SUPERVISORY CONTROL AND HEALTH MONITORING FRAMEWORK FOR LARGE-SCALE ADDITIVE MANUFACTURING SYSTEMS

Andrzej Nycz
William Carter
Mark Noakes
Luke Meyer

April, 2021
DOCUMENT AVAILABILITY


Website [www.osti.gov](http://www.osti.gov)

Reports produced before January 1, 1996, may be purchased by members of the public from the following source:

National Technical Information Service

5285 Port Royal Road

Springfield, VA 22161

Telephone 703-605-6000 (1-800-553-6847)

TDD 703-487-4639

Fax 703-605-6900

E-mail info@ntis.gov


Reports are available to DOE employees, DOE contractors, Energy Technology Data Exchange representatives, and International Nuclear Information System representatives from the following source:

Office of Scientific and Technical Information

PO Box 62

Oak Ridge, TN 37831

Telephone 865-576-8401

Fax 865-576-5728

E-mail reports@osti.gov

Website [http://www.osti.gov/contact.html](http://www.osti.gov/contact.html)
Supervisory Control and Health Monitoring Framework for Large-Scale Additive Manufacturing Systems

Authors
Andrzej Nycz
William Carter
Mark Noakes
Luke Meyer
CONTENTS

CONTENTS.......................................................................................................................................... viii
FIGURES AND TABLES........................................................................................................................ ix
ACKNOWLEDGEMENTS ....................................................................................................................... x
ABSTRACT ............................................................................................................................................ 1
1. SUPERVISORY CONTROL AND HEALTH MONITORING FRAMEWORK FOR LARGE-SCALE
ADDITIVE MANUFACTURING SYSTEMS............................................................................................ 1
   1.1 BACKGROUND .................................................................................................................... 1
   1.2 TECHNICAL RESULTS ........................................................................................................ 2
       1.2.1 SIGNALS AND ALARMS ............................................................................................. 2
       1.2.2 FRAMEWORK ARCHITECTURE .................................................................................. 5
       1.2.3 FRAMEWORK MODULES ........................................................................................... 7
       1.2.5 DATA STORAGE .......................................................................................................... 10
       1.2.6 REMOTE MONITORING ............................................................................................. 10
   1.3 IMPACTS ............................................................................................................................ 11
       1.3.1 SUBJECT INVENTIONS ............................................................................................... 11
   1.4 CONCLUSIONS ................................................................................................................. 12
2. PARTNER BACKGROUND ............................................................................................................. 12
FIGURES AND TABLES

Table 1. Signals Acquired and Processed by the Supervisory Framework ........................................... 4
Table 2. Framework Alarms ................................................................................................................... 4
Figure 1. Framework Hardware System.................................................................................................. 5
Figure 2. Framework Planned Architecture .......................................................................................... 6
Figure 3. DCAF Editor Panel (Host and Target Modules on the Left) ..................................................... 7
Figure 4. Primary Framework User Interface ....................................................................................... 9
Figure 5. Additional User Interface Information .................................................................................. 9
Figure 6. Data File Created by the Supervisory Framework ................................................................. 10
Figure 7. SystemLink Software Dashboard for Remote AM Process Monitoring ................................. 11
ACKNOWLEDGEMENTS

This CRADA NFE-19-07585 was conducted as a technical collaboration project within the Oak Ridge National Laboratory (ORNL) Manufacturing Demonstration Facility (MDF), sponsored by the US Department of Energy Advanced Manufacturing Office (CPS Agreement Number 24761). Opportunities for MDF technical collaborations are listed in the announcement, “Manufacturing Demonstration Facility Technology Collaborations for US Manufacturers in Advanced Manufacturing and Materials Technologies,” posted at http://web.ornl.gov/sci/manufacturing/docs/FBO-ORNL-MDF-2013-2.pdf. The goal of technical collaborations is to engage industry partners to participate in short-term, collaborative projects within the MDF to assess applicability of new energy efficient manufacturing technologies. Research is sponsored by the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, Advanced Manufacturing Office, under contract DE-AC05-00OR22725 with UT-Battelle, LLC. This work is further supported by Lincoln Electric.
ABSTRACT

ORNL worked with NI to develop a large-scale, complex additive manufacturing (AM) systems framework for remote health monitoring and supervisory control. We found the framework, based on the Lincoln Electric Metal AM system located at ORNL’s MDF, capable of controlling the process, leading to improved part quality and digital twin creation.

1. SUPERVISORY CONTROL AND HEALTH MONITORING FRAMEWORK FOR LARGE-SCALE ADDITIVE MANUFACTURING SYSTEMS

We began this phase I technical collaboration project (MDF-TC-2019-158) with NI, a large business, on January 15th, 2019, and concluded it on October 31st, 2020.

1.1 BACKGROUND

NI brings together people, ideas, and technology so forward thinkers and creative problem solvers can take on humanity’s biggest challenges. From data and automation to research and validation, NI provides the tailored, software-connected systems engineers and enterprises need to Engineer Ambitiously™ every day. For this MDF, we used NI data acquisition and control system hardware and software.

AM systems of the future will require advanced process-logging, controls, and the ability to create “digital twins” to meet quality-control criteria. But current state-of-the-art, large-scale AM systems offer very limited capabilities. And at this point, there are no standards, universal approaches, compatible interfaces, or paths forward. However, robotics, automation, and modern AM sensing stand to propel advancements in this area. The MDF/ORNL has expertise in AM and access to many different printers, while NI offers robotics and data monitoring and control expertise. Working together, we established a universal supervisory control and health monitoring framework for large-scale AM.

At the beginning of this collaboration, ORNL and NI expected to:

- Improve print quality
- Reduce engineer burden by identifying and resolving system faults
- Lower the required effort to oversee each individual system’s operation
- Log data to both provide print-quality reports to industrial customers and give researchers the means to perform fault analysis and analyze print variables for process/system improvements
- Gather feedback to improve NI commercial off-the-shelf software and framework
1.2 TECHNICAL RESULTS

1.2.1 SIGNALS AND ALARMS

One of the project’s first tasks was to create a list of signals, alarms, and system parameters that the framework would collect. Table 1 presents the current list of signals and signal types, and Table 2 shows the list of alarms based on the acquired signals.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Source</th>
<th>Postprocessed</th>
<th>Role</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Time</td>
<td>Robot</td>
<td>No</td>
<td>Log Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Layer Info</td>
<td>Robot</td>
<td>No</td>
<td>Layer Info</td>
<td>String</td>
</tr>
<tr>
<td>Task Info</td>
<td>Robot</td>
<td>No</td>
<td>Type of Activity</td>
<td>String</td>
</tr>
<tr>
<td>User Pause</td>
<td>User</td>
<td>No</td>
<td>Pause Record</td>
<td>Boolean</td>
</tr>
<tr>
<td>Reason for Pause</td>
<td>User</td>
<td>No</td>
<td>Reason for Pause</td>
<td>String</td>
</tr>
<tr>
<td>Torch 1 Wire Info</td>
<td>User</td>
<td>No</td>
<td>Description</td>
<td>String</td>
</tr>
<tr>
<td>Torch 2 Wire Info</td>
<td>User</td>
<td>No</td>
<td>Description</td>
<td>String</td>
</tr>
<tr>
<td>Total Time</td>
<td>DAQ</td>
<td>No</td>
<td>Print Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Uptime</td>
<td>DAQ</td>
<td>No</td>
<td>Active Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Downtime</td>
<td>DAQ</td>
<td>No</td>
<td>Passive Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Current Layer Time</td>
<td>DAQ</td>
<td>No</td>
<td>Amount of Time in Current Layer</td>
<td>Numeric</td>
</tr>
<tr>
<td>Current Layer Time</td>
<td>DAQ</td>
<td>Yes</td>
<td>Last Layer—Amount of Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Current Layer Time</td>
<td>DAQ</td>
<td>Yes</td>
<td>Five Layers Average Time</td>
<td>Numeric</td>
</tr>
<tr>
<td>Feature</td>
<td>Source</td>
<td>Default</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>--------------</td>
<td>---------</td>
<td>------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Torch</td>
<td>Robot</td>
<td>No</td>
<td>String</td>
<td>Torch Choice</td>
</tr>
<tr>
<td>Weld Mode</td>
<td>Robot</td>
<td>No</td>
<td>String</td>
<td>Weld Mode Type</td>
</tr>
<tr>
<td>Shielding Gas</td>
<td>Welder</td>
<td>No</td>
<td>String</td>
<td>Gas Type</td>
</tr>
<tr>
<td>Weld Voltage</td>
<td>Welder</td>
<td>No</td>
<td>Numeric</td>
<td>Weld Voltage</td>
</tr>
<tr>
<td>Weld Current</td>
<td>Welder</td>
<td>No</td>
<td>Numeric</td>
<td>Welding Current</td>
</tr>
<tr>
<td>Weld Current</td>
<td>Welder</td>
<td>Yes</td>
<td>Boolean</td>
<td>Welding ON/OFF</td>
</tr>
<tr>
<td>Wire Feed Speed</td>
<td>Welder</td>
<td>No</td>
<td>Numeric</td>
<td>Wire Feed Value</td>
</tr>
<tr>
<td>Wire Feed Current</td>
<td>Welder</td>
<td></td>
<td>Numeric</td>
<td>Wire Feeder Motor Current</td>
</tr>
<tr>
<td>X Position</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>End Effector X Position</td>
</tr>
<tr>
<td>Y Position</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>End Effector Y Position</td>
</tr>
<tr>
<td>Z Position</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>End Effector Z Position</td>
</tr>
<tr>
<td>Positioner Joint 1</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Positioner Joint 1 Value</td>
</tr>
<tr>
<td>Positioner Joint 2</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Positioner Joint 2 Value</td>
</tr>
<tr>
<td>Search Offset</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Search Offset Value</td>
</tr>
<tr>
<td>Z Correction</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Z Correction Value</td>
</tr>
<tr>
<td>Commanded Speed</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Commanded Arm Speed</td>
</tr>
<tr>
<td>Actual Speed</td>
<td>Robot</td>
<td>No</td>
<td>Numeric</td>
<td>Actual Robot Speed</td>
</tr>
<tr>
<td>Torch Tip 1 Change</td>
<td>User</td>
<td>No</td>
<td>Boolean</td>
<td>Torch Tip Change 1</td>
</tr>
<tr>
<td>Torch Tip 1 Change</td>
<td>User</td>
<td>No</td>
<td>Boolean</td>
<td>Torch Tip Change 2</td>
</tr>
<tr>
<td>Infrared (IR) Image</td>
<td>IR Camera</td>
<td>No</td>
<td>Image</td>
<td>Image</td>
</tr>
<tr>
<td>IR Image</td>
<td>IR Camera</td>
<td>Yes</td>
<td>Max Temperature</td>
<td>Numeric</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>-----</td>
<td>-----------------</td>
<td>---------</td>
</tr>
<tr>
<td>Thermocouples</td>
<td>DAQ</td>
<td>No</td>
<td>Temperature Value from Thermocouples</td>
<td>Numeric</td>
</tr>
</tbody>
</table>

Table 1. Signals Acquired and Processed by the Supervisory Framework

<table>
<thead>
<tr>
<th>Alarm</th>
<th>Signal</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wire Feed Status</td>
<td>Wire Feed Current</td>
<td>Possible Wire Feeding Fault Warning</td>
</tr>
<tr>
<td>Height Control Status</td>
<td>Z Correction</td>
<td>Height Control Fault or Build Layer Growth Fault</td>
</tr>
<tr>
<td>Search Status</td>
<td>Current Task</td>
<td>Bad Part Searches</td>
</tr>
<tr>
<td>Weld in Place</td>
<td>Weld Current and Arm Speed</td>
<td>Weld in Place Fault</td>
</tr>
<tr>
<td>Temperature High</td>
<td>Maximum Temperature</td>
<td>Temperature over User Defined Value</td>
</tr>
</tbody>
</table>

Table 2. Framework Alarms
We acquired and processed the signals above using an NI cRIO-9035 embedded controller, shown in Figure 1, and used the LabVIEW graphical programming environment to create the framework.

1.2.2. FRAMEWORK ARCHITECTURE
Our project aimed to develop and test a modular and easily extendable framework, as shown in Figure 2.

![Diagram](image_url)

**Figure 2.** Framework Planned Architecture

Based on this structure, we chose the NI Distributed Control and Automation Framework (DCAF) platform for building and testing the framework. DCAF, based on LabVIEW, is modular and scalable data acquisition software. A DCAF application consists of one or more modules running on one or more runtime engines. These modules share the data with one another using the DCAF tag bus. We selected DCAF for this project because its modular nature makes it easy to make changes and apply to multiple systems. Instead of having to rewrite lots of code, we were able to use the configuration manager to easily reconfigure the system to change what data we’re collecting, or even which system we’re monitoring. Figure 3 shows the DCAF platform editor.
1.2.3 FRAMEWORK MODULES
We developed DCAF modules for both the local host computer (operator interface) and remote target (the CompactRIO device).

HOST COMPUTER
- **UI Module**—Displays information from the tag bus on the user interface
- **UDP Module**—Passes small pieces of data, such as numerics and strings, between the PC and CompactRIO device
- **E2E Tag Exchange Module**—Passes larger pieces of data, such as images, between the PC and CompactRIO device
- **User Prompt Module**—Hands over user-prompted information requests triggered by modules running on the CompactRIO device to the PC

REMOTE TARGET (COMPACTRIO DEVICE)
- **Flir IR Module**—Handles IR camera setup and data processing
- **Welder Communication Module**— Receives data from the welder
- **Wolf Communication Module**—Receives data from and sends commands to the robot
• **UDP Module**—Passes small pieces of data, such as numerics and strings, between the PC and CompactRIO device
• **E2E Tag Exchange Module**—Passes larger pieces of data, such as images, between the PC and CompactRIO device
• **SystemLink Module**—Passes data to the SystemLink™ software server for remote monitoring
• **Movement Filter Module**—Filters out movement that occurs when the weld is off for better process monitoring
• **IMAQdx Module**—Uses the NI-IMAQdx driver to configure and acquire images from GigE compatible cameras
• **Layer Timer Module**—Provides information relating to uptime, downtime, and layer times
• **Tip Change Module**—Records tip changes
• **Range Check Module**—Checks if values are within a given range
• **Saturation Check Module**—Checks control signals for saturation
• **Weld in Place Detection Module**—Looks at the welder and robot status to determine if a weld-in-place error is occurring
• **Print Log Module**—Creates a log containing information that doesn’t change in the middle of a build, such as consumables, slice parameters, weld parameters, and build plate data
• **Data Recording Module**—Records data received from other modules
• **Save Image Module**—Saves images of the printed part

1.2.4. USER INTERFACE
A critical component is the user interface or UI module, which is presented in Figure 4.

The primary UI tab gives quick access to the most important data for monitoring build progress. The system displays the current build status, including the current bead and layer, current task, weld mode, and shielding gas, as well as data from the robot, including its current speed and position, and the torch that is currently in use. Also, the system creates and displays a plot of the current robot movement speed versus the error in build height. By monitoring this plot during the build, we quickly can determine if the system is making the necessary adjustments for a high-quality build, or if operator intervention is required. Additionally, the system displays an IR image of the current layer, along with any thermocouples in use. This, along with the built-in alarm, directly affects print quality and increases operator awareness.
In addition to the primary UI tab, there are tabs that give more in-depth looks into the current process (Figure 5). These include a tab dedicated to weld data, IR imaging tabs for both a...
top-down view and side view, a tab for setting alarm parameters, and a tab displaying a
detailed build-time breakdown.

1.2.5. DATA STORAGE
The system stores all data in three file types:

- **Data File**—Basic build info in header followed by real-time build data recorded at 1 Hz
  (rate can be adjusted as needed) shown in Figure 6
- **Print Log File**—Detailed settings including build plate info, consumable info, slicing
  parameters, and detailed weld parameters
- **Images**—Build progress images automatically taken by the system

---

![Figure 6. Data File Created by the Supervisory Framework](image)

1.2.6. REMOTE MONITORING
Finally, we achieved remote monitoring over multiple systems using the NI SystemLink
software platform, which helps remote operators easily check system status. The system
sends instrumentation data—including robot, weld, and task data—to the SystemLink
software server and displays it on a customizable dashboard, as shown in Figure 7.
1.3 IMPACTS

By properly monitoring the complex processes of large-scale AM, we:

- Quickly developed a new AM method and material
- Created a new standard for logging and data analytics
- More quickly identified system faults
- Reduced feedstock usage through more efficient process control
- Lowered operator device monitoring and technical expertise requirements

NI expressed interest in a phase 2 of the project, in which we expand and test the framework on multiple systems with varying configurations. As such, we have created a foundation for future AM direct energy deposition system products.

1.3.1 SUBJECT INVENTIONS

N/A
1.4 CONCLUSIONS

We created and tested a modular supervisory framework using an NI data acquisition hardware and software platform. We implemented the framework on the existing Lincoln Electric ARC1 system located at the ORNL Manufacturing Demonstration Facility. We use it to gather process data from different subcomponents, present the data, store the data for later postprocessing (digital twin), and remotely access the system if needed. We also can use it to control signals to be transferred back from the control modules (user or algorithmic) to the printer. Using the modular DCAF architecture, we can achieve quick deployment and changes without costly code adaptations.

2. PARTNER BACKGROUND

For more than 40 years, NI, a global company headquartered in Austin, TX, has developed automated test and automated measurement systems that help engineers solve the world’s toughest challenges.