Smart and Sustainable Manufacturing Systems

doi:10.1520/SSMS20190046 / Vol. 4 / No. 2 / 2020 / available online at www.astm.org

REVIEW PAPER

Roby Lynn,[1] Moneer Helu,[2] Mukul Sati,[3] Tommy Tucker,[4] and Thomas Kurfess[5]

The State of Integrated Computer-Aided Manufacturing/Computer Numerical Control: Prior Development and the Path Toward a Smarter Computer Numerical Controller

Reference

R. Lynn, M. Helu, M. Sati, T. Tucker, and T. Kurfess, "The State of Integrated Computer-Aided Manufacturing/Computer Numerical Control: Prior Development and the Path Toward a Smarter Computer Numerical Controller," *Smart and Sustainable Manufacturing Systems* 4, no. 2 (2020): 25–42. https://doi.org/10.1520/SSMS20190046

Manuscript received September 11, 2019; accepted for publication February 18, 2020; published online April 1, 2020. Issue published July 29, 2020.

[1] George W Woodruff School of Mechanical Engineering, Georgia Institute of Technology, 771 Ferst Dr. NW, Atlanta, GA 30332, USA (Corresponding author), e-mail: roby.lynn@gatech.edu, https://orcid.org/0000-0002-0198-9100

[2] Systems Integration Division, National Institutes of Standards and Technology, 100 Bureau Dr., Gaithersburg, MD 20899, USA

[3] School of Interactive Computing, Georgia Institute of Technology, 771 Ferst Dr. NW, Atlanta, GA 30332, USA

[4] Tucker Innovations, Inc, Waxhaw, NC 28173, USA

[5] Manufacturing Demonstration Facility, Oak Ridge National Laboratory, 2370 Cherahala Blvd. NTRC-2, Knoxville, TN 37932, USA

ABSTRACT

Current industrial practice in automated manufacturing operations relies on low fidelity data transmission methods between computer numerical control (CNC) machine tools and the computer-aided manufacturing (CAM) systems used to program them. The typical language used to program CNC machines, known as G-Code, has been in existence for nearly sixty years and offers limited resolution for command data. In addition, the proprietary nature of industrial CNC systems hampers the ability of manufacturers to expand and improve upon the capability of existing machine tools. G-Code was not designed to support transmission of feedback data, and thus both the CAM system and higher level organizational control systems are frequently blind to the state of the production process. In response, separate standards that enable data exchange with machine tools have been used by industry, such as MTConnect and Open Platform Communications Unified Architecture. However, these standards enable data pathways that are independent of the G-Code command data pathway, and thus they provide practically no means to affect the state of a process on receipt of feedback data. As a result, control and data acquisition exist in separate realms, which makes the implementation of self-optimizing smart CNC systems challenging. This state-of-the-art review surveys existing methods for data transmission to and from machine tools and explores the current state of so-called integrated CAM/CNC systems that enable more thorough control of the machining process using intelligence built into the CAM system. The literature survey reveals that integrated CAM/CNC systems are impeded both by the data exchange methods used to interface with CNC systems in addition to the proprietary and closed architecture of the CNC systems themselves. Future directions in integrated CAM/CNC research are identified based on the requirements identified for such systems.

Keywords

computer-aided manufacturing, computer numerical control, smart manufacturing, Industry 4.0, Internet of Things, G-Code, machining

Introduction

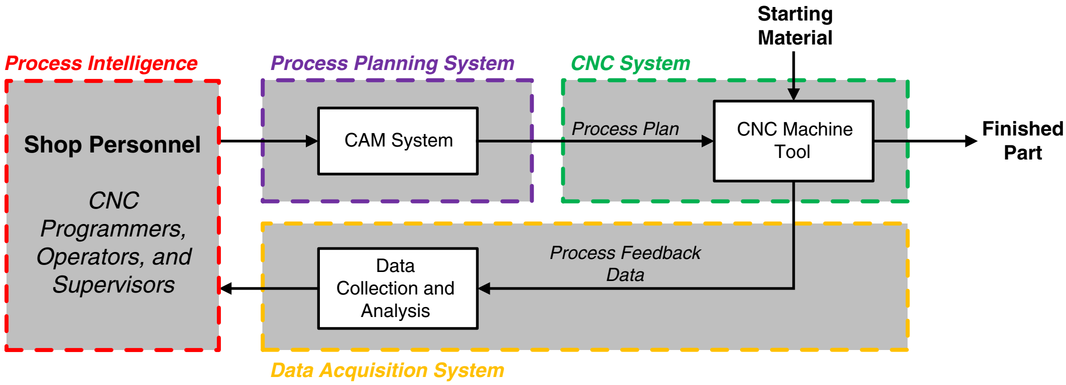
All computer numerical control (CNC) systems for machine tools rely on some method of data exchange with a process planning system to enable transfer of command and control information for realizing a digital representation of a part. The machining process planning system, which is typically referred to as a computer-aided manufacturing (CAM) system, is responsible for the creation of cutting tool motion commands based on starting material condition and desired part geometry. CAM systems customarily provide a computer-aided design (CAD)-like environment for intuitive, interactive manipulation of digital geometric data and also a subsystem for converting the geometric data into motion commands. The resulting motion commands generated by the CAM system are often formatted in some variant of a text-based language known as G-Code, although some alternative methods do exist, such as the STEP-NC standard or proprietary conversational formats. G-Code is the most widely used machine tool programming language, and although portions of the language have been standardized by ISO 6983, *Automation Systems and Integration – Numerical Control of Machines – Program Format and Definitions of Address Words*, many variants of the language are in common use. The complexity of a complete process plan for a given part is dependent on a variety of factors, such as part geometry and machine capability. The process plan represented in G-Code can be thousands or millions of lines of code for a complex part. The process plan represented in STEP-NC consists of abstractions of geometric features to be machined (e.g., a pocket or slot in the case of milling, or a groove or bore in the case of turning).[1] Regardless of the information format used to transfer the process plan, the machine tool is still responsible for interpreting the given commands and converting them to motion trajectories that are suitable for execution by the feed axes of the machine.

Process feedback and monitoring of the machine tool can be enabled using a number of available manufacturing data standards, such as MTConnect or Open Platform Communications Unified Architecture (OPC UA). These standards can be used to provide motion or state information from the machine tool to a monitoring platform over a network connection; the resulting data can then be collected and used for visualization or analysis. Operators, programmers, supervisors, and other shop personnel can use the results of the data analysis to improve process performance in a number of ways, including:

(1) Physical Changes that Affect Process Capabilities: Physical components in the process could be tuned or modified (e.g., the use of different tooling or workholding, change of lubricant type, recalibration of the machine tool), or the capability of the machine tool used to execute the process could itself be changed (e.g., by using a different machine or adding hardware, such as live tooling).
(2) Process Plan Redesign: Specifics of the process plan can be modified, including changes to the toolpath geometry, cutting parameters, or order of operations.
(3) Equipment Utilization: Up-time and overall equipment effectiveness can be increased through personnel and scheduling changes

Redesigning or tuning the process plan is a common way to adjust a process to improve performance. However, such a process plan improvement typically requires the involvement of a CNC programmer to create a new process plan from the CAM system using information collected from analyzing the process feedback data. As a result, significant manual effort must be exerted to optimize a machining process in what is known as an "open-loop" configuration: typical CAM systems require data input only during the process design phase and do not allow for an automated means of altering process decisions based on data collected during machining (see **fig. 1**).

FIG. 1 Traditional open-loop configuration of CAM and CNC systems with external data acquisition.



CAM/CNC integration refers to the idea that the CAM system and the machine tool controller should function as a cohesive unit with automatic, bidirectional data flow of command and feedback information. Such an architecture removes the human-in-the-loop that is present when process plan generation, execution, and analysis are performed using three separate systems. Instead, all control and analysis tasks are performed on the same platform, which enables enhanced control and awareness of the process in question. Such an architecture can provide a host of benefits to a manufacturing process, including:

(1) reducing the time between the identification and resolution of process and equipment issues;
(2) decreasing the cost and increasing the scale of process planning by relying more on the intelligence potentially contained within the CAM system; and
(3) improving part quality and cycle time because of automatic optimization of process parameters.

These benefits can alleviate the burden on the manufacturing engineers responsible for process development and monitoring. CAM/CNC integration is an integral component in implementing cyber-physical systems, smart manufacturing, and Industry 4.0 in a meaningful way on the shop floor.[2,3]

An integrated CAM/CNC system does not necessarily entail the existence of a centralized computing platform for process planning and feedback data aggregation and analysis. Rather, such a system can range from centralized architectures in which both the CAM system and the machine tool controller are operating on the same hardware (e.g., some conversational programming techniques, such as Mazak's Mazatrol system, approach this level of integration, though advanced machining functionality may be limited) to distributed architectures in which a separate system accepts feedback information, such as that obtained by MTConnect, for optimization purposes. Although the exact architecture of an integrated system influences the flexibility of the process planning and analysis capabilities of the architecture, all integrated CAM/CNC systems share the common trait that they enable a more robust and automated means of controlling a machining process than systems with many disparate elements by possessing a direct feedback loop between the system for process planning and that for feedback data aggregation and analysis. The additional automation afforded by an integrated CAM/CNC system can enable manufacturing processes to be controlled in response to machine and quality data feedback with minimal human involvement. This article reviews the current state of integrated CAM/CNC systems and details standards and technology developments to realize such systems. The common traits that are desirable in an integrated CAM/CNC system are extracted from a review of the literature, and the challenges to implement fully integrated systems are explored. Finally, a future vision for these systems is presented using the current trajectory of research.

Control Hierarchy in Manufacturing Automation

The CNC system is an integral part of a larger process planning and execution chain, which can be described using ANSI/ISA-95, *Enterprise-Control System Integration*. This standard defines the organizational, operational, and process control subsystems and interconnections of an automated manufacturing process.[4] The CNC system connects the process control level to the manufacturing process itself (i.e., it is a bridge from the cyber world to the physical world) and is responsible for the physical control of the machining process. CNC systems are monitored by supervisory control and data acquisition systems. Operational control is performed by a manufacturing execution system (MES), which is responsible for routing and ensuring the successful completion of orders through the factory. Toolpath generation for the CNC system is performed by the CAM system, which resides in the operational control level. The business level houses the enterprise resource planning (ERP) system, in addition to CAD and product lifecycle management systems. This control hierarchy is illustrated in figure 2.

THE DIGITAL THREAD

The overarching theme of improving data flow between top-level planning systems and the shop floor environment can be encapsulated in the digital thread concept, which extends model-based enterprise (MBE) concepts through the entire process planning and execution chain. According to Hedberg et al.,[5] the digital thread "would enable real-time design and analysis, collaborative process-flow development, automated artifact creation, and full-process traceability in a seamless real-time collaborative development among project participants." MBE is the practice of using digital models instead of analog drawings and unorganized part requirements to drive the manufacturing enterprise. For example, a model-based definition would include all of the data necessary to manufacture a given part such as geometry, lifecycle information, manufacturing instructions, and inspection data.[6] The digital thread concept has been described in many different contexts, such as STEP Application Protocol 242,[7] defense,[8] additive manufacturing,[9] robotic systems using Robot Operating System,[10] and machining.[11] Commercial software products that leverage the digital thread concept to enable near-real-time (RT) simulation of a machining process (referred to as a "digital twin of the machining process") have also appeared, such as NC.js, which is maintained by STEP Tools, Inc.[12]

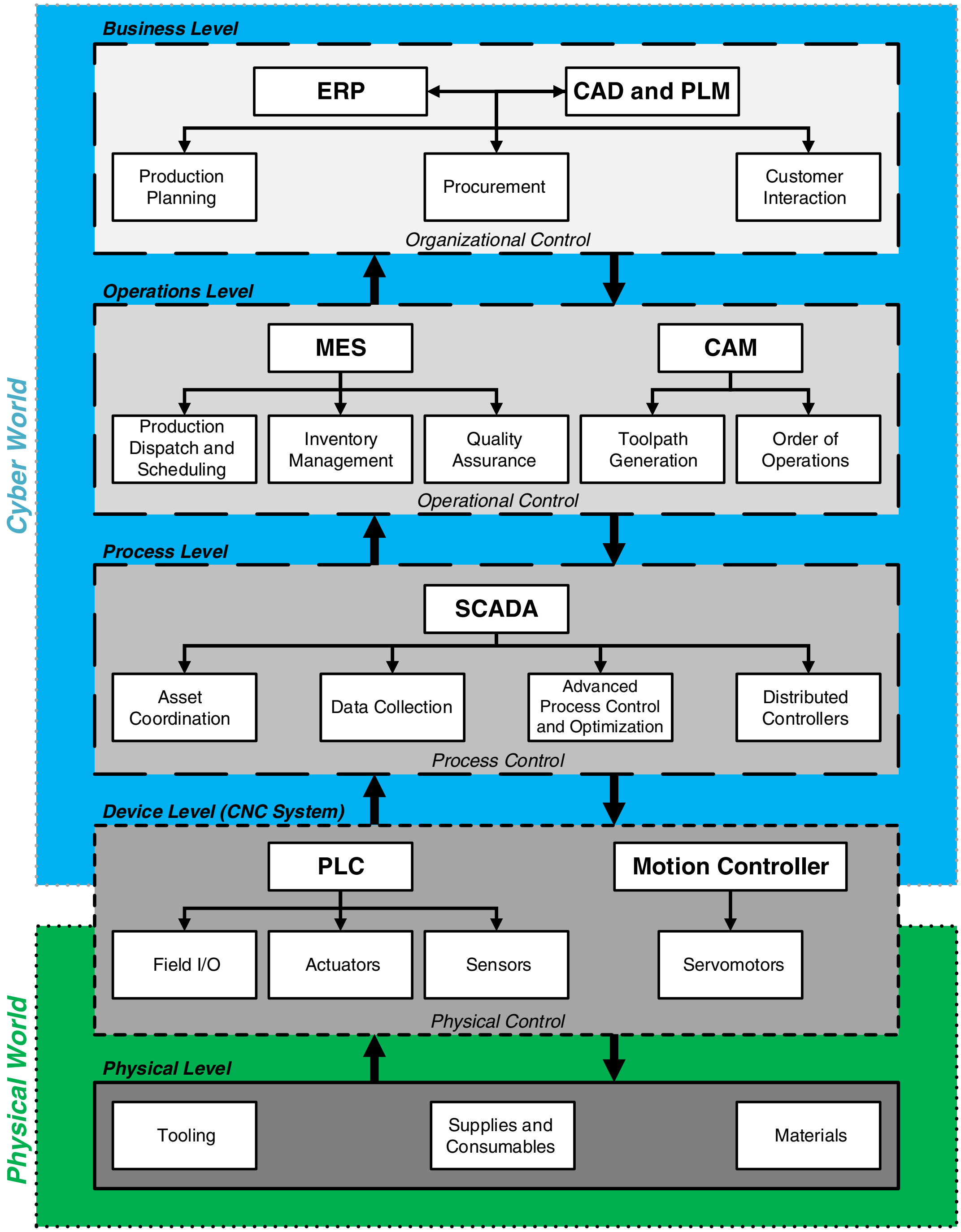
Control Data Generation and Transmission

The interconnection of the components in the process planning and execution chain requires numerous data formats and communication protocols. Some of these protocols exist only to support the traditional view of process planning and execution shown in figure 2. Of particular interest are those formats and protocols that are used to transfer data between CAM systems and CNC machine tools. Specifically, ISO 6983 (G-Code) is the industry standard for sending toolpath data to CNC machine tools, though STEP-NC is a feature-based standard that has been used primarily in the research community.

CAM

Much of the intelligence in machining operations lies at the level of the process designer, who is responsible for converting the desired part to be machined into a complete and functional process plan that defines the order of operations and sequence of machine movements necessary to machine the part. The designer must not only have an intimate knowledge of the capability of available equipment and tooling but also possess an innate understanding of the mechanics of machining to be capable of developing an efficient and robust process plan suitable for production. To aid in development of the plan, the designer will typically rely on a CAM system that can create toolpaths using various cutting strategies (e.g., two-dimensional pocketing, three-axis surfacing, five-axis swarfing). The designer interacts with the CAM system graphically and relies on both experience and training to pick suitable tooling and toolpath geometry for a given part.

FIG. 2 Process planning and execution chain (after ANSI/ISA-95).



Numerous CAM vendors exist on the market today, and each will frequently introduce enhancements to toolpath generation to improve machining efficiency. However, the general nature of the CAM system remains constant among all commercially available solutions: it is an upstream element from the CNC system that creates complex toolpaths from part geometry, (occasionally) automation scripts, and input from an expert operator, where the latter is by far the most important element.

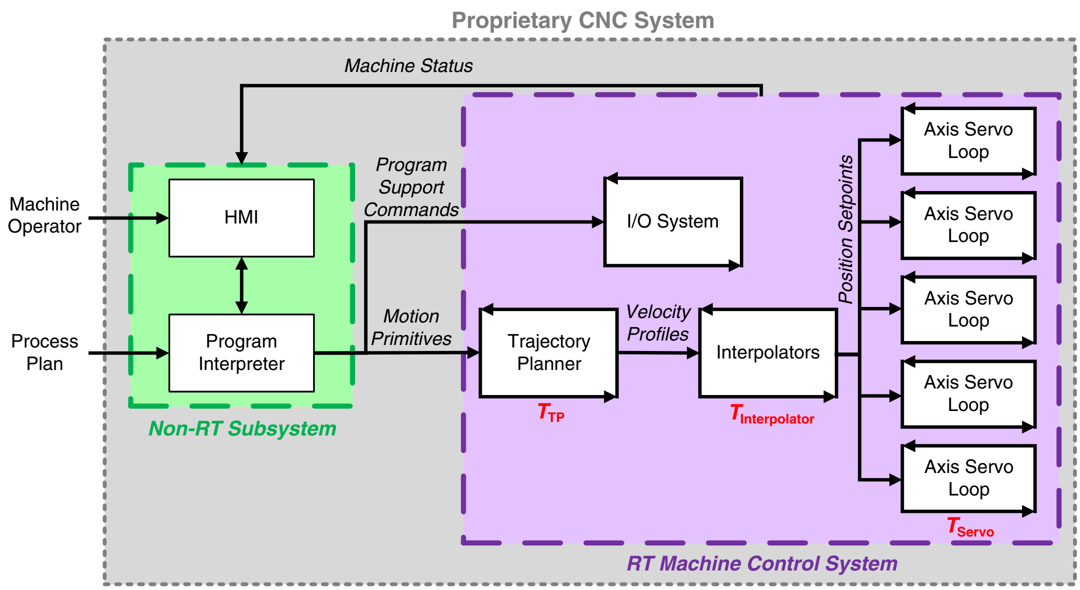
The debugging and optimization of toolpaths generated using CAM can be performed with an NC simulation software, such as Vericut, but frequently also requires execution on an actual machine tool for complete verification.[13,14] However, as machine state and motion information is infrequently relayed up the process chain to the CAM system,[15] the toolpath designer is forced to manually verify the part program at the machine. This can be time-consuming because the designer may have to make several iterations of the process plan design and validation process.

ISO 6983 FOR TEXT-BASED PART PROGRAMS

The most common way to program a CNC machine tool is through a text-based format colloquially known as G-Code, which was originally standardized as RS-274 by the Engineering Industries Association in the 1960s.[16] Eventually, the language was standardized internationally as ISO 6983.[17] A typical G-Code program consists of words and associated data that can denote geometric primitives (e.g., lines or arcs), axis address labels (e.g., X, Y, Z), and miscellaneous commands (e.g., M-Codes that can denote noncutting operations such as tool changes or control of the coolant system). The name G-Code is derived from the fact that the words used to denote motion commands are preceded with the letter G. G-Code programs are created from CAM through the use of a postprocessor, which creates the G-Code necessary for a specific machine tool based on the toolpath generated by the CAM system.

The program is interpreted by the CNC, which then performs the necessary trajectory planning and interpolation of the motion commands to drive the cutting tool along the desired toolpath.[18] **Figure 3** illustrates the functional blocks within a typical CNC system implemented on a commercial machine tool. There are two main elements to the CNC system itself: a non-RT front end that is responsible for servicing the user interface and other low-priority tasks and an RT subsystem that is responsible for controlling the motion, auxiliary, and input/output (I/O) functions of the machine itself. The RT subsystem performs trajectory planning with a period of T_{TP}, which involves fitting and sampling a time-parameterized curve at T_{Servo}, the rate of the axis servo controllers. The curve specifies the motion of the axes of the CNC machine, and the trajectory resulting from sampling the curve is sent

FIG. 3 CNC system architecture. HMI = human machine interface.



to the axis servo controllers to realize geometry conforming to the G-code instructions. Although the architecture of the CNC system is relatively constant among vendors,[19] commercially available systems are frequently proprietary and offer limited facilities for user modification.[20]

Although G-Code is a widely adopted standard, many machine tool builders have supplemented the language with their own custom control codes to expand the capabilities of RS-274 and ISO 6983. These custom codes are output by machine-specific postprocessors that are either purchased from a CAM vendor or created and modified by the CAM programmer. As a result, different CNC systems interpret different dialects of G-Code, making program portability between machines difficult.[21] Forced reliance on postprocessors is a fundamental deficiency in G-Code as a toolpath data format: even with some of the expanded capabilities that are introduced with each new control iteration, the very structure of G-Code sets up a limited data transmission pathway between the CAM system and the CNC itself.[22] From a motion control perspective, this is inherent in the structure of G-Code because G-Code requires that control instructions consist of geometric motion primitives. There are two issues here:

(1) Geometric Data Loss: Although such a decomposition is lossless for some parts that exhibit a high level of geometric regularity, this decomposition essentially involves approximation for free-form parts.
(2) Control Data Loss: G-code syntax does not provide the structure to affect low-level trajectory control.

As an example, consider the trajectory planning and interpretation stages in **figure 3**: the designer of the control system (e.g., a control manufacturer such as FANUC, Siemens, or Heidenhain) determines the trajectory planning strategy to use (e.g., constant-acceleration trajectory planning, constant-jerk trajectory planning, sinusoidal trajectory planning) and also determines the interpolation scheme to use on the resulting trajectories.[23] The CAM user therefore has limited control over the low-level intricacies of the motion of the machine tool.[24,25]

STEP-NC

In response to some of the criticisms of traditional machine tool programming with ISO-6983 compliant G-Code, a new process plan interchange format known as STEP-NC was developed as ISO 10303-238, *Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 238: Application Protocol: Application Interpreted Model for Computerized Numerical Controllers* (or AP238).[26] STEP-NC grew from the need to use standard data at the level of the machine tool controller itself[27] and was accelerated by the standardization of product data in the Standard for Product Model Data (STEP) format (ISO 10303).[28] A STEP-NC process plan is composed of working steps that define features of the process plan (e.g., a pocket); each working step is subsequently composed of machining operations (e.g., in the case of the pocket, the appropriate machining operation would be milling).[29] The complete definition for STEP-NC files includes the machining models defined by ISO 14649, *Industrial Automation Systems and Integration—Physical Device Control—Data Model for Computerized Numerical Controllers*.[1]

As demonstrated by Hardwick and Loffredo,[30] the machining of parts from STEP-NC process plans can enable data interchange between multiple disparate CAM and CNC systems without the use of a traditional postprocessor. However, the current state of CNC systems at the time of this demonstration was such that the AP238 process plan still had to be translated to ISO 6983-compliant G-Code for execution because the machine tools under study did not possess native STEP-NC interpreters. In the years since that demonstration, many researchers have developed successful native STEP-NC interpreters and even fully functional machine tools that can manufacture parts directly from STEP-NC process plans.

One of the key benefits to STEP-NC is the ability to implement data flow from the machine tool back to the CAM system for the purposes of process feedback. This architecture enables the capture of valuable and often underreported input from the machine operator concerning the state of the production process[31] and also provides a means for the CNC system to communicate changes in the process plan that can result from information garnered during production.[32] In contrast with an ISO 6983 program, where changes to the low-level part program can necessitate a complete reworking of the entire part program, the higher abstraction level provided by STEP-NC provides for more modularity in the process plan. However, this introduces two challenges when implementing STEP-NC: (1) the higher level abstraction limits the ability of manufacturers to differentiate their services to customers, and (2) the increased portability and modularity of STEP-NC programs may increase liability to the manufacturer without a clear means of validating the appropriateness of the program for a specific machine.

The practice of using machine and operator feedback within the STEP-NC framework is referred to as a "closed-loop" process or integrated process chain, and researchers are continuing to explore the area. Recent developments include the implementation of native or semi-integrated STEP-NC interpreters for various CNC systems,[33–35] integration of inspection data into the closed-loop manufacturing process concept,[36–39] and synthesis of STEP-NC process data with data from higher-level information management (e.g., Product Data Management, ERP, Manufacturing Execution) systems.[40–42]

Process Data Feedback

The collection of process data from machine tools has historically been a difficult task because control manufacturers did not provide a means for the communication of such information.[43] However, the emergence of standards for data exchange from industrial automation equipment[44] has motivated control builders to implement means for such data export. Two such standards that have been used for machine tool data collection, MTConnect and OPC UA, are gaining traction in digital manufacturing operations today. The data pathways provided by these standards are key enablers of CAM/CNC integration because they provide a means to supply process data to devices upstream of a machine tool.

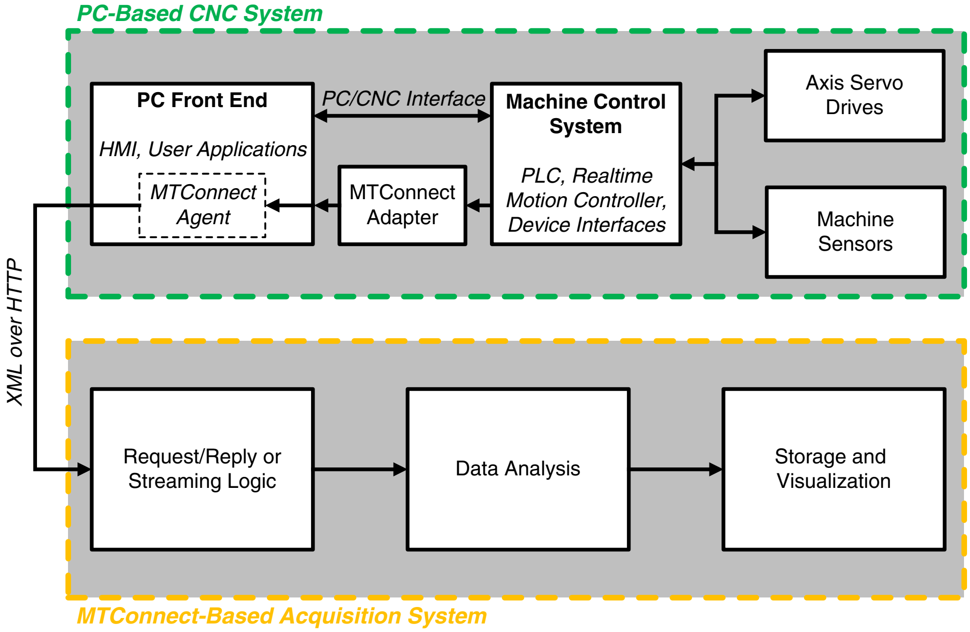
MTConnect

MTConnect is an open, royalty-free, extensible data-interoperability standard that provides a common vocabulary and information models so that manufacturing equipment can generate structured, contextualized data.[45] MTConnect is developed by the MTConnect Institute, which is an ANSI-accredited standards development organization, and it has broad adoption by manufacturing end-users as well as machine and control vendors. An MTConnect-compliant device exposes available data through a piece of software called an MTConnect Agent, which is a special purpose HyperText Transfer Protocol server that provides a representational state transfer interface that a client application uses to request data from the MTConnect-compliant device. For each request, the MTConnect Agent publishes a response document. It also organizes and manages data that may be provided by an MTConnect Adapter, which is an optional tool that collects and filters data about the current state of the MTConnect-compliant device. An MTConnect Adapter is typically a piece of software that interfaces with the machine's control system, but it can also be hardware based for legacy machine tools if the control system cannot support a software adapter. Although MTConnect can enable the streaming of near-RT data as well as polling,[46] MTConnect is strictly a read-only protocol that supports only data collection and not machine command transmission. Implementers of MTConnect-compliant manufacturing systems would need to maintain two separate pathways for data transmission: the forward pathway carries machine commands (e.g., in the form of G-Code), and the feedback path carries process data in the MTConnect format. An example architecture of an MTConnect-based monitoring system with a PC-based CNC is shown in **figure 4**.

Both the research and industrial communities have demonstrated significant interest in deploying MTConnect as a means to collect process data from manufacturing equipment. For example, a large body of work has leveraged MTConnect to collect data using a discrete data acquisition system for the purpose of process improvement by either plant personnel or a supervisory control system.[47–49] Other works have studied the following:

(i) RT machining process improvement using MTConnect data;[50]
(ii) MTConnect-based monitoring of additive manufacturing equipment running on open-source controllers;[51]

FIG. 4 Example of a typical MTConnect system architecture. HMI = human machine interface.



(iii) deployment of Internet-of-Things (IoT) devices for the collection and transmission of MTConnect data;[52–54]
(iv) use of popular open-source software platforms for collecting MTConnect data;[44]
(v) correlation of planned and actual product and process data using MTConnect;[55,56]
(vi) integration of process and metrology data;[57] and
(vii) performance and quality-of-service implications in MTConnect deployments.[58]

Numerous commercial solutions that leverage MTConnect data for process monitoring and dashboard visualization, such as Memex MERLIN, TechSolve ShopViz, FORCAM Force, and System Insights VIMANA, are also in use in production environments.[59]

OPC UA

Another data exchange standard of interest to researchers and developers in industrial automation is known as OPC 10000-1 – Part 1, *OPC Unified Architecture—Part 1: Overview and Concepts*,[60] which provides a platform that enables data exchange between various levels of the process planning and execution chain.[61] In contrast to MTConnect, OPC UA provides syntactic (not semantic) interoperability. OPC UA, which is maintained by the OPC Foundation (where OPC was originally known as Object Linking and Embedding for Process Control, but is now simply Open Platform Communications), is an evolution of the original OPC standard that is based on Microsoft's Distributed Component Object Model (DCOM). OPC UA was developed to address concerns with the proprietary nature of DCOM and to increase the extensibility of the standard to cover additional devices and systems that were not possible to integrate into OPC.[62] OPC UA adopts a service-oriented architecture and defines a standard data format for the exposure of actions and attributes for a compliant device in a unified data model. Communication of OPC UA data is accomplished using either XML (known as UA Web Services) or binary (known as UA Native) communication methods between OPC UA clients and servers. The OPC UA standard defines only the format for messages that are passed between clients and servers and does not provide a standardized application programming interface (API) for implementing a complete OPC UA stack; as a result, it is the responsibility of the systems integrator to develop a suitable API for a given device.[63]

Current research directions with OPC UA have been more varied than those with MTConnect for two primary reasons: (1) the original OPC standard has been in existence for longer than MTConnect, and OPC UA builds upon the momentum of OPC; and (2) the syntactic interoperability provided by OPC UA enables the interconnection of a wide range of devices with user-defined data models.[64] Thus, implementers of OPC UA do not have to rely on the standards development process to add additional data items to the standard and can instead simply define data models as necessary. Although the lack of semantic interoperability when using OPC UA can enable more rapid deployment to a variety of systems, it also does not ensure that all devices conforming to the OPC UA standard can exchange information effectively. As a result, research in the use of OPC UA for control and monitoring of an industrial process includes examples from pharmaceutical manufacturing,[65] aluminum rolling,[66] and power generation and distribution.[67] Research within the discrete manufacturing domain has focused on the following:

(i) development of an architecture to use OPC UA as a means to enable data exchange between vertically separated systems in the process planning, control, and execution chain (e.g., ERP, MES, and CNC systems);[68,69]
(ii) development and implementation of data acquisition systems based on IoT platforms that rely on OPC UA for data transmission;[70,71]
(iii) control and monitoring of a flexible manufacturing system for machining and assembly;[72] and
(iv) construction of predictive models based on process data gathered using an OPC UA stack.[73]

In contrast with MTConnect, OPC UA and simplified versions of the OPC architecture also enable the transmission of control commands to manufacturing equipment, which has been demonstrated as a means to operate machine tools remotely.[74,75]

Efforts Toward CNC Intelligence and CAM/CNC Integration

Disparities between methods for communication of data between process planning systems and machine tool controllers has motivated interest in so-called integrated CAM/CNC manufacturing systems. Such manufacturing systems enable more complete flow of data between the CAM system and the CNC machine tool than is possible with the typical G-Code–based architecture and can therefore enable more complete data flow through the entire process planning and execution chain.[76] Based on a review of the literature, the distinct characteristics that characterize integrated CAM/CNC systems can be grouped into the categories in **Table 1**.

The concepts in **Table 1** are captured in the digital thread concept, which is enabled by open communication standards and technologies. Systems that separately implement some of these characteristics have been demonstrated using the data transmission methods in the surveyed literature. Researchers have incorporated additional intelligence into the process planning and execution chain primarily through the design and implementation of

TABLE 1
Characteristics of an integrated CAM/CNC system

Intelligence	Incorporation of additional intelligence into the planning and execution chain,[95] which can enable automatic process optimization and control
Control	Additional low-level control over both the cutting tool trajectory and the physics of the cutting process than is possible using the control methods popular in literature[20]
Data	Higher availability and automatic archival of fine-grained process data from the machine tool's sensors to enable traceability and historical analysis[95,98,99]
Granularity	Device-level control of machine tool subsystems through the CNC kernel[87] and facility for incorporation of proprietary process intelligence possessed by the machine tool owner
Awareness	Enhanced RT and interactive process awareness for shop personnel and higher level planning systems
Teleoperation	Remote control of assets in a manufacturing environment
Automation	Automatic toolpath generation
Collaboration	Enablement of distributed and collaborative manufacturing[100]

STEP-NC manufacturing systems that enable closed-loop machining using standard or modified versions of STEP-NC.[27,32,77] Enhanced trajectory control has been enabled through the use of custom and fully open architecture control systems[20,78–82] to avoid artificial constraints that are placed on trajectory commands by commercial control manufacturers; additionally, the open-source LinuxCNC and Machinekit projects have been used as the basis for control systems presented in literature.[18] Higher availability and resolution of process data has been accomplished through the construction of MTConnect and OPC-based monitoring systems and accompanying analysis and visualization applications, and the fusion of collected process data with an open-loop process plan has been realized using STEP-NC and MTConnect.[83,84] Device-level control of machine tool subsystems, which is currently hampered both by the means of data transmission to the machine tool as well as the interfaces between the subsystems (e.g., proprietary nature of servo drives) and the CNC kernel,[85] has been explored using heterogeneous motion control hardware configurations,[86] the development of custom RT communication protocols,[87] and the design and implementation of open CNC kernels. Enhanced process awareness for shop personnel and high-level planning systems has been realized through integration of STEP-NC data with MES and ERP systems, trends toward cloud manufacturing, and development of local and web applications for data monitoring and visualization.[88,89] Remote control of manufacturing assets has been accomplished using OPC and OPC-like architectures through local and internet connections,[74,75] and automatic generation of toolpaths has been performed through integrated CAD/CAM systems that create process plans based on part features.[90,91] Finally, distributed and collaborative manufacturing systems have been explored in the context of collaborative robots[92,93] and cloud manufacturing.[94]

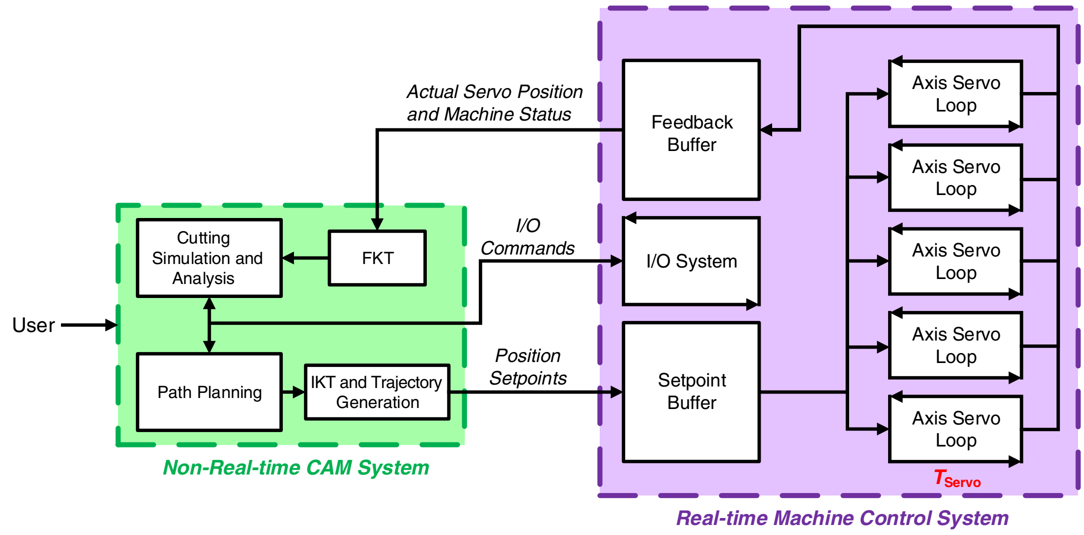
Current Challenges for Smarter CNC Systems

Despite significant effort toward the development of integrated CAM/CNC systems, their current state remains lacking. Systems presented in the literature either do not address each of the desired attributes in **Table 1** or their capability in addressing a certain requirement remains a fundamental deficiency. Attempted implementation of a complete integrated CAM/CNC architecture is frequently hindered by the following.

(1) Closed Architecture: The proprietary nature of commercial CNC kernels or their accompanying I/O and servo control hardware.
(2) Data Availability: Lack of access to certain data or sensor measurements, or no provision for high-frequency data acquisition that is required for thorough process analytics.
(3) Extensibility: Limited capability for modification and incorporation of additional intelligence.

Unfortunately, control and monitoring methods presented in this research rely on smart and extensible controllers. As identified by Xu and He,[95] a major challenge to widespread adoption of STEP-NC lies in the development of intelligent machine tool control systems with integrated CAM functionality to realize cutter motion from STEP-NC data. Michaloski et al.[96] also point out that future CNC systems require intelligence to operate in a factory where they can be presented with missing or incomplete data from process plans or other collaborating pieces of equipment. This challenge is not unique to proponents of STEP-NC: increasing automation in smart factories will inevitably rely on increased intelligence from each asset involved. Higher levels of abstraction in command information, coupled with more conversational requests for production (e.g., “machine this set of features in some order at whatever time is convenient”), will be necessary to emulate the capability of a manufacturing operation that is completely controlled by humans. This level of functionality is not compatible with the commonly accepted architecture for the manufacturing enterprise in **figure 2**. Through future enhancements in widely adopted standards, such as STEP-NC, MTConnect, and OPC UA (in addition to standards and protocols that have not yet enjoyed widespread adoption), manufacturers will be able to enjoy smarter and more automated means of production that come closer to fully realizing the characteristics in **Table 1**. As pointed out by Lu et al., the development and adoption of appropriate standards remains a major research challenge in smart manufacturing system deployment.[97] The success further standards development and enhancement will hinge on

FIG. 5 CAM-controlled CNC system.[101] FKT = forward kinematic transformation; IKT = inverse kinematic transformation.

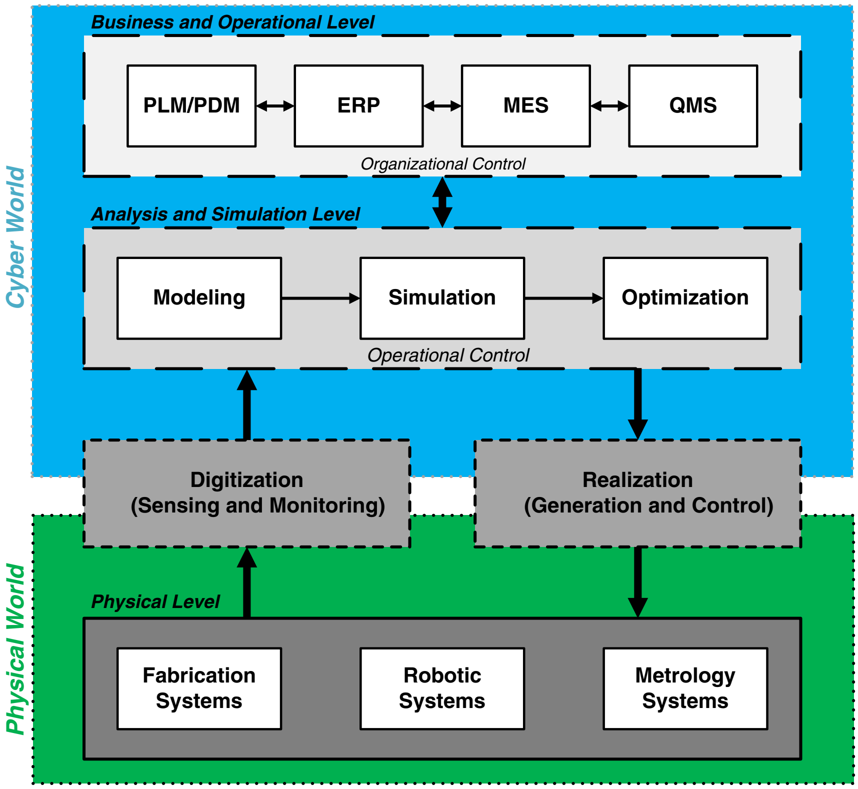


effective collaboration between the builders of machine control systems, researchers, and manufacturers: fully integrated manufacturing systems will not be possible unless all of those involved are willing to drive together toward the goal of a smarter shop floor.

Future Developments

Higher automation in machining will require additional intelligence of machine tool controllers to enable enhanced process awareness, analysis, and automatic optimization. Traditionally, these are in the realm of the CAM system; the machine tool is merely a servant to be controlled by explicit commands from some other system. The fallacy in this architecture lies in the lack of communication between the CAM system and the CNC. Current literature has shown that a major impediment to realization of smarter and more integrated CAM/CNC systems lies both in the proprietary nature of CNC systems themselves and the data pathways that are used to transfer information to and from machine tools, which is called out by various researchers.[20,80,95] The CNC needs more information from the CAM than simple motion commands, as it must be able to react to changing machining conditions in RT. For instance, the CNC system should be aware of the material properties of the workpiece to enable proper control of cutting conditions if some parameters of the process are unexpectedly out of bounds. Likewise, the CAM system needs process information from the machine tool to create and optimize the most effective toolpaths. These requirements necessitate the development of a smarter and more integrated CNC system, where the demarcation between CAM and CNC is blurred and the two function as a cohesive unit. RT process feedback will be provided to the CAM system by the machine controller, which will be used to improve the productivity of the process and the quality of the resulting parts; the CAM system will thus serve as the intelligence of the CNC machine in the integrated architecture. The architecture of such a CNC system is presented in figure 5,[18] where the CAM system takes the place of the non-RT Human Machine Interface (HMI) component in the typical CNC system from figure 3. Instead of interpreting a traditional process plan generated offline by a CAM system, the CNC is controlled directly by the CAM system: all trajectories are generated in the CAM system using the desired part geometry, the forward and inverse kinematic transformations of the machine tool, and the dynamic motion constraints of the machine axes. Process data are fed back to the CAM system from the suite of sensors (including axis position sensors) on the machine tool, enabling robust toolpath analysis and optimization capabilities.

FIG. 6 Fully integrated process planning and execution system (after Hedberg, Helu, and Sprock[102]). QMS = quality management system.



These low-level and high-resolution process data will be available to upper levels of the manufacturing enterprise to enable full process awareness at the operational level. Computer-aided engineering functionality will be incorporated to the CAM/CNC system to enable near-RT simulation of the process for control and learning. The trajectory planner will no longer be a proprietary element of the CNC system and will instead be an open and customizable subsystem that a manufacturer can tune according to process requirements. An illustration of such an integrated architecture that enables complete data flow throughout the process planning and execution chain is presented in figure 6.

The additional intelligence that CNC systems must possess may not necessarily reside on the machine controller itself; with the advent of cloud service providers that offer ever-increasing amounts of computing power and graphics processing unit–accelerated instances, some intelligence can be incorporated into low-latency offsite computing facilities. The distribution of intelligence away from the machine tool will enable further collaboration between both collocated assets in the manufacturing process as well as facilities in different geographic areas. These developments will contribute substantially to the efficiency and productivity of the smart factory but will require smarter, more open, and more extensible CNC systems.

Conclusions

This article summarized the current state of integrated CAM/CNC manufacturing systems, including the technologies that enable such systems and the research efforts currently under way to leverage those technologies to create a smarter shop floor. Much research effort has been devoted to the development and use of standards for use in the manufacturing enterprise, such as MTConnect, STEP-NC, and OPC UA. This research work and the review performed in this article revealed the fundamental characteristics of a truly smart and integrated manufacturing system, as well as the deficiencies in current technologies that must be addressed to realize such a manufacturing system. Openness in data availability and interfaces, coupled with collaboration between equipment builders, researchers, and manufacturers will be required for the eventual realization of an integrated CAM/CNC system that fully realizes all of the fundamental characteristics that were identified in this review.

ACKNOWLEDGMENTS

This work was supported by NIST award 70NANB18H157 and NSF awards CMMI-1646013, IIP-1631803, and DGE-1650044. Certain commercial systems are identified in this article. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the products identified are necessarily the best available for the purpose.

References

1. *Industrial Automation Systems and Integration—Physical Device Control—Data Model for Computerized Numerical Controllers*, ISO 14649 (Geneva, Switzerland: International Organization for Standardization, 2003).
2. J. Lee, B. Bagheri, and H.-A. Kao, "A Cyber-Physical Systems Architecture for Industry 4.0-Based Manufacturing Systems," *Manufacturing Letters* 3 (January 2015): 18–23, https://doi.org/10.1016/j.mfglet.2014.12.001
3. L. D. Xu, E. L. Xu, and L. Li, "Industry 4.0: State of the Art and Future Trends," *International Journal of Production Research* 56, no. 8 (March 2018): 2941–2962, https://doi.org/10.1080/00207543.2018.1444806
4. *Enterprise-Control System Integration*, ANSI/ISA-95 (Research Triangle Park, NC: International Society of Automation, 2010).
5. T. Hedberg Jr., J. Lubell, L. Fischer, L. Maggiano, and A. B. Feeney, "Testing the Digital Thread in Support of Model-Based Manufacturing and Inspection," *Journal of Computing and Information Science in Engineering* 16, no. 2 (March 2016): 021001, https://doi.org/10.1115/1.4032697
6. A. M. Miller, N. W. Hartman, T. Hedberg, A. B. Feeney, and J. Zahner, "Towards Identifying the Elements of a Minimum Information Model for Use in a Model-Based Definition," in *12th International Manufacturing Science and Engineering Conference* (New York: American Society of Mechanical Engineers, 2017), MSEC2017-2979, https://doi.org/10.1115/MSEC2017-2979
7. A. B. Feeney, S. P. Frechette, and V. Srinivasan, "A Portrait of an ISO STEP Tolerancing Standard as an Enabler of Smart Manufacturing Systems," *Journal of Computing and Information Science in Engineering* 15, no. 2 (June 2015): 021001, https://doi.org/10.1115/1.4029050
8. E. M. Kraft, "The Air Force Digital Thread/Digital Twin—Life Cycle Integration and Use of Computational and Experimental Knowledge," in *54th AIAA Aerospace Sciences Meeting* (Reston, VA: American Institute of Aeronautics and Astronautics, 2016), https://doi.org/10.2514/6.2016-0897
9. A. R. Nassar and E. W. Reutzel, "A Proposed Digital Thread for Additive Manufacturing," in *24th International Solid Freeform Fabrication Symposium – An Additive Manufacturing Conference* (Austin, TX: University of Texas, 2013), 19–43.
10. J. Langsfeld, M. Robinson, S. Singh, and W. Sobel, "Enabling Facility-Level Interoperability between Robot Teams and Machine Cell Devices," in *Ninth Model-Based Enterprise Summit* (Gaithersburg, MD: National Institute of Standards and Technology, 2018).
11. T. D. West and A. Pyster, "Untangling the Digital Thread: The Challenge and Promise of Model-Based Engineering in Defense Acquisition," *Insight* 18, no. 2 (August 2015): 45–55, https://doi.org/10.1002/inst.12022
12. STEP Tools, Inc., "NC.js," GitHub, 2017, http://web.archive.org/web/20200311194738/https://github.com/steptools/NC.js/blob/master/README.md
13. D. Jang, K. Kim, and J. Jung, "Voxel-Based Virtual Multi-Axis Machining," *International Journal of Advanced Manufacturing Technology* 16, no. 10 (August 2000): 709–713, https://doi.org/10.1007/s001700070022
14. U. Sungurtekin and H. Voelcker, "Graphical Simulation & Automatic Verification of NC Machining Programs," in *IEEE International Conference on Robotics and Automation* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 1986), 156–165, https://doi.org/10.1109/ROBOT.1986.1087729
15. M. Wosnik, H. Rüdele, and P. Klemm, "Process-Informed Machining Objects for Closed-Loop Process Planning," *International Journal of Computer Integrated Manufacturing* 22, no. 11 (November 2009): 1054–1064, https://doi.org/10.1080/09511920902741109
16. *Interchangeable Variable Block Data Format for Positioning, Contouring, and Contouring/Positioning Numerically Controlled Machines*, RS-274-D (Arlington, VA: Electronic Industries Association, 1979).
17. *Automation Systems and Integration – Numerical Control of Machines – Program Format and Definitions of Address Words*, ISO 6983 (Geneva, Switzerland: International Organization for Standardization, 2009).
18. R. Lynn, M. Sati, T. Tucker, J. Rossignac, C. Saldana, and T. Kurfess, "Realization of the 5-Axis Machine Tool Digital Twin Using Direct Servo Control from CAM," in *NIST Model-Based Enterprise Summit* (Gaithersburg, MD: National Institute of Standards and Technology, 2018), 1–22.

19. G. Pritschow, Y. Altintas, F. Jovane, Y. Koren, M. Mitsuishi, S. Takata, H. van Brussel, M. Weck, and K. Yamazaki, "Open Controller Architecture – Past, Present and Future," *CIRP Annals* 50, no. 2 (2001): 463–470, https://doi.org/10.1016/S0007-8506(07)62993-X
20. Y.-F. Tsai, R. T. Farouki, and B. Feldman, "Performance Analysis of CNC Interpolators for Time-Dependent Feedrates along PH Curves," *Computer Aided Geometric Design* 18, no. 3 (April 2001): 245–265, https://doi.org/10.1016/S0167-8396(01)00029-2
21. D. Odendahl, S. Venkatesh, J. L. Michaloski, and F. M. Proctor, "Standardization of Auxiliary Equipment for Next Generation CNC Machining," in *ISA EXPO Technical Conference* (Red Hook, NY: Curran Associates, 2008), 40–48.
22. J. S. Toquica, S. Živanović, A. J. Alvares, and R. Bonnard, "A STEP-NC Compliant Robotic Machining Platform for Advanced Manufacturing," *International Journal of Advanced Manufacturing Technology* 95, no. 9 (April 2018): 3839–3854, https://doi.org/10.1007/s00170-017-1466-8
23. K. Erkorkmaz and Y. Altintas, "High Speed CNC System Design. Part I: Jerk Limited Trajectory Generation and Quintic Spline Interpolation," *International Journal of Machine Tools and Manufacture* 41, no. 9 (July 2001): 1323–1345, https://doi.org/10.1016/S0890-6955(01)00002-5
24. G. Q. Huang, "Review of Integrating Advanced Computer-Aided Design, Manufacturing, and Numerical Control: Principles and Implementations, by X. Xu," *International Journal of Production Research* 49, no. 11 (March 2011): 3425–3426, https://doi.org/10.1080/00207543.2010.501547
25. M. Tolouei-Rad, "Efficient CNC Milling by Adjusting Material Removal Rate," *International Journal of Mechanical and Mechatronics Engineering* 5, no. 10 (2011): 342–346.
26. A. B. Feeney and S. P. Frechette, "Testing STEP-NC Implementations," in *Fifth Biannual World Automation Congress* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2002), 39–44, https://doi.org/10.1109/WAC.2002.1049418
27. S.-H. Suh, J.-H. Cho, and H.-D. Hong, "On the Architecture of Intelligent STEP-Compliant CNC," *International Journal of Computer Integrated Manufacturing* 15, no. 2 (2002): 168–177, https://doi.org/10.1080/09511920110056541
28. X. Xu and Q. He, "STEP-NC to Re-shape the Manufacturing Industry," in *Fifth International Conference on Frontiers of Design and Manufacturing* (Dalian, China: Dalian University of Technology Press, 2002), 125–131.
29. M. Hardwick, "On STEP-NC and the Complexities of Product Data Integration," *Journal of Computing and Information Science in Engineering* 4, no. 1 (March 2004): 60–67, https://doi.org/10.1115/1.1641188
30. M. Hardwick and D. Loffredo, "Lessons Learned Implementing STEP-NC AP-238," *International Journal of Computer Integrated Manufacturing* 19, no. 6 (2006): 523–532, https://doi.org/10.1080/09511920600627170
31. M. P. Brundage, B. Kulvatunyou, T. Ademujimi, and B. Rakshith, "Smart Manufacturing through a Framework for a Knowledge-Based Diagnosis System," in *12th International Manufacturing Science and Engineering Conference* (New York: American Society of Mechanical Engineers, 2017), MSEC2017-2937, https://doi.org/10.1115/MSEC2017-2937
32. C. Brecher, M. Vitr, and J. Wolf, "Closed-Loop CAPP/CAM/CNC Process Chain Based on STEP and STEP-NC Inspection Tasks," *International Journal of Computer Integrated Manufacturing* 19, no. 6 (2006): 570–580, https://doi.org/10.1080/09511920600622064
33. S. H. Suh, B. E. Lee, D. H. Chung, and S. U. Cheon, "Architecture and Implementation of a Shop-Floor Programming System for STEP-Compliant CNC," *Computer-Aided Design* 35, no. 12 (October 2003): 1069–1083, https://doi.org/10.1016/S0010-4485(02)00179-3
34. X. W. Xu, "Realization of STEP-NC Enabled Machining," *Robotics and Computer-Integrated Manufacturing* 22, no. 2 (2006): 144–153, https://doi.org/10.1016/j.rcim.2005.02.009
35. M. Rauch, R. Laguionie, J.-Y. Hascoet, and S.-H. Suh, "An Advanced STEP-NC Controller for Intelligent Machining Processes," *Robotics and Computer-Integrated Manufacturing* 28, no. 3 (June 2012): 375–384, https://doi.org/10.1016/j.rcim.2011.11.001
36. P. Hu, Z. Han, Y. Fu, and H. Fu, "Implementation of Real-Time Machining Process Control Based on Fuzzy Logic in a New STEP-NC Compatible System," *Mathematical Problems in Engineering* 2016 (June 2006): 9814973, https://doi.org/10.1155/2016/9814973
37. C. Liu, Y. Li, and W. Shen, "Dynamic Feature Modelling for Closed-Loop Machining Process Control of Complex Parts," *International Journal of Computer Integrated Manufacturing* 28, no. 7 (July 2015): 753–765, https://doi.org/10.1080/0951192X.2014.900870
38. P. Hu, Z. Han, H. Fu, and D. Han, "Architecture and Implementation of Closed-Loop Machining System Based on Open STEP-NC Controller," *International Journal of Advanced Manufacturing Technology* 83, no. 5 (March 2016): 1361–1375, https://doi.org/10.1007/s00170-015-7631-z
39. J. Tan, C. Zhang, R. Liu, and X. Liang, "Study on Framework of STEP-NC Controller with On-Machine Inspection," in *International Conference on Artificial Intelligence and Computational Intelligence* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2009), 40–44, https://doi.org/10.1109/AICI.2009.396
40. C. Danjou, J. Le Duigou, and B. Eynard, "Closed-Loop Manufacturing Process Based on STEP-NC," *International Journal on Interactive Design and Manufacturing* 11, no. 2 (May 2017): 233–245, https://doi.org/10.1007/s12008-015-0268-1
41. R. D. Allen, J. A. Harding, and S. T. Newman, "The Application of STEP-NC Using Agent-Based Process Planning," *International Journal of Production Research* 43, no. 4 (2005): 655–670, https://doi.org/10.1080/00207540412331314406

42. R. Laguionie, M. Rauch, J. Y. Hascoët, and S. H. Suh, "An eXtended Manufacturing Integrated System for Feature-Based Manufacturing with STEP-NC," *International Journal of Computer Integrated Manufacturing* 24, no. 9 (July 2011): 785–799, https://doi.org/10.1080/0951192X.2011.592992
43. A. Vijayaraghavan, W. Sobel, A. Fox, D. Dornfeld, and P. Warndorf, "Improving Machine Tool Interoperability Using Standardized Interface Protocols: MTConnect," in *ISFA 2008 International Symposium on Flexible Automation* (New York: American Society of Mechanical Engineers, 2008), 1–6.
44. R. Lynn, W. Louhichi, M. Parto, E. Wescoat, and T. Kurfess, "Rapidly Deployable MTConnect-Based Machine Tool Monitoring Systems," in *12th International Manufacturing Science and Engineering Conference* (New York: American Society of Mechanical Engineers, 2017), 1–10, https://doi.org/10.1115/MSEC2017-3012
45. W. Sobel, *MTConnect Standard Part 1 - Overview and Protocol* (McLean, VA: The Association for Manufacturing Technology, 2012).
46. S. Singh, A. Angrish, J. Barkley, B. Starly, Y.-S. Lee, and P. Cohen, "Streaming Machine Generated Data to Enable a Third-Party Ecosystem of Digital Manufacturing Apps," *Procedia Manufacturing* 10 (2017): 1020–1030, https://doi.org/10.1016/j.promfg.2017.07.093
47. G. Ćwikła, "Methods of Manufacturing Data Acquisition for Production Management – A Review," *Advanced Materials Research* 837 (November 2013): 618–623, https://doi.org/10.4028/www.scientific.net/AMR.837.618
48. B. E. Lee, J. Michaloski, F. M. Proctor, S. Venkatesh, and N. Bengtsson, "MTConnect-Based Kaizen for Machine Tool Processes," in *30th Computers and Information in Engineering Conference* (New York: American Society of Mechanical Engineers, 2010), 1183–1190, https://doi.org/10.1115/DETC2010-28763
49. B. Edrington, B. Zhao, A. Hansel, M. Mori, and M. Fujishima, "Machine Monitoring System Based on MTConnect Technology," *Procedia CIRP* 22 (2014): 92–97, https://doi.org/10.1016/j.procir.2014.07.148
50. F. Ridwan and X. Xu, "Advanced CNC System with In-Process Feed-Rate Optimisation," *Robotics and Computer-Integrated Manufacturing* 29, no. 3 (June 2013): 12–20, https://doi.org/10.1016/j.rcim.2012.04.008
51. X. F. Liu, S. M. N. A. Sunny, M. R. Shahriar, M. C. Leu, M. Cheng, and L. Hu, "Implementation of MTConnect for Open Source 3D Printers in Cyber Physical Manufacturing Cloud," in *36th Computers and Information in Engineering Conference* (New York: American Society of Mechanical Engineers, 2016), DETC2016-60378, https://doi.org/10.1115/DETC2016-60378
52. R. Lynn, E. Wescoat, D. Han, and T. Kurfess, "Embedded Fog Computing for High-Frequency MTConnect Data Analytics," *Manufacturing Letters* 15 (January 2018): 135–138, https://doi.org/10.1016/j.mfglet.2017.11.002
53. R. Lynn, A. Chen, S. Locks, C. Nath, and T. Kurfess, "Intelligent and Accessible Data Flow Architectures for Manufacturing System Optimization," in *IFIP International Conference on Advances in Production Management Systems* (Cham, Switzerland: Springer, 2015), 27–35.
54. M. Parto, M. Dinar, and T. Kurfess, "An MTConnect-Compatible Platform for Secured Machine Monitoring through Integration of Fog Computing, Cloud Computing, and Communication Protocols," in *International Symposium on Flexible Automation* (Tokyo: Japan Society of System Control and Information Engineers, 2018), 329–336.
55. P. D. U. Coronado, R. Lynn, W. Louhichi, M. Parto, E. Wescoat, and T. Kurfess, "Part Data Integration in the Shop Floor Digital Twin: Mobile and Cloud Technologies to Enable a Manufacturing Execution System," *Journal of Manufacturing Systems* 48 (2018): 25–33, https://doi.org/10.1016/j.jmsy.2018.02.002
56. M. Helu, A. Joseph, and T. Hedberg Jr., "A Standards-Based Approach for Linking As-Planned to As-Fabricated Product Data," *CIRP Annals* 67, no. 1 (2018): 487–490, https://doi.org/10.1016/j.cirp.2018.04.039
57. M. Hardwick, "Operate, Orchestrate, and Originate" (paper presentation, Model-Based Enterprise Summit 2016, Gaithersburg, MD, April 12–14, 2016).
58. J. Michaloski, B. Lee, F. Proctor, S. Venkatesh, and S. Ly, "Quantifying the Performance of MT-Connect in a Distributed Manufacturing Environment," in *29th Computers and Information in Engineering Conference* (New York: American Society of Mechanical Engineers, 2009), 1–7, https://doi.org/10.1115/DETC2009-86666
59. M. Helu and B. Weiss, "The Current State of Sensing, Health Management, and Control for Small-to-Medium-Sized Manufacturers," in *11th International Manufacturing Science and Engineering Conference* (New York: American Society of Mechanical Engineers, 2016), MSEC2016-8783, https://doi.org/10.1115/MSEC2016-8783
60. *OPC Unified Architecture—Part 1: Overview and Concepts*, OPC 10000-1 – Part 1 (Scottsdale, AZ: OPC Foundation, 2017).
61. M. Schleipen, "OPC UA Supporting the Automated Engineering of Production Monitoring and Control Systems," in *International Conference on Emerging Technologies and Factory Automation* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2008), 640–647, https://doi.org/10.1109/ETFA.2008.4638464
62. T. Hannelius, M. Salmenperä, and S. Kuikka, "Roadmap to Adopting OPC UA," in *Sixth International Conference on Industrial Informatics* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2008), 756–761, https://doi.org/10.1109/INDIN.2008.4618203
63. S.-H. Leitner and W. Mahnke, "OPC UA – Service-Oriented Architecture for Industrial Applications," *Softwaretechnik – Trends* 26, no. 4 (2006).
64. A. Balador, N. Ericsson, and Z. Bakhshi, "Communication Middleware Technologies for Industrial Distributed Control Systems: A Literature Review," in *22nd International Conference on Emerging Technologies and Factory Automation* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2017), 1–6, https://doi.org/10.1109/ETFA.2017.8247730

65. D. Treffer, P. Wahl, D. Markl, G. Koscher, E. Roblegg, and J. G. Khinast, "Hot Melt Extrusion as a Continuous Pharmaceutical Manufacturing Process," in *Melt Extrusion: Materials, Technology and Drug Product Design* (New York: Springer, 2013), 363–396, https://doi.org/10.1007/978-14614-8432-5
66. T. Terzimehic, M. Wenger, A. Zoitl, A. Bayha, K. Becker, T. Müller, and H. Schauerte, "Towards an Industry 4.0 Compliant Control Software Architecture Using IEC 61499 & OPC UA," in *22nd International Conference on Emerging Technologies and Factory Automation* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2017), 1–4, https://doi.org/10.1109/ETFA.2017.8247718
67. S. Rohjans, M. Uslar, and H. J. Appelrath, "OPC UA and CIM: Semantics for the Smart Grid," in *2010 PES Transmission and Distribution Conference and Exposition* (Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2010), 1–8, https://doi.org/10.1109/TDC.2010.5484299
68. M. Hoffmann, T. Meisen, and S. Jeschke, "OPC UA Based ERP Agents: Enabling Scalable Communication Solutions in Heterogeneous Automation Environments," in *International Conference on Practical Applications of Agents and Multi-Agent Systems* (Cham, Switzerland: Springer, 2017), 120–131, https://doi.org/10.1007/978-3-319-59930-4_10
69. M. Hoffmann, C. Büscher, T. Meisen, and S. Jeschke, "Continuous Integration of Field Level Production Data into Top-Level Information Systems Using the OPC Interface Standard," *Procedia CIRP* 41 (2016): 496–501, https://doi.org/10.1016/j.procir.2015.12.059
70. D. Mourtzis, N. Milas, and A. Vlachou, "An Internet of Things-Based Monitoring System for Shop-Floor Control," *Journal of Computing and Information Science in Engineering* 18, no. 2 (March 2018): 021005, https://doi.org/10.1115/1.4039429
71. F. Tao, J. Cheng, and Q. Qi, "IIHub: An Industrial Internet-of-Things Hub toward Smart Manufacturing Based on Cyber-Physical System," *IEEE Transactions on Industrial Informatics* 14, no. 5 (May 2018): 2271–2280, https://doi.org/10.1109/TII.2017.2759178
72. A. Girbea, C. Suciu, S. Nechifor, and F. Sisak, "Design and Implementation of a Service-Oriented Architecture for the Optimization of Industrial Applications," *IEEE Transactions on Industrial Informatics* 10, no. 1 (February 2014): 185–196, https://doi.org/10.1109/TII.2013.2253112
73. D. Wu, S. Liu, L. Zhang, J. Terpenny, R. X. Gao, T. Kurfess, and J. A. Guzzo, "A Fog Computing-Based Framework for Process Monitoring and Prognosis in Cyber-Manufacturing," *Journal of Manufacturing Systems* 43 (April 2017): 25–34, https://doi.org/10.1016/j.jmsy.2017.02.011
74. L. E. S. Oliveira and A. J. Álvares, "Axiomatic Design Applied to the Development of a System for Monitoring and Teleoperation of a CNC Machine through the Internet," *Procedia CIRP* 53 (2016): 198–205, https://doi.org/10.1016/j.procir.2016.06.099
75. N. M. Torrisi and J. F. G. de Oliveira, "Remote Monitoring for High-Speed CNC Processes over Public IP Networks Using CyberOPC," *International Journal of Advanced Manufacturing Technology* 60, no. 1 (April 2012): 191–200, https://doi.org/10.1007/s00170-011-3580-3
76. T. Hedberg Jr., A. B. Feeney, M. Helu, and J. A. Camelio, "Toward a Lifecycle Information Framework and Technology in Manufacturing," *Journal of Computing and Information Science in Engineering* 17, no. 2 (June 2017): 021010, https://doi.org/10.1115/1.4034132
77. P. Lei, L. Zheng, W. Xiao, C. Li, and D. Wang, "A Closed-Loop Machining System for Assembly Interfaces of Large-Scale Component Based on Extended STEP-NC," *International Journal of Advanced Manufacturing Technology* 91, no. 5 (July 2017): 2499–2525, https://doi.org/10.1007/s00170-016-9904-6
78. J. S. Albus and R. Lumia, "The Enhanced Machine Controller (EMC): An Open Architecture Controller for Machine Tools," *Journal of Manufacturing Review* 7, no. 3 (1994): 278–280.
79. T. R. Kramer, F. M. Proctor, and E. Messina, *The NIST RS274NGC Interpreter – Version 3, NISTIR 6556* (Gaithersburg, MD: National Institute of Standards and Technology, 2000).
80. X. Beudaert, S. Lavernhe, and C. Tournier, "Feedrate Optimization in 5-Axis Machining Based on Direct Trajectory Interpolation on the Surface Using an Open CNC," in *11th International Conference on High Speed Machining* (Lyon, France: Hyper Articles en Ligne, 2014), 14042.
81. K. Oldknow and I. Yellowley, "Design, Implementation and Validation of a System for the Dynamic Reconfiguration of Open Architecture Machine Tool Controls," *International Journal of Machine Tools and Manufacture* 41, no. 6 (May 2001): 795–808, https://doi.org/10.1016/S0890-6955(00)00109-7
82. E. Uhlmann, A. J. Abackerli, K. Schützer, H. A. Lepikson, A. L. Helleno, M. C. O. Papa, E. G. del Conte, and J. Mewis, "Simulation and Analysis of Error Impact on Freeform Surface Milling," *International Journal of Advanced Manufacturing Technology* 70, no. 1 (January 2014): 607–620, https://doi.org/10.1007/s00170-013-5280-7
83. J. G. Campos and L. R. Miguez, "Standard Process Monitoring and Traceability Programming in Collaborative CAD/CAM/CNC Manufacturing Scenarios," *Computers in Industry* 62, no. 3 (April 2011): 311–322, https://doi.org/10.1016/j.compind.2010.09.003
84. S.-J. Shin, J. Woo, D. B. Kim, S. Kumaraguru, and S. Rachuri, "Developing a Virtual Machining Model to Generate MTConnect Machine-Monitoring Data from STEP-NC," *International Journal of Production Research* 54, no. 15 (August 2016): 4487–4505, https://doi.org/10.1080/00207543.2015.1064182
85. M. Wosnik, C. Kramer, A. Selig, and P. Klemm, "Enabling Feedback of Process Data by Use of STEP-NC," *International Journal of Computer Integrated Manufacturing* 19, no. 6 (2006): 559–569, https://doi.org/10.1080/09511920600622080

86. G. M. Martinov, S. V. Sokolov, L. I. Martinova, A. S. Grigoryev, and P. A. Nikishechkin, "Approach to the Diagnosis and Configuration of Servo Drives in Heterogeneous Machine Control Systems," in *International Conference on Swarm Intelligence* (Cham, Switzerland: Springer, 2017), 586–594, https://doi.org/10.1007/978-3-319-61833-3_62
87. P. Li, T. Hu, and C. Zhang, "A Unified Communication Framework for Intelligent Integrated CNC on the Shop Floor," *Procedia Engineering* 15 (2011): 840–847, https://doi.org/10.1016/j.proeng.2011.08.156
88. W. Shen, S. Y. T. Lang, and L. Wang, "iShopFloor: An Internet-Enabled Agent-Based Intelligent Shop Floor," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* 35, no. 3 (August 2005): 371–381, https://doi.org/10.1109/TSMCC.2004.843224
89. Y. Yang, C. Yin, X.-B. Li, and L. Li, "Multi-source Information Intelligent Collection and Monitoring of CNC Machine Tools Based on Multi-agent," in *Monterey Workshop 2016: Challenges and Opportunity with Big Data* (Cham, Switzerland: Springer, 2017), 111–121, https://doi.org/10.1007/978-3-319-61994-1_11
90. A. J. Álvares, J. C. E. Ferreira, and R. M. Lorenzo, "An Integrated Web-Based CAD/CAPP/CAM System for the Remote Design and Manufacture of Feature-Based Cylindrical Parts," *Journal of Intelligent Manufacturing* 19, no. 6 (June 2008): 643, https://doi.org/10.1007/s10845-008-0117-1
91. R. Fukuda, R. Kiyooka, and H. Aoyama, "Development of Automatic System for Process Planning and NC Program Generation on Turning-Milling Machine Tool with Three Turrets," in *International Mechanical Engineering Congress and Exposition* (New York: American Society of Mechanical Engineers, 2017), V011T15A002, https://doi.org/10.1115/IMECE2017-70491
92. A. Cherubini, R. Passama, A. Crosnier, A. Lasnier, and P. Fraisse, "Collaborative Manufacturing with Physical Human–Robot Interaction," *Robotics and Computer-Integrated Manufacturing* 40 (August 2016): 1–13, https://doi.org/10.1016/j.rcim.2015.12.007
93. A. Djuric, R. Urbanic, and J. Rickli, "A Framework for Collaborative Robot (CoBot) Integration in Advanced Manufacturing Systems," *SAE International Journal of Materials and Manufacturing* 9, no. 2 (2016): 457–464, https://doi.org/10.4271/2016-01-0337
94. D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, "Cloud-Based Design and Manufacturing: A New Paradigm in Digital Manufacturing and Design Innovation," *Computer-Aided Design* 59 (February 2015): 1–14, https://doi.org/10.1016/j.cad.2014.07.006
95. X. W. Xu and Q. He, "Striving for a Total Integration of CAD, CAPP, CAM and CNC," *Robotics and Computer-Integrated Manufacturing* 20, no. 2 (April 2004): 101–109, https://doi.org/10.1016/j.rcim.2003.08.003
96. J. Michaloski, T. Kramer, F. Proctor, X. Xu, S. Venkatesh, and D. Odendahl, *STEPNC++ – An Effective Tool for Feature-Based CAM/CNC* (Surrey, UK: Springer-Verlag, 2009).
97. Y. Lu, C. Liu, K. I.-K. Wang, H. Huang, and X. Xu, "Digital Twin-Driven Smart Manufacturing: Connotation, Reference Model, Applications and Research Issues," *Robotics and Computer-Integrated Manufacturing* 61 (February 2020): 101837, https://doi.org/10.1016/j.rcim.2019.101837
98. D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer, "Cloud Manufacturing: Strategic Vision and State-of-the-Art," *Journal of Manufacturing Systems* 32, no. 4 (October 2013): 564–579, https://doi.org/10.1016/j.jmsy.2013.04.008
99. G. W. Vogl, B. A. Weiss, and M. Helu, "A Review of Diagnostic and Prognostic Capabilities and Best Practices for Manufacturing," *Journal of Intelligent Manufacturing* 30, no. 1 (January 2019): 79–95, https://doi.org/10.1007/s10845-016-1228-8
100. M. A. Othman, M. Minhat, and Z. Jamaludin, "An Overview on STEP-NC Compliant Controller Development," *IOP Conference Series: Materials Science and Engineering* 257 (November 2017): 012048, https://doi.org/10.1088/1757-899X/257/1/012048
101. D. R. Lynn, "Direct Servo Control of Positional Derivatives for 5-Axis CNC Machine Tools Using Densely-Sampled Toolpaths" (PhD thesis, Georgia Institute of Technology, 2019).
102. T. Hedberg Jr., M. Helu, and T. Sprock, "A Standards and Technology Roadmap for Scalable Distributed Manufacturing Systems," in *13th International Manufacturing Science and Engineering Conference* (New York: American Society of Mechanical Engineers, 2018), MSEC2018-6550, https://doi.org/10.1115/MSEC2018-6550