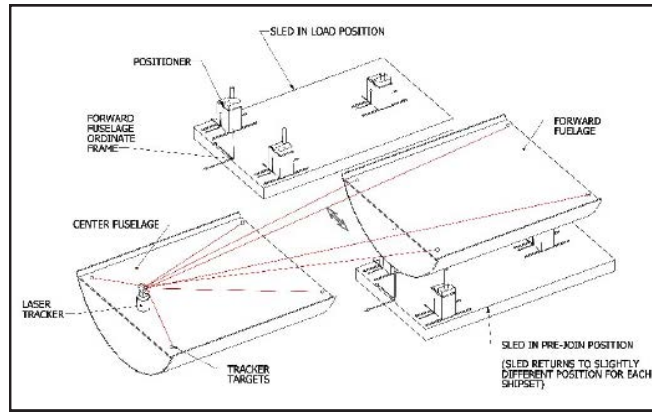


Figure 7. *Schematic showing tracker and target positioning*

ALIGNING WITH A POORLY KNOWN POSITIONER ORIGIN

Tracker positioning

The forward fuselage-to-center fuselage alignment presented a slightly different problem than the other joins. The forward fuselage positioner itself moves in and out of the work center to facilitate part loading. A cup-and-cone system ensures a repeatable return to approximate position, but as a result, the home and orientation of the forward positioner is different for each join, and a variability of up to 0.05 in. in position had to be compensated

Join process	Target points	Largest error
Wing return from deburr	5 on wing	0.0011 in.
Fuselage return from deburr	4 on fuselage	0.0027 in.
Forward fuse move to join	4*	0.0024 in.
Aft fuselage move to join	4*	0.0027 in.

*Figure 8. Table showing process accuracy (*For the fuselage join moves, no appropriate iron fuselage was available. Instead, the vector bars were set on tripods and the center fuselage points were set on towers. This gave an excellent check on the process but not on error to be expected from the target measurements.)*

for by the system. Because it was undesirable to add additional trackers and no existing trackers were adequately positioned to measure the targets on the forward positioner, a single tracker inside the center fuselage was used to execute this join. This tracker has good lines of sight to the center and forward fuselage targets, but has no line of sight to its own positioner or to the FRS targets. The requirement became to achieve the join within these limitations, as seen in the schematic in figure 7.

Measurement solution

Although it was possible to repeat the measure-move process several times to achieve an accurate join, the team sought a more direct solution that would not require multiple iterations. This evolved to a hybrid approach of measurement plus machine action.

Prior to attempting a join, some background measurements and assumptions are required.

- It was necessary to determine the average location of the forward positioner system as it was repeatedly moved in and out of the prejoin position to outside the work center. These measurements permit a reasonable "average" position to be determined. This becomes the nominal location of the forward positioners and is assumed to be the origin of the positioner system for all calculation purposes. (This is a one-time calibration process, not a production process.)
- The *X* and *Y* axes are assumed to be parallel to the work center axes.

Given the above conditions, measurements are then taken of all forward fuselage targets, followed by a long single axis move and then another measurement of all forward fuselage targets.

Using the data from the above measurements, a rotation is calculated to bring the assumed positioner *Z* axis in line with the actual positioner *Z* axis. The same action is repeated with the *Y* axis to calculate the direction of that axis.

Theoretically, it is possible to carry out similar corrections for the rotational axes, but these appeared impractical. Geometry was not adequate for sound correctional rotations, and such further enhancements proved to be unnecessary because good results were achieved without them.

The result of the two axis corrections was that correctional moves were made by the positioner to an excellent level of accuracy, yielding sub-0.005 in. joins with the test setup.

Aft fuselage join—aligning via a vector bar

The aft fuselage join requires three trackers—one behind each wing and one inside the center fuselage. A special problem arose here because the inside tracker shares no common points with the outside trackers. To overcome this limitation, six vector bars were used. The vector bars allow the exterior trackers to be related to the interior tracker, and thus to all the interior points. First, the inside tracker measures the internal aircraft targeting. Next, the outside two trackers, being tied together through an FRS, measure the left and right vector bars, respectively, as well as the left and right external aircraft features, successfully relating all trackers together for accurate positioning.

Defining ordinate frames

Two fundamentally different approaches were considered for calculating the necessary transformation matrixes used in completing each join. Both solutions depend on calculating a coordinate system or an "ordinate frame" for each part to be joined based on measured data. A transformation matrix can then be generated such that the generated frame (and thus the part) is moved to be identical to the frame of the corresponding nominal part location.

Best-fit frames

The first approach, called best-fit frames, involves taking the CAD nominal values for the set of targets and best-fitting the measured data to them. The result is an ordinate frame for the measured data that coincides with the CAD frame. This method is simple and evenly distributes measurement errors across all points.

Constructed frames

The second approach, which is much more complex, is to construct a frame from specific points, lines, and planes created by a specified process. For example, it is possible to note that the left seat tracks were defining elements in previous hard-tooled stations, and create an ordinate frame that reflects this reality. The constructed-frame approach enables tolerances to be tightened around critical features and loosened around other features, as required. It is worth noting that constructed frames will tend to have features set based on just a few measurements, and thus this approach is potentially more sensitive to measurement errors than the best-fit approach.

Join process accuracy

By nature of this application, the practical attainable accuracy remains somewhat elusive. Temperature variance due to thermal growth, distortion, and other factors caused by the frequently opened bay doors all add process variation. Nonetheless, process tests were conducted to develop some rough expectations for accuracy. For the wing-wing join tests, the quality of the join was checked by inserting undersized pins through the alignment holes. Imperfect alignment results in a smaller allowable pin size, providing an accurate alignment measurement. Tested join alignment was ± 0.003 in. for the worst of the four holes. Additional results can be seen in the table in figure 8.

Our expectation is that with daily use, not all results will be as good as those we obtained, especially during days with dramatic temperature shifts. Carelessness with targeting may also add significant errors. The HMI enables rejection of excessive

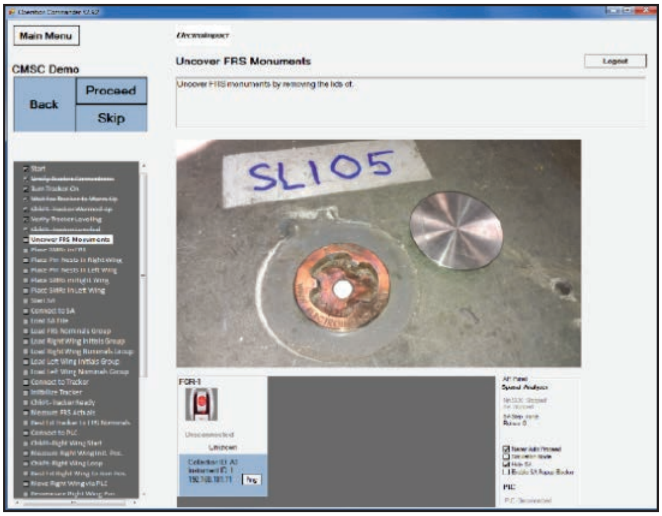


Figure 9. The HMI GUI

errors, so once a production history is established it will be possible to use demanding but not excessively tight tolerances for each application.

HMI

Overview

A custom computer HMI was created to control all automated equipment used in each work center. The operator can oversee an entire join procedure by starting the appropriate set of step-by-step instructions (called a task script) and following the on-screen instructions. The HMI uses an application programming interface to control SpatialAnalyzer for instrument measurement, geometric computations, and quality checks. Minimal SA training is required as the HMI intentionally hides SA and manages all tracker-related functions, providing a simplified user interface for the operator. For PLC communications, a simple transmission control protocol (TCP) is used to pass data to read buffers on the PLC for sending transformation matrixes and PLC commands. For the FAL, it was determined to use Leica AT401/402 laser trackers; however the HMI has already been tested for use on other metrology instruments and has the potential to support communication with any instrument with which SA can interface.

Figure 10. A best-fit result screen

Strong graphical interface

The HMI presents a cohesive interface for the operator to follow each task. Key features of the graphical user interface (GUI) can be seen in figure 9 and include written instructions for the current task, an image to assist with the instructions, and a checklist showing all tasks in the task script. To complete a task, the operator reads the instructions and carries out any manual actions as necessary before pressing the "proceed" button to continue. In addition to written instructions, tasks can control SA or PLC functions. Figure 10 shows an example of an automated best-fit task. For such a task, the HMI invokes SA to compute the best-fit transformation between two groups of points. Instead of an image being displayed, the transformation matrix and deviations appear along with the status of the best fit. The user can review the results and proceed to the recommended action to ensure quality.

When a check fails, the HMI regresses back to a checkpoint in the spreadsheet where the operator can redo previous tasks, such as remeasuring points, and attempt the check again. When a check passes, the user may choose to regress again to improve results or continue to the subsequent task in the script.

1	Task Title	Task Text	Task image filename	Method call	Arg_1	Arg_2	Arg_3	Arg_4	Checkpoi	Collection ID	Instrument ID	Nominals	Actuals	Proceed T	Auto_Pro	Funct_Before_BTN
14	Place SMRs in Left W	Place SMRs in the following l	SMRinWing.png	DisplayOperatorTask	none	none	none	none	none	none	none	none	none	Proceed	FALSE	FALSE
15	Start SA	OpCom will now automatical	InvokeSA.png	StartSA	none	none	none	none	none	none	none	none	none	Proceed	FALSE	FALSE
16	Connect to SA	OpCom will now automatical	InvokeSA.png	ConnectToSA	none	none	none	none	none	none	none	none	none	Proceed	FALSE	FALSE
17	Load SA File	OpCom will now automatical	OpenSAFile.png	OpenSAFile	A3CMSC.xit	none	none	none	none	none	none	none	none	Proceed	FALSE	FALSE
18	Load FRS Nominals (	OpCom will now automatical	OpenPointsFile.png	ImportFromCSV	A3FRSNominals.	Inches	none	none	none	A3	none	FRS_Nominals	none	Proceed	TRUE	FALSE
19	Load Right Wing Init	OpCom will now automatical	OpenPointsFile.png	ImportFromCSV	A3RightWingInit	Inches	none	none	none	A3	none	Right_Wing_Ir	none	Proceed	TRUE	FALSE
20	Load Right Wing No	OpCom will now automatical	OpenPointsFile.png	ImportFromCSV	A3RightWingNo	Inches	none	none	none	A3	none	Right_Wing_N	none	Proceed	TRUE	FALSE
21	Load Left Wing Initia	OpCom will now automatical	OpenPointsFile.png	ImportFromCSV	A3LeftWingInitia	Inches	none	none	none	A3	none	Left_Wing_Ini	none	Proceed	TRUE	FALSE
22	Load Left Wing Nom	OpCom will now automatical	OpenPointsFile.png	ImportFromCSV	A3LeftWingNom	Inches	none	none	none	A3	none	Left_Wing_Nc	none	Proceed	TRUE	FALSE
23	Connect to Tracker	OpCom will now automatical	LeicaConnect.png	ConnectInstrument	FCR-1	FALSE	Leica emScc	TRUE	none	A3	1	none	none	Proceed	FALSE	FALSE
24	Initialize Tracker	OpCom will now automatical	InitializeTracker.png	InitializeInstrument	none	none	none	none	none	A3	1	none	none	Proceed	FALSE	FALSE
25	ChkPt-Tracker Ready	Tracker is connected, initializ	Background.png	Checkpoint	none	none	none	none	none	none	none	none	none	Proceed	TRUE	FALSE
26	Measure FRS Actuals	OpCom will now automatical	TrackerMeasure.png	MeasurePointByPoint	3	none	none	none	ChkPt-Tra	A3	1	FRS_Nominals	FRS_Actuals	Proceed	TRUE	FALSE
27	Best Fit Tracker to F	OpCom will now automatical	Transform.png	Transform	0.05	0.05	FRSTransfor	####	none	A3	1	FRS_Nominals	FRS_Actuals	Proceed	TRUE	FALSE
28	Connect to PLC	OpCom will now automatical	PLCConnect.png	PLCConnect	192.168.101.99	2000	2001	none	none	none	none	none	none	Proceed	FALSE	FALSE
29	ChkPt-Right Wing St	Starting the Right Wing Join P	Background.png	Checkpoint	none	none	none	none	none	none	none	none	none	Proceed	TRUE	FALSE
30	Measure Right Wing	OpCom will now automatical	TrackerMeasure.png	MeasurePointByPoint	3	none	none	none	ChkPt-Rig	A3	1	Right_Wing_Ir	Right_Wing_A	Proceed	TRUE	FALSE
31	ChkPt-Right Wing Lo	Beginning of the Right Wing a	Background.png	Checkpoint	none	none	none	none	none	none	none	none	none	Proceed	TRUE	FALSE
32	Best Fit Right Wing t	OpCom will now automatical	Transform.png	Transform	0.05	0.05	WingTransfc	####	none	A3	1	Right_Wing_N	Right_Wing_A	Proceed	TRUE	FALSE
33	Move Right Wing via	OpCom will now automatical	PLCCom.png	PLCTransformMove	none	RightWing	LGT020	none	none	none	none	none	none	Proceed	TRUE	FALSE
34	Remeasure Right Wi	OpCom will now automatical	TrackerMeasure.png	MeasurePointByPoint	3	none	none	none	ChkPt-Rig	A3	1	Right_Wing_N	Right_Wing_A	Proceed	TRUE	FALSE
35	Verify Tolerance	OpCom will now automatical	GroupToGroup.png	FitQuery	0.05	0.05	WingTransfc	none	ChkPt-Rig	A3	none	Right_Wing_N	Right_Wing_A	Proceed	TRUE	FALSE
36	ChkPt-Left Wing Sta	Starting the Left Wing Join Pr	Background.png	Checkpoint	none	none	none	none	none	none	none	none	none	Proceed	TRUE	FALSE
37	Measure Left Wing I	OpCom will now automatical	TrackerMeasure.png	MeasurePointByPoint	3	none	none	none	ChkPt-Lef	A3	1	Left_Wing_Ini	Left_Wing_Ac	Proceed	TRUE	FALSE
38	ChkPt-Left Wing Loc	Beginning of the Left Wing ac	Background.png	Checkpoint	none	none	none	none	none	none	none	none	none	Proceed	TRUE	FALSE
39	Best Fit Left Wing to	OpCom will now automatical	Transform.png	Transform	0.05	0.05	WingTransfc	####	none	A3	1	Left_Wing_Nc	Left_Wing_Ac	Proceed	TRUE	FALSE
40	Move Left Wing via	OpCom will now automatical	PLCCom.png	PLCTransformMove	none	LeftWing	LGT020	none	none	none	none	none	none	Proceed	TRUE	FALSE
41	Remeasure Left Win	OpCom will now automatical	TrackerMeasure.png	MeasurePointByPoint	3	none	none	none	ChkPt-Lef	A3	1	Left_Wing_Nc	Left_Wing_Ac	Proceed	TRUE	FALSE
42	Verify Tolerance	OpCom will now automatical	GroupToGroup.png	FitQuery	0.05	0.05	WingTransfc	none	ChkPt-Lef	A3	none	Left_Wing_Nc	Left_Wing_Ac	Proceed	TRUE	FALSE
43	Create relationship f	OpCom will now automatical	Reporting.png	MakeGroupToGroupRe	FRS_Fit	none	none	none	none	A3	none	FRS_Nominals	FRS_Actuals	Proceed	TRUE	FALSE
44	Create relationship f	OpCom will now automatical	Reporting.png	MakeGroupToGroupRe	Right_Wing_Fit	none	none	none	none	A3	none	Right_Wing_N	Right_Wing_A	Proceed	TRUE	FALSE

Figure 11. A task script

Hiding excess information

One of the many features of the HMI is its simple design that hides unneeded complexity from the operator. Connection maintenance and tracker health are managed in the background of the HMI to allow the operator to focus on the instructions, prompting the operator to troubleshoot only after automated procedures do not succeed in restoring connectivity. Simple indicators in the GUI can be seen at the lower right of figures 9 and 10. The HMI shows the real-time status of SA, retry attempts for the current command, and PLC connectivity. To the left of the indicators is the instrument panel. This shows the basic information of all the trackers being used, tracker IP addresses, connection status, and activity.

Task scripts

The HMI is controlled by high-level instruction sets called task scripts, as seen in figure 11, for easy implementation and complete customization by nonengineers. Each task script is stored in a Microsoft Excel spreadsheet, with each row being a discrete task with a displayed image, instruction paragraph, and title fields in standardized columns, as seen in figure 11. Each row also has a keyword called a "method call" that denotes what command to execute. The HMI reads each command and gathers data from other cells when the task script is loaded. The HMI will execute each command as the operator follows each step. The HMI now has more than 100 different method calls, including user instruction, SA interaction, PLC communication, data reporting, and file importation.

As an example, the method call "Transform," which can be seen in figure 12, uses the data in the "Arg1" cell as the maximum tolerance of the fit, "Arg2" as the RMS tolerance, "Arg3" as a configuration file for other settings, and "Arg4" if remeasuring points is allowed when a best-fit failure occurs. The HMI also uses the "Collection ID," "Instrument ID," "Nominals," and "Actuals" cell values, which directly correspond to parameters in SA needed for the best fit.

Easy task-script writing

The HMI not only executes the task scripts, but also debugs them during development. When a new task script must be written, the author creates a new spreadsheet with the appropriate columns and adds each task to perform. When the task script is ready for testing, the author adds a line to a file that keeps track of all task scripts, entering the file path of the new spreadsheet. When the HMI starts up, it reads the list and allows operators to run the newly added task script. When the task script is opened, the HMI checks it for errors and rewrites the spreadsheet, highlighting bad cells and adding cell comments to explain what was wrong.

Automated health checks

The HMI has several method calls specifically designed to aid in the quality control of every operation and lower the skill level barriers necessary for operators to perform quality fits. As previously seen, the "transform" command is used to invoke SA to compute a best-fit transformation and review tolerance values, prompting the user to measure bad points when the fit fails. In addition, the HMI also supports a drift check, backsight check, planarity check, and a nominal-to-actual check, as seen in figure 13, all with similar error-handling designs.

SA error handling

During SA-related commands, the HMI maintains control over SA through several different means. When the HMI sends a command to SA, SA will execute the command. When it is finished, SA will send back a report indicating success or failure of the command.

1	Task Text	Task Image filename	Method call	Arg_1	Arg_2	Arg_3	Arg_4	Output	Checkpoint	Colle	Instru	Nominals	Actuals	Proceed Te	Auto_Proce	Funct_Before_BTN
28	OpCom will now	none	Transform	0.05	0.05	DefaultTransform.txt	FALSE	none	none	A3	0	FRS_Nomi	FRS_Act	Proceed	TRUE	FALSE

Figure 12. A best-fit task

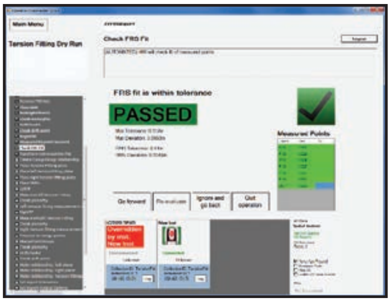


Figure 13. A nominal-to-actual check

The HMI can use these data to determine if the operator can proceed to the next step in the task script, if the command must be retried, or if connections to the tracker or SA application must be restored. If SA cannot complete the command, leaving the HMI without a report, a command-specific timeout trips, alerting the HMI of a command failure. The HMI can then restore SA and try again or prompt the operator for troubleshooting. These features allow the HMI to detect and handle problems such as missing targets or a network failure, while only displaying useful and relevant information to the user.

PLC communication monitoring

During PLC-active operations, the HMI monitors the connection status of the PLC. This is done by performing a series of checks to verify the network location, connectivity, and code status of the PLC by sending ping requests, checking TCP ports, and toggling a "heartbeat" bit to check read-and-write capabilities. When a join move is pending, the HMI waits until the join move is completed before allowing the user to continue to the next step.

FUTURE WORK

Current work remains two steps removed from final validation. Actual aircraft parts are not yet available for testing, and actual operators are not yet available for training and performance evaluation of the system as a whole. Unknown unknowns will be revealed when these limitations are removed, and no doubt further challenges will have to be met.

CONCLUSION

In this article, we automated an FAL by using adaptive tooling techniques and created a unique HMI that communicates with SpatialAnalyzer to support an expansive list of instruments and close the control loop with PLC positioners, while presenting the operator with cohesive directions through a customizable instruction set stored in a spreadsheet. Our results show that the approach and implementations for the FAL can be used to produce quality manufacturing in an industrial work environment, while deskilling the procedures so that minimal training is required for operation.

ACKNOWLEDGEMENTS

The authors would like to thank the rest of the Electroimpact team and our development partners, Tom Kinnare and Kevin Cruickshanks, both of ECM Global, for their many contributions to this project.