2012-01-1887

On the Development of Predictive Simulation Methods for Automated Fastening

Brett Malone, Rick Guptill and Yash Khandhia
AC&E, Inc.
Paul Lindstadt, Dean Cross and Viet Hoang
Spirit AeroSystems, Inc.

ABSTRACT

Offline programming and hardware simulation has been undergoing rapid and dramatic changes as software technologies have provided advanced capabilities. The application of simulation to verify specific autofastening operations has become a timesaving capability to maintain production rates as well as avoid costly collisions and subsequent downtime. Through the use of simulation, engineers can develop additional insight into the autofastening process, including useful parameters that affect maintenance, cost, and cycle time reductions. The work documented here outlines several advances that have been made including the software simulation methods developed and implemented, and the resulting operational benefit achieved. Specific areas that are covered include operation tracking and reporting, predictive wear for tooling, fastener-specific operations, tool path optimization.

INTRODUCTION

Simulation of complex manufacturing operations has evolved to become a necessary tool for production engineers and machine programmers [1]. The use of virtual models to simulate the operation of complex and expensive equipment gives the engineer the ability to test, verify, and optimize difficult operations offline without risking damage to machines and parts.

The use of composite materials has driven the need for more precise operations given the increasing cost associated with the typical composite aircraft component. In addition, an increase in requirements for production rates has driven more automation from the fastening and drilling machine providers [2,3]. This in turn has created demand for more programming and verification.

With this increased production rate and automation comes added requirements for quality assurance and checks [4]. Large aircraft assemblies that have over 10,000 drill and fastener locations create the opportunity for errors at the point of entry. These errors include duplicate operations, missing operations, fastener deletions, incorrect fastener types, or missing drill operations.

The purpose of this paper is to describe methods that have been developed to provide engineers and programmers the ability to verify program operations at each point in the assembly. In addition, these methods can provide the added benefit of computing and predicting tooling parameters such as drill life and cycle times for optimal replacement periods.

Our research objectives encompassed the creation of a software capability that would provide engineers with program verification, reporting on operations at each point, and estimates of tooling wear. The implications of this work have led to more robust programs and more knowledge of tooling aspects, providing opportunity for optimal manufacturing environment.

By focusing on software that simulates the operations of the Broetje automated ring riveter in place at Spirit Aerosystems, the research team was able to develop accurate kinematic movements coupled with a visual system to provide an off-line program creation and validation tool for the Spirit programmers. Furthermore, by incorporating knowledge of the engineering data from the part files, we were able to determine the location and type of operation for each fastener. Taken together, this information forms the basis for an embedded tool that provides engineers with reporting features to support quality audit procedures.

BACKGROUND

The use of highly automated equipment such as complex multi-axis drilling and fastening systems and robotic placement facilities has driven production rates and allowed for the reduction of human operations. Greater precision and repeatability can be gained when deploying hardware to support such operations. In addition, human operators are removed from potentially dangerous operations in cramped locations of integrated assemblies.

However, with the growth of automation techniques comes the need for more efficient programming methods. Activities such as development of machine instructions, validation and verification of those instructions, collision detection, and final QA of the program before deploying to the physical operating environment all provide important information before the physical part is engaged.

As aerospace structures increase in cost due to materials and manufacturing steps, it is critical to avoid damaging collisions between the end effector and the part surface. Large assembly parts can include over 10,000 locations for drilling and fastening, as prescribed by detailed engineering drawings.

In addition, highly engineered, precise machines and specialized end effectors represent capital assets whose operating time must be maximized. To take an operating machine offline for development testing purposes represents a reduction in capacity and an inefficient use of a prime asset.

Earlier versions of robotics and automation equipment could be manually programmed through activities such as teach-pendant or manually controlled positioning [5]. These methods are suitable for small, simple parts [6], but one is quickly overwhelmed with the time consuming activity for more complicated parts. Errors in the programming when using manually guided hardware are frequent and one runs the risk of part collision and/or missing fasteners or operations.

The advent of 3D modeling introduced the ability to create accurate models of workcells and the interactions of the machines in the working environment. This level of accuracy allowed for more refined development of collision detection models, providing engineers with a way to route toolpaths and operations more effectively. This modeling capability also provided the basis for building accurate representations of tooling and fixtures to support flexible machine design. Current simulation software now allows users to create automation programs in a virtual environment and test the output of the program without engaging costly hardware and parts before the program is verified. Figure 1 provides an example of such modeling software where a fully integrated workcell is used to test programs.

As modeling technology matured, so also did the simulation of the kinematics of the machines [7]. Knowing the precise joint behaviors and having accurate inverse kinematics for the machine drivers allowed for the execution of the NC or robotic program entirely in virtual space. Coupled with the 3D models of the workcell, the kinematically accurate models of the machines gave engineers the ability to program, test, and correct operations before reaching the production floor. Figure 2 demonstrates the ability of the software to model and control each joint angle and demonstrate the working envelope for reachability studies as well as coverage and path planning for collision detection.

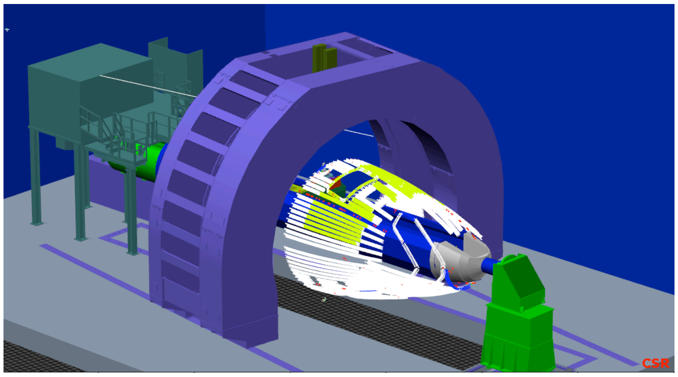
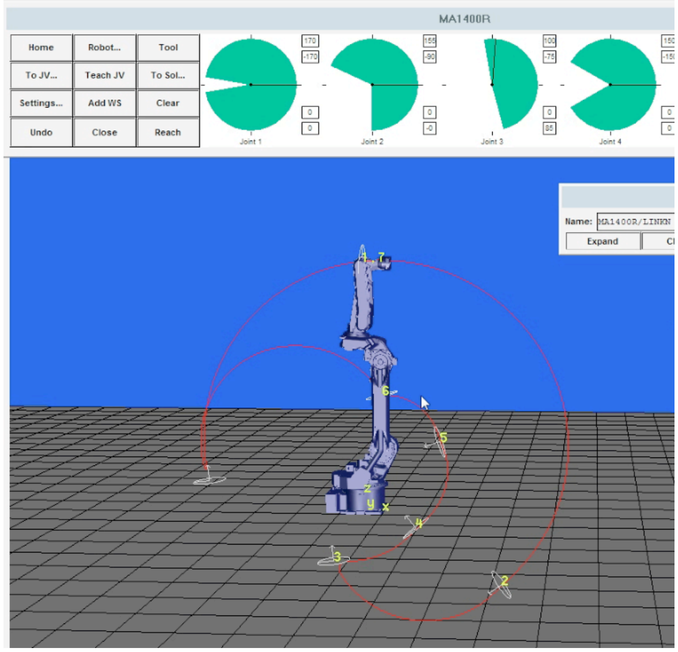


Figure 1. Development of 3D modeling provided tools for interference and collision detection.

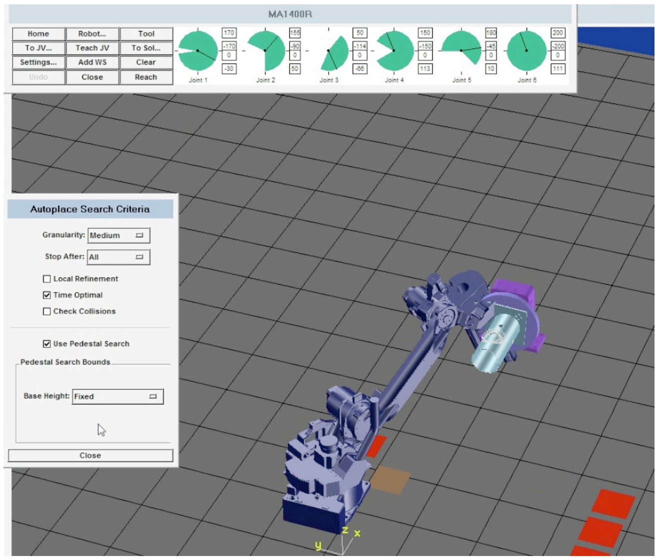


Name: MA1400R/LINKN

Figure 2. Advanced kinematics analysis coupled with controller emulators provide a suitable environment for off-line program development.

METHODS ENABLING ADVANCED ANALYSIS

With the advent of software libraries for integration (such as CAD data, object models, and discrete event information) the use of simulation software for program verification has expanded into both a design and a predictive tool. Engineers can now use simulation and kinematic models to design machines and workcells for optimal performance (Figure 3).



Granularity: Medium
Stop After: All
☐ Local Refinement
☑ Time Optimal
☐ Check Collisions
☑ Use Pedestal Search
Base Height: Fixed

Figure 3. Advanced methods such as optimal placement analysis and joint angle visualization have helped engineers apply simulation software tools across the design-to-verification spectrum.

In addition, rich CAD standards have allowed for the seamless integration of engineering data, robotic models, tooling, and additional fixtures into a fully operational virtual workcell. This is a critical feature should one desire a successful application of off-line virtual environments. Typically the engineering information comes from one group, tooling and fixtures from another, and the robot and controller data from a third source. A key enabling advance for building virtual workcells efficiently is the integration of this information through standards and toolkits.

Similar to integrating geometric data, the off-line programming software system serves as a platform that contains kinematic models and end-effector dynamics necessary for accurate modeling of tool paths with proper joint speeds and linear motion. These kinematic models, once integrated into a full-motion workcell, serve as the basis for cycle time estimates and tool-path optimization.

Further analysis is enabled by the inclusion of additional engineering detail into the part files. By extracting hole placement, drill parameters, and fastener type (length, diameter, etc) the simulation can match the actual operation with the specified operation and report on variances in the productions. Figure 4 illustrates the engineering model with frames and locations of drill and fastener operations. The model is shown mounted on the simulation of the Broetje hardware.

Other enabling technologies include the ability to read program code for incorporation and testing of virtual environments. This code is typically developed by the programmer in other tools and provides an initial NC program for testing. In the case of Spirit, the CSR environment reads the NC programs developed by the programmers and then creates the motions dictated by the program. A virtual model of the Broetje ring riveter is supplied with motion for the inner and outer anvils. The programs developed for both drill and rivet are then executed against that model to verify function for each step.

This combination of integrated geometric data, kinematic models, and program import provides the engineer with a design and development environment capable of program creation, verification, and optimization. The following sections outline a specific project that puts these enabling technologies together in a way to provide production value for automated fastening.

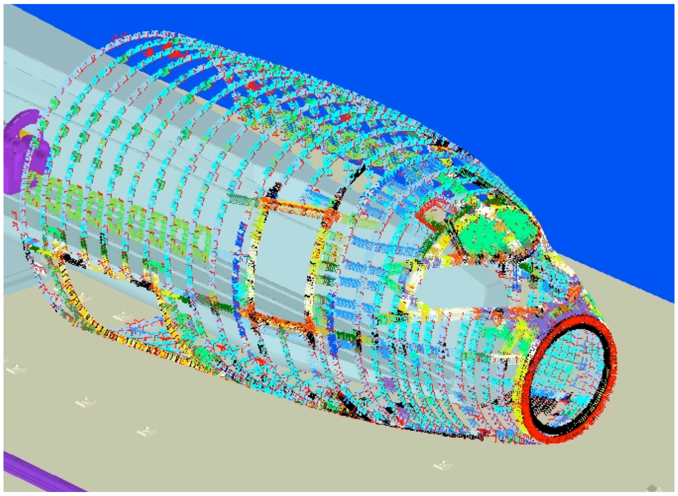


Figure 4. Engineering data that includes items such as fastener detail enables virtual audit of operations.

DEVELOPMENT OF SIMULATION

As part of a working pathfinder for advancing the state of production at Spirit Aerosystems (Spirit), a research team comprised of engineers from Spirit and AC&E, Inc. began investigating the use of an off-line programming and simulation environment to verify automated fastening operations as specified by in-house developed programs. The existing software platform, CimStation Robotics (CSR®) [8], was in use by Spirit engineers and AC&E worked to make enhancements to the software to enable reporting of operations and predictive estimates for tooling life. Figure 5 shows the operating end effector for the Spirit autofastening system. The CSR simulation was developed to model this assembly and provide a rapid way of feeding back program verification to the engineer.

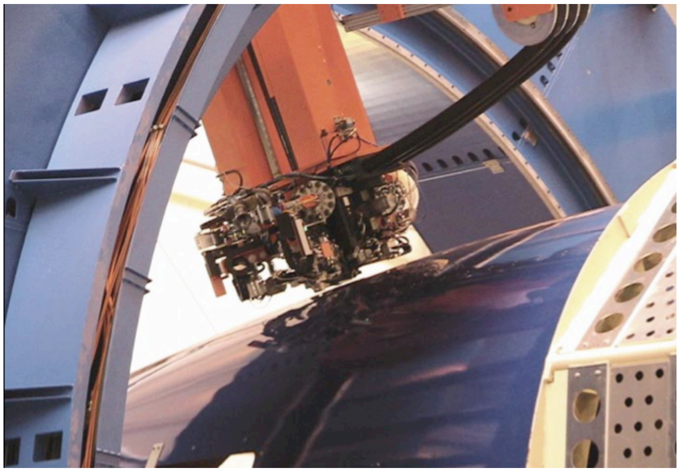


Figure 5. Operating end effector unit of the Broetje autofastening system. Engineers at Spirit use predictive simulation to verify material, stock requirements, and operations at each frame of operation to ensure program quality and consistency.

DESIGN DRIVERS

Spirit Aerosystems is a major manufacturing partner to Boeing, Airbus and other top aircraft producers with the development of large, complex aerospace structures. As part of an on-going quality assurance program, the developers of Spirit machine programs are continuously seeking new ways to verify consistent operations such as hole placement, fastener location, and drill site operations. A key driver for the project was the ability to create specific reports that highlighted the operations conducted at each frame location. In addition, the team also desired to demonstrate the tooling impact of drilling operations so that stock supplies could be estimated more accurately.

Incorporating quality assurance early in the engineering and planning cycle reduces downstream risk and flags errors earlier for remediation at much lower cost. For example, trapping an unprocessed fastener location in simulation saves time and rework compared to detection at depot or final check before shipment of a fuselage component.

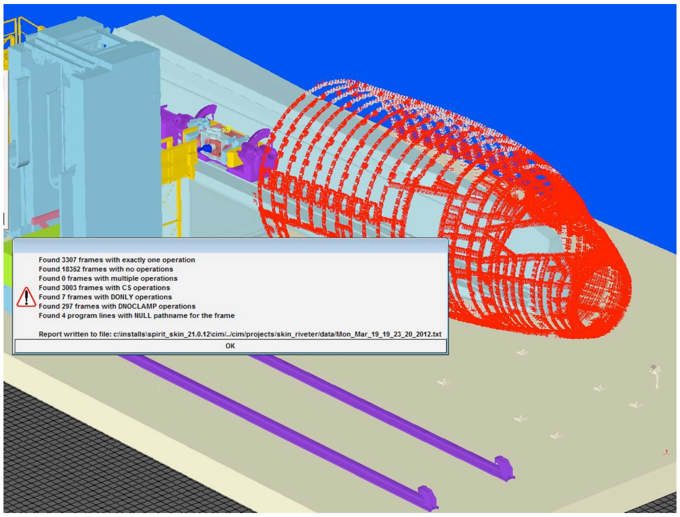
The driver for this project was the need to understand how to inject higher-level verification of the auto-fastening program through simulation before operations begin. A secondary objective of this work was to derive simulation-based metrics that could be used to predict manufacturing needs and optimize processes. Both objectives were driven by the desire for Spirit to improve quality and lower manufacturing cost.

APPROACH AND METHODS

Figure 6 shows an example Broetje integrated cell for operating on fuselage sections. This “ring” fastener system allows for rapid drilling and insertion of fasteners over a complete fuselage section. The CSR software system from AC&E was used to model this machine and provide accurate kinematics for all motions relative to drill and fastener placement operations.

Figure 6. Broetje Fuselage Operating Cell.

AC&E utilized the placement of frames as an anchor to introspect the NC programs for each operation. By reading the CATIA V5 data model for the engineering part, we were able to place each operation within the specified frame for the program. Then, the frame was given a tag for the operation (drill, fastener, etc..) and batched for reconciliation in a final report. Figure 7 shows the frames attached to the engineering model in proximity to the location of the specified operation.



```
Found 3307 frames with exactly one operation
Found 18352 frames with no operations
Found 0 frames with multiple operations
Found 3003 frames with CS operations
Found 7 frames with DONLY operations
Found 297 frames with DNOCLAMP operations
Found 4 program lines with NULL pathname for the frame

Report written to file: c:\installs\spirit_skin_21.0.12\cim/../cim/projects/skin_riveter/data/Mon_Mar_19_19_23_20_2012.txt
```

Figure 7. By using the frames specified at each location, the user can cycle through all points to verify operation.

A new interface was created within the CSR environment to allow for tracking of the frames and each operation. Using the CATIA model as a basis from engineering operations, the frame placement formed the template against which programming verification could take place. By coupling the NC program input to the CAD model and the known frame location, the differentials can be computed between the expected operation and the actual operation that was programmed.

The known kinematics for the Broetje machine provided an additional level of verification for tool path analysis. Using the simulation of the machine, the program for autofastening operations was executed *in silico* to provide a spatial analysis of the trace of the end effector. This provided the team with an understanding of path planning and any possible reverse runs or cross-overs that may limit the machine's cycle time.

Collision detection and avoidance was facilitated through the execution of the NC program in the simulation. By understanding the tool trace and the path for the drill/fastener hardware, we could assess potential collisions and flag the user for path rework. This approach is standard and existed prior to the current work, however, the path planning and optimization was integrated with the collision detection to allow a seamless program verification strategy.

Reporting was facilitated through special output that was created to show the linear operations and the total counts from each frame. This output was used by the operator to understand what material was being drilled and the location with the total amount of distance travelled by the end unit. By reporting operations metrics in this manner the research team expected to provide other key departments within manufacturing with information to drive stocking and resupply needs.

RESULTS

Using the new algorithm in the CSR framework, AC&E engineers provided Spirit with initial reports. Spirit engineers used the reporting algorithms to verify actual programs for benchmarks and testing. The testing consisted of evaluating known part count files and comparing with frame counts for the program as sequenced through the virtual equipment.

Program Validation

Initial tests were run with standard NC programs. Follow-up tests were run with more complete validation of program points using engineering data that included fastener information for operations at each location. Once the programs were verified, the fastener locations were verified for correct placement in the program.

Operations for Frames

Table 1 shows an example output report with the operations at each frame and the resulting frame count. The program evaluates operations at each point based on a frame-by-frame assessment of drill function, type, and fastener operation. The Table shows for the sample part evaluated that 3307 frames resulted in singular operations. No frames showed multiple operations. Of the 3307 frames captured, 3003 were tagged with CS operation, 7 were drill only (DONLY) and 297 were specialty operations signifying Drill No Clamp (DNOCLAMP).

Table 1. Output from program verification highlighting activity occurring at each frame (drill/fasten point).

`Output Description`	**`Summary`**
	`Found 3307 frames with one operation.`
	`Found 18352 frames with no operations.`
	`Found 0 frames with multiple operations.`
	`Found 3003 frames with CS operations.`
	`Found 7 frames with DONLY operations.`
	`Found 297 frames with DNOCLAMP operations.`

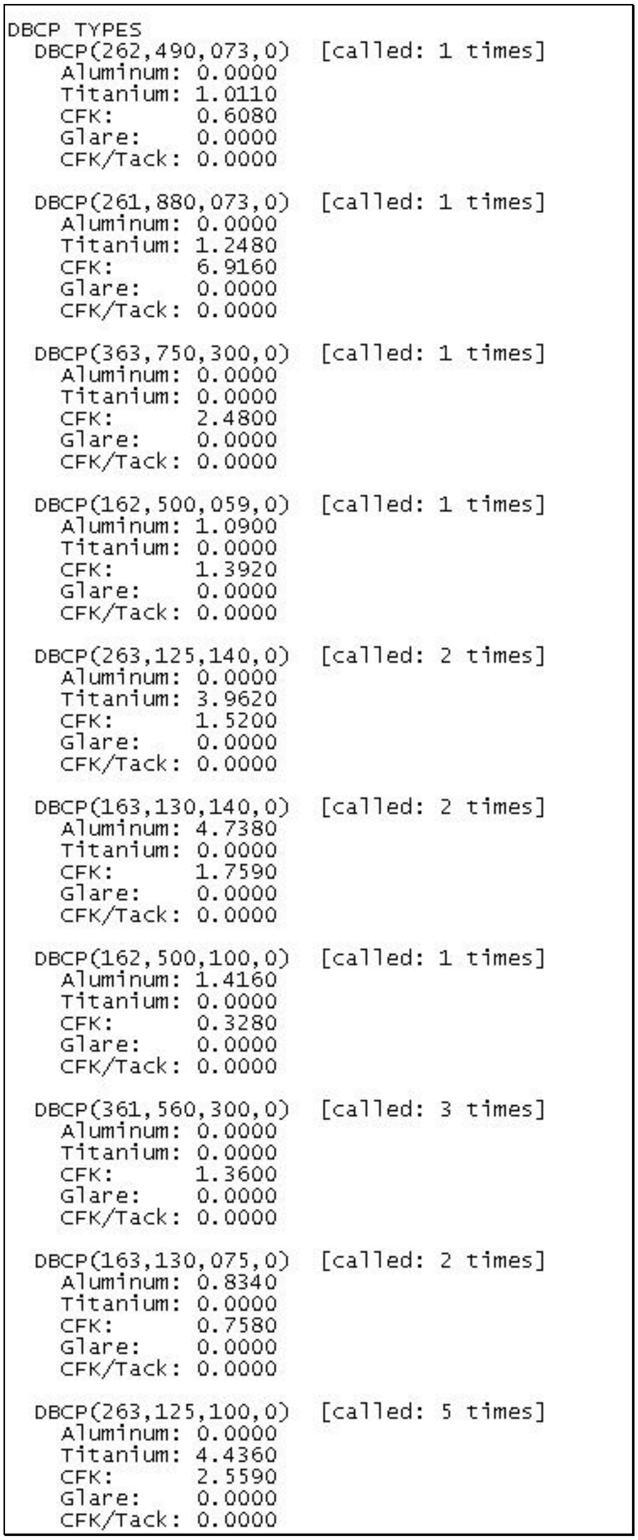
With the ability to evaluate operations at each frame, one can begin to record the total results of each operation type and form an overall estimate of movements for the machine as well as consumables occurring at each station and overall for the entire part. Once the basic reporting capabilities were established and in place, the research team then sought to create assessment data based on totals from operations at each location.

Drill Life Estimates

Table 2 shows the output for the report that highlights the operation types at each location. The reporting algorithm polls the part at each frame location for the material type and the thickness to create a composite output of distance drilled and the material type at each operation. Using this information a summary of each material can be provided for each operation. As shown, material summaries for Aluminum, Titanium, CFK, Glare, and CFK/Tack has been developed. The report also shows where each operation has been called and the number of times utilized by the program.

The drill life report predicts where multiple materials are drilled and gives the actual linear distance for each material. Coupling this information with a wear-life model opens the possibilities for more detailed analysis of duration of drill life and material requirements for stocking and logistics.

Table 2. Program Output for Drill Lifetime Report.



```
DBCP TYPES
  DBCP(262,490,073,0)  [called: 1 times]
    Aluminum: 0.0000
    Titanium: 1.0110
    CFK:      0.6080
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(261,880,073,0)  [called: 1 times]
    Aluminum: 0.0000
    Titanium: 1.2480
    CFK:      6.9160
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(363,750,300,0)  [called: 1 times]
    Aluminum: 0.0000
    Titanium: 0.0000
    CFK:      2.4800
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(162,500,059,0)  [called: 1 times]
    Aluminum: 1.0900
    Titanium: 0.0000
    CFK:      1.3920
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(263,125,140,0)  [called: 2 times]
    Aluminum: 0.0000
    Titanium: 3.9620
    CFK:      1.5200
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(163,130,140,0)  [called: 2 times]
    Aluminum: 4.7380
    Titanium: 0.0000
    CFK:      1.7590
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(162,500,100,0)  [called: 1 times]
    Aluminum: 1.4160
    Titanium: 0.0000
    CFK:      0.3280
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(361,560,300,0)  [called: 3 times]
    Aluminum: 0.0000
    Titanium: 0.0000
    CFK:      1.3600
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(163,130,075,0)  [called: 2 times]
    Aluminum: 0.8340
    Titanium: 0.0000
    CFK:      0.7580
    Glare:    0.0000
    CFK/Tack: 0.0000

  DBCP(263,125,100,0)  [called: 5 times]
    Aluminum: 0.0000
    Titanium: 4.4360
    CFK:      2.5590
    Glare:    0.0000
    CFK/Tack: 0.0000
```

Using this data, engineers can begin to analyze the required stock and drill material components for the manufacturing operations. With the information gathered from the sample program, engineers at Spirit are now collecting metrics from actual operations to compare the accuracy of the distance, material type durations, and the stock materials required for the drilling operations for each major part and part type.

Tool Trace

Understanding the path of the tool, location of the drill operation, and the actual trace of the machine helps with overall optimization of path planning. The research team's work with the report generator resulted in a tool trace method where the frames are red and silver. The tool trace helps with machine motion efficiency and the color frames identified that a drilling operation was programmed at that location.

The benefit of the tool trace is to highlight positions with single or multiple operations. Also, the operator is shown the path for the tool and any directional changes, impingements, or overlaps that may occur. Figure 8 shows an example of the tool trace with the colored highlights for operations as indicated.

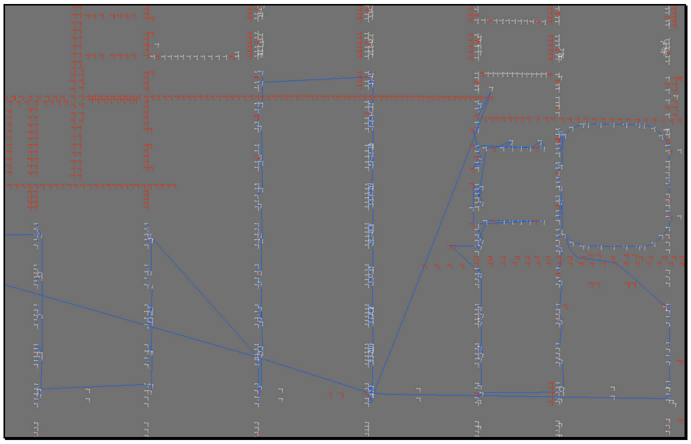


Figure 8. Tool trace shows frames (red/silver) where drill/fastener locations are programmed.

SUMMARY/CONCLUSIONS

Using the new software approach, engineers have been able to develop reports detailing specific location-based operations that validate programmed autofastening routines. In addition, the identification of operations at each specific location provides for meaningful and traceable information for quality assurance purposes. The methods developed by the research team have provided engineering with useful reports for coverage of fastener locations, fastener types, and drill operations. Information is now provided for improved wear prediction and operational cycle times; both valuable components required for any advanced manufacturing facility seeking to optimize production assets.

FUTURE WORK

Work will continue on the simulation environment to enhance operation reporting for other engineering data including less robust engineering master models used in older aircraft structure definitions. The investigators will also seek to expand the predictive capabilities of the simulation to support additional production line analysis such as tooling stock and fastener size optimization. Integration of the drill life model based on material type will provide additional level of fidelity to the reporting as well. Finally, integration of the engineering CAD model with the drill life prediction could give manufacturing engineers a useful tool for cost estimation prior to job start. This could help drive cost down on projects and provide more accurate estimates based on the operations at each fastener location.

REFERENCES

1. Fujiuchi, M., Nakamura, T., Yamaguchi, M., Ishiguro, Y. et al., "Development of a Robot Simulation and Off-Line Programming System," SAE Technical Paper 922120, 1992, doi:10.4271/922120.
2. Devlieg, R. and Szallay, T., "Applied Accurate Robotic Drilling for Aircraft Fuselage," *SAE Int. J. Aerosp.* 3(1):180-186, 2010, doi:10.4271/2010-01-1836.
3. Endres, T., "Advanced Robotic Fastening Machine," SAE Technical Paper 922413, 1992, doi:10.4271/922413.
4. Holden, R., Lightowler, P., and Brady, N., "Robot Integrated Metrology for Complex Part Manufacturing," SAE Technical Paper 2010-01-1859, 2010, doi:10.4271/2010-01-1859.
5. Hewitt, B. and McKeown, S., "The Introduction of CNC Auto-Fastening to the Fokker 100 Wing Assembly Line," SAE Technical Paper 941850, 1994, doi:10.4271/941850.
6. Bafunno, F., "Robotic Sealant Application," SAE Technical Paper 872280, 1987, doi:10.4271/872280.
7. Rasaiah, J. and Jones, A., "Robot Replacement using Simulation and Calibration," SAE Technical Paper 2000-01-2736, 2000, doi:10.4271/2000-01-2736.
8. AC&E, Inc., *CimStation Robotics Operators Manual*, 2010.

CONTACT INFORMATION

Primary author: Brett Malone, Ph.D., brett.malone@acel.us

ACKNOWLEDGMENTS

The authors would like to thank Spirit Aerosystems engineering and production for providing models and simulation data for verification. The authors would also like to acknowledge The Boeing Company for the use of 787 models and the Broetje Company for integration of models for the ring autofastening system.