



A review of G code, STEP, STEP-NC, and open architecture control technologies based embedded CNC systems

Kamran Latif¹ · Anbia Adam² · Yusri Yusof² · Aini Zuhra Abdul Kadir³

Received: 11 August 2020 / Accepted: 1 February 2021 / Published online: 17 April 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd. part of Springer Nature 2021

Abstract

Computer Numeric Control (CNC) is a significant pillar of manufacturing for decades. These systems are an essential part of future manufacturing. In the urge towards the development of next-generation manufacturing systems, various technologies have been introduced. Moreover, based on those technologies, numerous systems have been developed all around the world. In order to assist in future research, the related major technologies and developments are discussed in this content. This paper presents the review of basics CN), Computer-Aided Design (CAD), Computer-Aided Process Planning (CAPP), Computer-Aided Manufacturing (CAM), G codes (ISO 6983), Standard for the Exchange of Product (STEP) model data (STEP), STEP-numeric control (STEP-NC), and open architecture control (OAC) technologies. The in-depth discussion on the development and implementation of the STEP and STEP-NC technologies is also addressed. The review also highlights the benefits and role of the open architecture control technology in the CNC system development. Apart from that, the paper also presents the last 17 years of embedded CNC systems development in tabular form, which shows the cocktail of various technologies and ISO data interface models. Finally, the paper concludes with future directions and recommendations.

Keywords STEP · STEP-NC · CNC · ISO 10303 · ISO 14649 · G Code · ISO 6983 · Open Architecture Control (OAC)

1 Development of STEP (ISO 10303)

CNC system uses minicomputers or microcomputers to generate, parse, and execute the sequential control that

✉ Kamran Latif
kamran@utem.edu.my

Anbia Adam
anbiaadam@gmail.com

Yusri Yusof
yusri@uthm.edu.my

Aini Zuhra Abdul Kadir
aini@mail.fkm.utm.my

¹ Faculty of Mechanical and Manufacturing Engineering Technology, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia

² Faculty of Mechanical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia (UTHM), Parit Raja, Malaysia

³ School of Mechanical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia (UTM), Johor Bahru, Malaysia

describes the end effector behavior. The first-ever CNC machine was developed in the 1970s, where the electronic hardware and punch card of the previous NC systems were replaced by computers [1]. In further evolution towards modern systems, the need to produce a wide range of parts arose during the 1970s and 1980s, which created the requirement of a flexible manufacturing system (FMS). In order to achieve a flexible environment for manufacturing systems, the CNC machines play a critical role because of their ability to be reprogrammed for the manufacturing of different and complex parts in bulk quantities [2]. The development of these types of parts requires sophisticated programs. Therefore, CAD and CAM systems were used to generate CNC part programs [3]. The output of the CAM system is in .NC file that translates the job information from drawing to computer-controlled machine unit. That specific language was initially known as the Automatically Programmed Tool (APT), and currently, it has been recognized as ISO 6983, RS-274D, or G M codes. In this standard, the operator tells the computerized machine unit “how to make,” which defines the instructions of where, how, and what path to move. However, in the development of a flexible manufacturing environment, the programming tasks became more complex and challenging [3]. In order

to fulfill these requirements, the ISO 6983 data interface model was found to have limited capabilities. There are a number of problems that were found in ISO 6983 such as:

- The ISO 6983 language is focused on programming the path of the cutter center location with respect to the machine axis, rather than the machining tasks with respect to the part [4–6].
- The standard defines the syntax of program statements, but in most cases leaves the semantics unclear, together with limited low-level control over program execution. These programs become machine-dependent when processed in a CAM system by machine-specific post-processor [5–7].
- The vendor usually enhances the language with further extension commands to provide new features, while these extensions are not covered by ISO 6983. Hence, it becomes a machine-specific language, and programs are not exchangeable between other machine tools [5, 6, 8].
- The flow of information from design to manufacturing is uni-directional. There is no feedback on data in ISO 6983. Therefore, the last minute changes and modifications of machining problems on the shop floor are hardly possible [5, 9].
- The CAD data is not directly used on the machine tool. It is processed by means of a machine-specific post-processor in terms of low-level data. This incomplete dataset makes verification and simulations very difficult [5].
- This standard does not support today's demand in the area of five-axis milling or high-speed machining because it is incapable of processing Spline data [5, 9].

From these limitations, it is clear that there are two significant issues: interoperability and adaptability of CNC machines that need to be addressed. However, in reality, these GM code programs are still precious because they integrate the micro-process plan with operator experience [10]. But in order to meet the challenges of future manufacturing, the new data interface model is required. The initial challenge towards the development of a new data model was the enabling of seamless geometrical data flow between CAD and CAM systems. During the 1980s, different data formats were proposed, but none of them was able to satisfy the needs of developers and users [11]. Then, in the mid-1980s, the international community decided to develop a better standard for geometrical data exchange between CAD and CAM systems. The result was the ISO 10303 standard, commonly known as STEP.

ISO 10303 was developed to cover all the aspects of CAD/CAM data exchange with the implementation and acceptance by industry. The objective of STEP is “to provide a means of describing the product data throughout the

life cycle of a product, which is independent of any particular computer system” [12]. The evolution towards STEP was started in 1979 with the development of Initial Graphics Exchange Specification (IGES). IGES was the first standard format for the CAD information exchange. The major drawback of IGES was the incapability of exchanging data among free form surfaces [2]. Later, VDA, a German company, developed Verband der Automobilindustrie Flächenschnittstelle (VDAFS) to focus on free-form surface information translation [13]. During 1984, the initial development of ISO 10303 was started to overcome the drawbacks of IGES and VDA-FS. In 1988, the first major release of STEP was published, in which a large set of models had been assembled into a single model called Integrated Product Information Model (IPIM) [14]. By the following year, STEP was diverted to use the Application Protocol (AP) as a subset. The architecture of APs was developed in the following few years. Then finally, in 1994, the first version of STEP was adopted as an ISO standard. Soon after release, the established companies like General Motors, and Boeing also committed to using STEP. During 1994/1995, ISO published the initial release of STEP as an international standard. In that stage, STEP Parts 1, 11, 21, 31, 42, 43, 44, 46, 101, AP201, and AP 203 were introduced. The next significant development in STEP occurred during the year 2002. Where the capabilities of STEP were expanded in different industries (automotive, electronic manufacturing, aerospace and electrical, etc.) with the introduction of AP 202, AP 209, AP 210, AP 212, AP 214, AP 224, AP 225, AP 227, and AP 232, after that, the next development in the STEP was the introduction of STEP modular architecture. Modular architecture solved the problems emerging from large data structures. Later, a new AP 242 was introduced for geometric dimensions and tolerance in combination with AP 203 and AP 214 [2]. The implementation of STEP has been carried out by three methods: ISO 10303-21, STEP Data Access Interface (SDAI), and ISO 10303-28. Among all these methods, the most popular are STEP Part 21 for offline manufacturing and STEP Part 28 for online manufacturing or e-manufacturing. STEP significantly improves the interoperability between CAD systems, but in the meantime, it also creates the requirement of the new standard to exchange the information between CNC machines as well as CAM systems. Accordingly, in 1999, an international project was started to bring the benefits of STEP to CAM and CNC, known as STEP-NC (ISO 14649) [4].

2 Establishment of STEP-NC (ISO 14649)

ISO 10303 standard resolves the problems relating to the product data exchange between CAD, CAPP, and CAM systems. Therefore, for establishing a seamless data flow

between CAM and CNC, a new standard, ISO 14649, was introduced commonly known as STEP-Compliant Numerical Control or STEP-NC in short. This standard offers the possibility of seamless data integration of applications throughout the design to the manufacturing cycle [15]. The attention of ISO is on the development of the STEP manufacturing environment, which includes STEP in, STEP out, and STEP throughout. The aim of STEP-NC is to provide remedies for the shortcomings of ISO 6983 by specifying machining processes rather than tool motion. It is done by using the object and feature-oriented concept of working steps, which provides a seamless link in computer-aided systems (CAX) to make CNC more open, interoperable, portable, adaptable, flexible, and intelligent. The significant benefit of STEP-NC is that it uses existing data models of ISO 10303 for enabling smooth and seamless information exchange in CAX [16]. ISO 14649 contains a high degree of information sets, which includes “What-to-make” (geometry) and “How-to-make” (process plan) [10].

2.1 Versions of STEP-NC

At the moment, there are two versions of STEP-NC that are under development by two different Sub-Committees (SC) of Technical Committee (TC) 184 under ISO: ISO 14649 and ISO 10303-238. The ISO 14649 is being developed by SC1 of TC 184 under ISO, whose preliminary focus is on machine control. The models of this version were written in EXPRESS language and are of Application Reference Model (ARM) type [17]. The second version (ISO 10303-238) is basically an upgradation of ISO 14649 standard. The development of this version is carried out under SC4 of TC 184, whose focus is on industrial data [17]. This is an Application Interpreted Model (AIM) because, it combines all the parts of ISO 14649 (except Part 1) into a single model with some modifications and also mapped them into STEP integrated resources to obtain an implementation AIM model [18]. The main aim of ISO 13030-238 is to enable total integration between CAD, CAM and CNC, or in other words, an integration of STEP and STEP-NC [12]. The main difference of ISO 10303-238 in comparison with ISO 14649 is that it includes 3D model and Geometric and Dimensioning and Tolerancing (GD&T) models [17]. This version is composed of three main parts: 3D representation, working plan for manufacturing, and technological parameters for manufacturing [12]. This version has overcome the issues of ISO 14649, namely, contain full CAD data including 3D, employed full tradition of EXPRESS and planned to integrate ISO 14649 and STEP.

2.2 Benefits of STEP-NC

STEP-NC is an object and feature-oriented data model, which provides new opportunities to support high-level and standardized information from design to manufacturing [19]. The development and implementation of STEP-NC will benefit the manufacturing world in various aspects. Some of the significant benefits are discussed in this section, which is a summary of the benefits highlighted by [2, 5–7, 10, 12, 17, 19, 20]

- In current CAX, the information can be lost in a pipeline because of the incomplete data transfer from CAD to CAM. In this situation, the fixes to the geometry are made in CAM and are not communicated back to the CAD, which causes limited data flow to the post. Another reason for information loss is the use of different standards, whereas STEP-NC addressed these issues by providing a complete and structured data model linked with geometrical and technical information. Also, it provides a single standard for data integration between different systems. Hence, no information is lost in design to the manufacturing cycle.
- In current systems, the information flow is uni-directional. There is no feedback on information between CAX. The last-minute changes are almost impossible due to the uni-directional data flow in the current CAX, whereas STEP-NC standard feature description and model structure are synchronized with ISO 10303; therefore, they support bi-directional information flow in CAX. Due to this, the shop floor modification in machining and technological data is possible.
- Current CNC machines are working on ISO 6983 data interface model, which contains low-level information based on “how-to-make” instructions, whereas STEP-NC standard contains high-level information based on “what-to-make” and “how-to-make” instructions. This high-level information can enable optimization, simulation, inspection, and other facilities in the CNC controller. Therefore, we can say that the data elements of STEP-NC are adequate enough to describe task-oriented NC data.
- Today’s systems use post-processor for the generation of ISO 6983 standard, which makes ISO 6983 more machine-specific and vendor-dependent. Hence, generated code cannot be used on different systems without modifications. Alternatively, the STEP-NC standard is composed of a neutral format, which is free from vendor dependency and machine specifications. Hence, it eliminates the need for post-processor and

also saves time by reducing CAx chain. STEP-NC file is not machine-specific and can be operated on any machine that meets tooling requirements.

- The data model of STEP-NC is extendable to further technologies and is also scalable with Conformance Classes (CC) to match the ability of specific CAM Shop Floor Programming (SFP) in NC systems.
- The STEP-NC provides enough information for the enabling of intelligent optimization facility that helps in reducing machining time for small- to medium-sized jobs.
- STEP-NC file contains all the information required to produce a job. Therefore, manufacturing operations may be adjusted to maximize production efficiency.
- STEP-NC provides complete safety and adaptability of machine tools because it is vendor-independent and defines safety areas for fixtures as part of the setup.
- There will be a significant reduction in the number of drawings sent from CAD to CAM, as STEP-NC data is self-documented.
- STEP-NC enables e-manufacturing or Web-based manufacturing with the help of XML (ISO 14649-28) information transfer.

2.3 Structure of STEP-NC

The STEP-NC file contains high-level information in object-oriented view of programming in terms of manufacturing features. The structure of the STEP-NC data model is divided into two sections: HEADER and DATA. The HEADER section contains general information such as file name, author, date, and organization. This section starts with the particular token HEADER and ends with the ENDSEC token, whereas the other, DATA, section is considered to be an essential section of the STEP-NC. This section is the main section of the STEP-NC data model, which begins with the DATA token and ends with the ENDSEC token. This section contains all the information regarding manufacturing tasks and geometries. The contents of this section are further divided into three parts: workplan or executables, technology description, and geometry description. Each STEP-NC program contains a PROJECT entity in the DATA section that indicates the starting point of the NC functions. This entity indicates the workplan to be executed and the workpiece upon which operations have to be performed [21]. The workplan describes the series of manufacturing tasks or commands in a pre-established order. This part may also include workpiece data information. The executables can be of three types: workingsteps, NC functions, and program structures. The workingsteps are the necessary executables of the workplan that define manufacturing features in two-and-a-half-dimensional (2.5D) and 3D regions. Each workingstep also contains sub-features (pocket, slot,

round hole, etc.) with cutting condition environment [8] [22]. The technology description part of the DATA section contains a detailed and complete description of all the workingsteps data such as a tool, machining strategy, definitions of the workpiece, depth of a hole, feed rate, spindle speed, and tool diameter [21]. However, the geometry description part includes all the geometry data used in the components, and it is described in the ISO 10303 format [8].

3 Implementation of STEP-NC (ISO 14649)

The STEP implementation on CNC was carried out from over the years in various research projects. Among them, only a few major projects that were funded by industries, companies, and institutes such as Intelligent Manufacturing System (IMS) STEP-NC [23], European Strategic Program on Research in Information Technology (ESPRIT) STEP-NC [24], Super Model, STEP Manufacturing Suite (SMS), Rapid Acquisition of Manufactured Parts (RAMP), and Intelligent Manufacture for STEP-NC Compliant Machining and Inspection [11]. The outcomes of these projects make STEP-NC an active industrial standard by providing new opportunities to CNC system controllers [17]. The implementation of STEP-NC was initiated on current commercially available CNC controller by translating STEP-NC data into G codes. This implementation failed to provide all the features of STEP-NC in CNC because the translation was in low-level language and another major problem was of the CNCs' machine vendor dependence [7]. The problem of vendors' specification dependence makes CNCs less adaptable. Various efforts had been made to address this problem, in particular, the trend towards Open Architecture Control (OAC) was initiated in 1980s [25].

3.1 Open Architecture Control (OAC)

The aim of OAC technology is to develop a neutral vendor control system interface and provide interoperability, interaction, portability, and scalability specifications to the system [26]. The ability to use the same component for working with different systems is enabled by interoperability. Interaction ensures the communication of standard data semantics. The natural transformation of application software from one environment to another is enabled by portability. The incremental and detrimental abilities of the system as per customer demands are described by scalability specifications [27, 28]. These specifications can also be used for the estimation of the openness of the system. The design philosophy of OAC gained wide attention all around the world. The USA was the first nation that rose the concept of OAC by introducing first OAC system MOSAIC in the year 1998. Since last two decades, OAC had made

great progress and became an important field of research to achieve the goal of modern manufacturing. There are many research works that had been carried out on OAC in the shape of significant projects such as Open System Architecture for Controls within Automation Systems (OSACA), Open System Environment for Controllers (OSEC), Open Modular Architecture Controller (OMAC), Japanese Open Promotion Group (JOP), Global HMI, Organic Reconfigurable Operating System (ORCOS), and Openness, Conscientiousness, Extroversion, Agreeableness, and Neuroticism (OCEAN) [29]. These systems had enhanced the openness in the control environment, capabilities, advantages, and benefits of the OAC by allowing the changes in both hardware and software configurations, provides a plug and play compatibility in real-time, able to integrate, and share data among the third-party developed hardware and software components, able to cooperate with other systems at a systematic level and others. These facilities of OAC make CNC more flexible by allowing the system to be adjusted easily. The OAC system has more advantages over the current commercial closed systems because its primary focus is on reusable software technology, unlike hardware, as in the current system of closed architecture. The adaptation of OAC technology enables more intelligence, monitoring, inspection, openness, flexibility, adaptability, and portability facilities to CNC controllers [30].

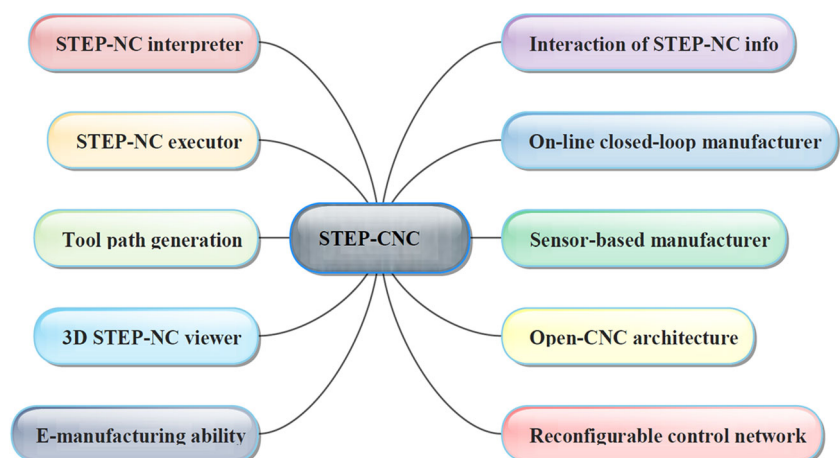
The evolution of STEP-NC in CNC controllers is categorized into three steps in terms of STEP-NC programming: Indirect, Interpreted, and Adaptive [20].

3.2 Indirect STEP-NC Programming Implementation Approach

The main focus of this approach was on the development of “interpreters/translators” that translates STEP-NC (AP 203 and AP 224) information to commercial G code controllers [31]. The first two stages of the Super Model and European STEP-NC project ESPRIT falls into this category of work. In the first stage of the Super Model, a range of software tools were developed (i-e; ST-Plan, ST-Machine, STIX [11]) with the involvement of Gibbs CAM and various third-party software tools. In the second stage, Gibbs CAM STEP-NC adapter plugins were developed to read the AP 238 file [20]. All of these software tools were developed by a small private company (STEP Tools Inc) specialized in the field of STEP-NC development and support. Apart from those, STEP Tools Inc also introduced some other software solutions for testing and using the STEP-NC data model in real-life applications. The STEP-NC Explorer is a tool used for opening and reading of STEP-NC files. This tool also converts the NC file into the STEP-NC file. The ST-Developer tool contains Dynamic Link Library (DLL) for reading, writing, and manipulation of STEP-NC files. It contains tools for AP 203, AP 209, and AP 214 file readings. It also shows the graphical view of the STEP-NC file in ST-Viewer [12].

Within the In-Direct STEP-NC programming approach, one of the first STEP-based prototype was developed within the contents of the ESPRIT project, which utilized CATIA

Fig. 1 Criteria of STEP-NC system



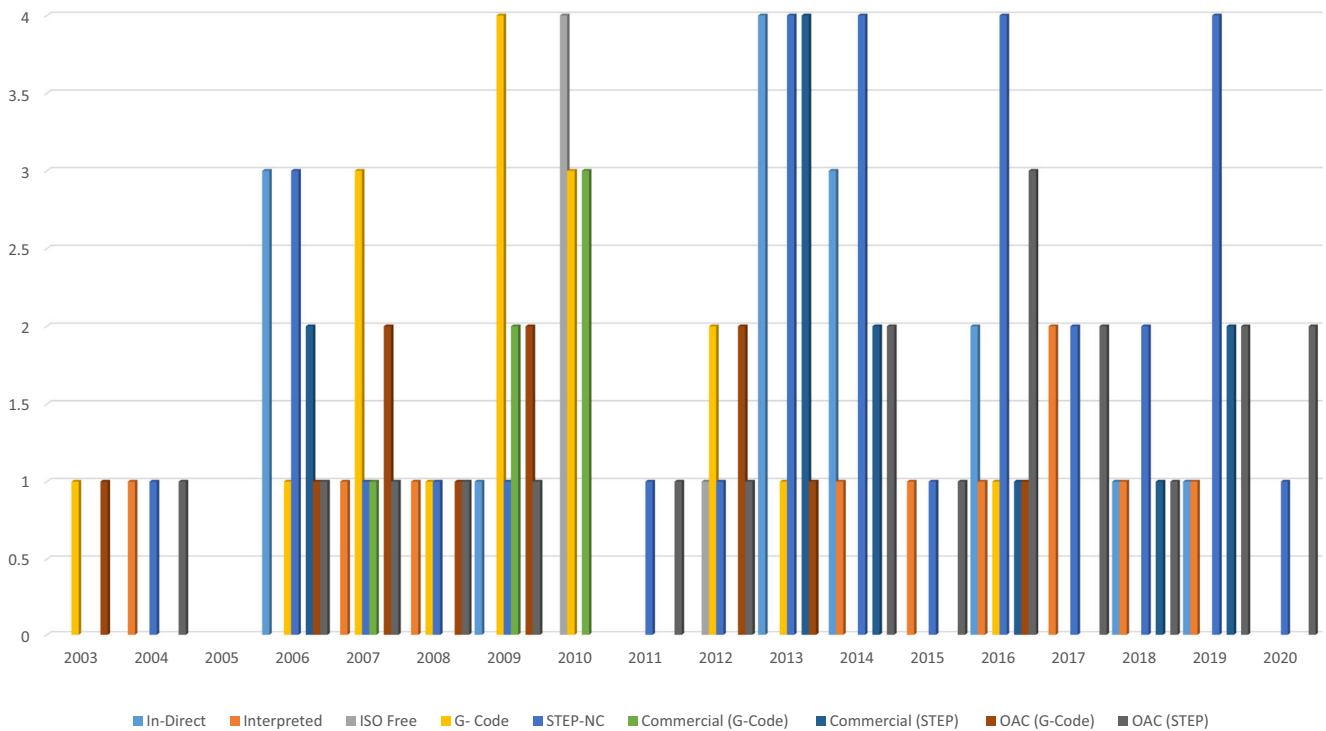


Fig. 2 Integrated CNC technologies development

and Open Mind CAD software for STEP-NC file generation and modified Siemens 840D controller for interpretation [32]. Similarly, an Agent-Based Computer-Aided Manufacturing (AB-CAM) system was developed at the AMST Centre in the Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, UK. The system generates the ISO 14649 part program and translates

it into G code for operation [33]. The system is composed of multi-agents where each agent is charged with the responsibility. It defines both simple and combined manufacturing features by Manufacturing Feature Agents (MFAs) that can acquire corresponding features information from the component and manufacturing resource models. After that, the process planning tasks are carried out, and the

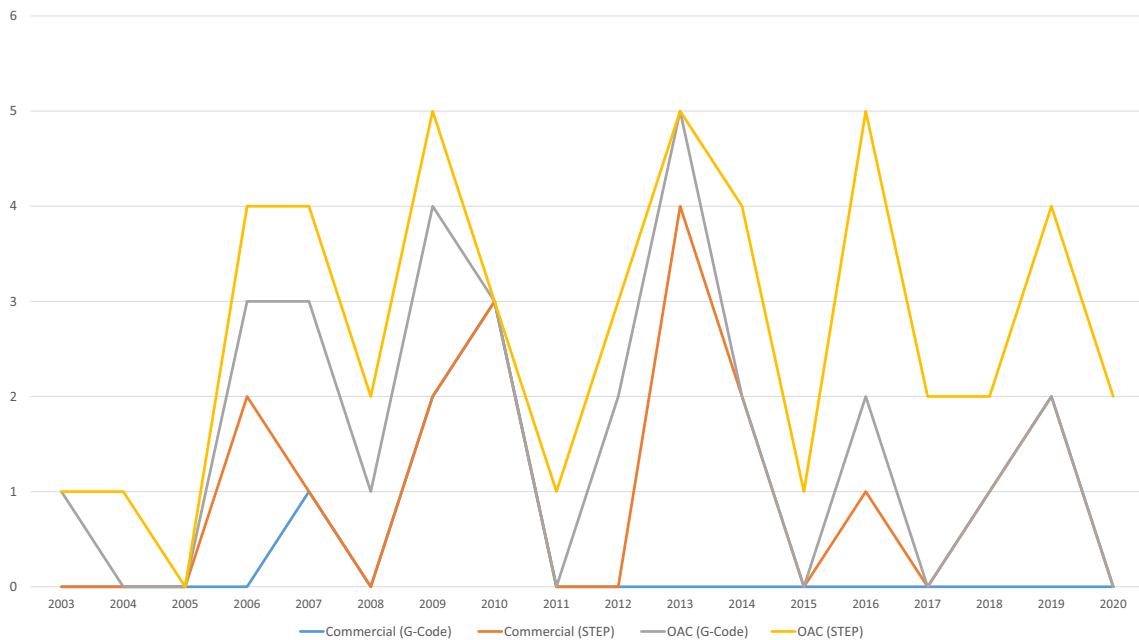


Fig. 3 Trend of OAC-based CNC systems

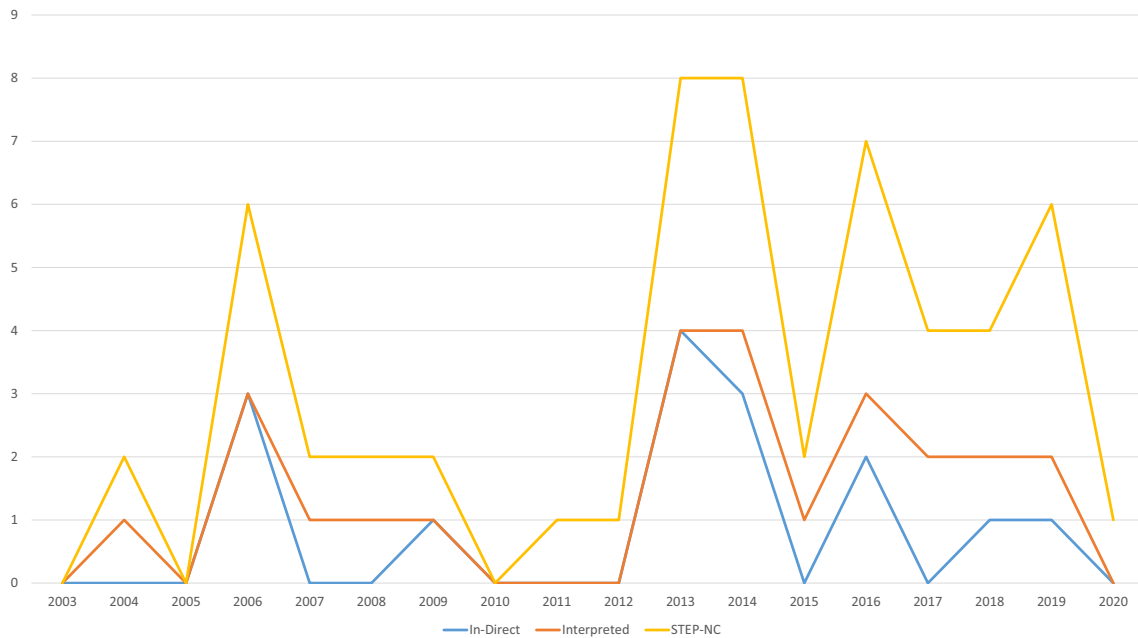


Fig. 4 Trend of STEP-NC and its implementation approaches

operation-related information is written into a corresponding workingsteps. From these workingsteps data, the complete workplan is then generated by optimizer agents via rearrangement of workingsteps. The demonstration platform of the system was developed in JAVA, known as the Integrated Platform for Process Planning and Control (IP3AC) [34]. A similar approach was introduced by University of Auckland that interprets STEP-NC part code and translates it into G code for operations. In that work, firstly, an interpreter was developed that translates ISO 14649 instructions [35] and secondly, an open nature prototype system was developed [36] for performing the tasks of the developed STEP-NC interpreter. Generally, the In-Direct STEP-NC programming approach had focused on the popular CNCs and open architecture controllers. Within this approach, the vision of intelligent CNC could not be achieved because of the translation of data from a high-level to low-level information. In order to perform high-level activities, a new controller is needed to be developed that is able to directly interpret STEP-NC data.

3.3 Interpreted STEP-NC Programming Implementation Approach

The aim of this approach was to execute axis commands directly from STEP-NC data interface model [31]. Based on this approach, the first prototype was developed at the National Research Laboratory for STEP-NC Technology (NRL-SNT) at PosTECH University Korea [4]. The system is composed of STEP-NC file generator, Tool Path Generator (PosTPG), Tool Path Viewer (PosTPV), Shop

Floor Programming System (PosSFP), Man-Machine Interface (PosMMI), CNC kernel (PosCNC), and machine tool driving and control modules. This system had not been implemented on industrial machine tool but was validated by a prototype system [20]. Within this category, a G code free machining scenario was presented by Manufacturing Systems Laboratory, University of Auckland, for lathe machine [37]. The work consisted of two parts; the first part was about the retrofitting of an existing CNC machine, whereas the second part was about the development of STEP-compliant NC (STEPcNC) converter. The retrofitting was done by using CompuCam's motion control system that has its own programming language (6K Motion Control), and it can interface with CAPP/CAM programs via VB, VC++, and Delphi platforms. The STEPcNC reads, understands, and translates the data to the CNC controller through HMI. Later, this work was extended by [38] to enable bi-directional data flow for STEP-NC data, machine-specific language, and low-level machine control commands. Another piece of research work was carried out in Germany. The first STEP-NC industrial prototype based on the Siemens 840D controller was developed by the Laboratory for Machine Tools and Production Engineering (WZL) at RWTH Aachen University. The prototype parses the STEP-NC physical file directly via Graphical User Interface (GUI) developed by using Shop Mill, and HMI developed by using Sinumerik 840D [39]. The ISW in Stuttgart, Germany, is also active in the STEP-NC field. ISW introduced a prototype system, STEPTurn, which adopted STEP and STEP-NC standards for turning operations. The main advantage of STEPTurn software application was the auto-

matic planning of the manufacturing steps sequence that was achieved by using two types of priority pairs. The software reads geometry data from a STEP AP-203 Part 21 file, performs process planning tasks, and generates a STEP-NC physical file [40] [41]. ISW also developed a system for parsing the feedback from CNC to CAM using ISO 14649. The system performed feedback operations by generating modified STEP-NC file via sorting cutting forces during the NC process into a database by investigating the stored data, compensation algorithms, and model for tool deflection [42]. A number of vendors and industries from Switzerland have also taken part in the STEP-NC research. Most of the work is focused on the STEP-based Wire Electric Discharge Machine (EDM) and its controller. In that work, a SolidWorks software was used for part modelling test. Alpha CAM was used for the development of the STEP-NC software module. And a prototype using the Agie-Charmilles controller was developed for experimental studies [43] [44]. The IRCCyN Laboratory of France is also working on STEP-NC implementation. They have developed a system known as STEP-NC Platform for Advance and Intelligent Manufacturing (SPAIM) based on the Interpreted category of STEP-NC programming approach. This platform is composed of HMI, master, execution, CAD reconstruction, tool path generation, and simulation modules. These modules are controlled by Delphi applications on the CNC controller and external PC. The tool path module of SPAIM is based on vendor components [20] [31]. Another system of similar capabilities comprising Intelligent and Interoperable Manufacturing Platform (IIMP) was developed by University of Auckland that argues with the SPAIM capabilities in terms of manufacturing efficiency and interoperability. The IIMP improved the data portability and process interoperability between heterogeneous formats of the CAD/CAM/CNC systems. The architecture of IIMP is composed of three elements: orchestrator, application module pool, and execution core. The main aim of IIMP is to gather and synchronize heterogeneous data among existing and future CAD/CAM/CNC systems [19]. Going even further beyond the capabilities of SPAIM and IIMP frameworks, an eXtended Manufacturing Integrated System (XMIS) for feature-based manufacturing with STEP-NC was developed. The XMIS is composed of a production project unit that includes planning, process analysis, process control, and process diagnosis units. The focus of XMIS is to control the manufacturing process from design to product stages [45]. In interpreted STEP-NC programming category, a self-learning, intelligent, and efficient CNC controller based on OAC was developed within the scope of the FP7 European Commission funded project. This FoFdration Smart Machine Controller Open Architecture

(FSMC-OA) project focused on converting industrial CNC (Cincinnati Milacron “Sabre” milling machine with NUM controller) into a unique multi-controller Integrated Test Platform (ITP). The ITP multi-controller platform contains the Linux CNC Enhanced Machine Controller (EMC)2 OAC, the legacy nC-12 controller from Fidia, and the NUM 750 numeric controller. The aim of ITP is to provide an all-in-one platform for STEP-NC demonstration by utilizing multi-controllers on one machine [20]. Malaysia universities are also active in this field of research; they have developed a three-axis CNC milling machine system based on interpreted STEP-NC programming approach by utilizing virtual component and open architecture control technologies [46].

The direct interpretation of STEP-NC enables more flexibility into the CNC system. This implementation makes CNC controller intelligent and self-learning. This category-based CNC controller provides shop floor programming, optimization, simulation, inspection, and other facilities within the core of the CNC controller. However, the dream of STEP-NC controller is yet to be achieved. The ultimate goal of STEP-NC is to enable Adoptive STEP-NC programming approach where the controller estimates online process data and optimizes machining parameters and tool paths in real time. At present, there is no prototype of this category implemented on any machine. The outcome of these demonstrations proved that STEP-NC is able to provide product and process data, but there is no commercially available implementation until now. It is because the manufacturers believe that conventional CNCs are powerful enough to handle machining tasks. The CNC and CAM vendors seem to have a limited concern in promoting STEP-NC due to their personal business interests. In fact, they have misread the vision of STEP-NC on the next generation of intelligent control, and that will be a drawback to vendors in the future [47]. Due to these reasons, this modification also requires a high cost. Overall, these factors have slowdown STEP-NC implementation on a commercial CNC machine. But fortunately, the introduction of OAC technology provides another platform for this implementation [36]. The introduction of OAC increases the possibility of STEP-NC implementation by providing more power to PCs [48]. The aim of OAC is to enable easy implementation and integration of customer-specific controls, reusability of software, and to integrate user-specific algorithms or application. OAC interfaces can play a vital role in the creation of reconfigurable CNC units. The performance of the unit can be easily improved by upgrading the hardware platform only [29]. Due to all these facilities of OAC, the modern CNC developers favor PC-based solutions to provide high flexibility and quality at low cost [25].

Table 1 Integrated CNC Systems from 2003 to 2020

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feedback	Monitoring	Input /Output	Other functions
[49], 2003	2 PCs(PC+ commercial controller) + MMC	Linux and Linux RT	C and PLC	ISO 6983	Yes(2 axis and spindle)	Yes	None	No	No	No	Yes	No
[50], 2004	ComputCam's motion control system	Linux /Windows	6K Motion Control VB, VC++ and Delphi	ISO 14649	Yes(2 axis lathe)	STEPcNC generates and translates STEP-NC into CNC understandable pattern for lathe only	Interpreted	No	Yes [38]	No	Yes	No
[34], 2006	Commercial Siemens 840D	Commercial machine environment	JAVA	ISO 14649	Yes(3 axis milling)	AB-CAM (Generates STEP-NC file and translate into G codes)	In-Direct	No	No	No	No	No
[28], 2006	2PCc(PC+ commercial controller)	Windows Xp professional	PLC	ISO 6983	Yes(3 axis and spindle)	Yes (Utilized commercial)	None	No	No	No	Yes	No
[18], 2006	PC + PCI 7212c motor control board, AD, DA and DIO boards	Windows 98, NT and 2000	VC++	ISO 14649-28	Yes (3 and 5 axis and spindle)	Converts STEP part 21 into STEP part 28 and generates tool path	In-Direct and Interpreted	No	Yes	No	Yes	No
[51], 2006	FBICS implemented commercial CNC	Commercial machine environment	C++, ST developer with Rose Library	ISO 14649 and 14649-238	Yes(3 axis and spindle)	STEP-NC part 21 with tool path for FBICS	In-Direct	No	Yes	No	No	No
[52], 2007	PC + OMAC	Win32 and RTx	Visual C++ and PLC	ISO 6983	Yes(3 axis and spindle)	Yes	None	No	No	No	Yes	No
[53], 2007	PC + SERCOS communication card	Linux (RT)	PLC and VC++	ISO 6983	Yes(3 axis and spindle)	No	None	No	No	No	Yes	No
[8], 2007	PC + Embedded microcontroller based on RCM 3700	Linux	C	ISO 14649	Yes(3 and 5 axis and spindle)	STEP-NC part 21 with tool path (pocket, slot and round hole)	Interpreted	No	Yes	Yes	Yes	No
[54], 2007	Commercial controller modified by own new developed MMC based on DSP and FPGA	Windows CE.net	Commercial application modified with Bresenham algorithm for 2D simulation and DirectX3D graphic library for 3D simulation	ISO 6983	Yes(3 axis milling bed)	G code interpreter	None	2D/3D G code simulator	No	No	No	No
[55], 2008	PC + OMAC + SERCANS	Windows	PLC	ISO 6983	Yes(5 axis and spindle)	G code(linear and spline)	None	No	No	No	Yes	No

Table 1 (continued)

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feed-back	Monitoring	Input /Output	Other functions
[56], 2008	PC + Field bus interface card	Windows	PLC, C++, ST developer with ROSE and STIX libraries	ISO 14649	Yes(3 axis and spindle)	STEP-NC interpreter Converts EXPRESS into C++ classes	Only frame-work (Interpreted)	No	No	No	No	No
[36], 2009	PC based EMC controller	Linux	Functional Blocks and JAVA	ISO 14649	Yes(3 axis mill bed)	Translates STEP into G code	In-Direct	Yes(3D visualization of tool path)	No	No	Yes	Yes(Functional Blocks in CNC development)
[30][57], 2009	PC + OMAC+ SERCANS board + DAQ board and force sensor	Windows (RTX)	VC++ and Soft SERCANS	ISO 6983	Yes(3 and 5 axis milling)	G code	None	No	No	Yes(Cutting Force)	Yes	No
[58], 2009	Commercial key board panel + PC + Customized MMC and I/O card	Windows Embedded 6.0	Visual Studio, Net, C# and C++	ISO 6983	Yes(3 axis milling, spindle and coolant)	G code translator	None	Yes	No	No	Yes	No
[26], 2009	2 PCs + PCI 8174 MMC + Ethernet card, DSP chip and ASIC motion control kernel	Windows	VC++, Matlab(neural networks)	ISO 6983	Yes(3 axis milling)	G code data mining algorithm(Depth of cut, spindle speed and feed rate)	None	No	No	Yes(Temperature and Surface finish)	No	Yes(Surface finish prediction via NN)
[59], 2009	Modified commercial controller + Mechatrolink-II MMC + network analyzer and I/O card	RT Linux	Red Hat Linux 9.0, Linux Kernel 2.4.20, Real Time Linux 3.1 and commercial CNC software	ISO 6983	Yes(4 axis milling machine)	G code Commercial system	None	No	No	No	Yes	No
[60], 2010	Modified commercial controller with DRC + MPU + SIM + FPGA + MMC + Vibration and Current analyzer	Linux Ubuntu and Windows Xp	Prtri net, GTK ++ and VC ++	ISO 6983	Yes(2 axis CNC lathe)	No	None	No	No	Yes(Current and Vibration)	Yes	Yes (Vibration and current analyzes)
[61], 2010	PC + PMAC + I/O	Windows	VC++ and ant colony algorithm	ISO free environment	Yes(3 milling)	Hoc format data generator	None	No	No	No	Yes	Yes (Feature recognition from 2D CAD design to generate automatic tool paths in hoc format)
[62], 2010	PC + AT8952 Microcontroller + RS 232 to TTL converter + Drivers ULN 2003 and Driver ULN 2803	Windows Xp	C#Net platform 3.5 and Kiel compiler	ISO free environment	Yes(3 milling)	Binary code generator	None	No	No	No	No	Yes(Image to binary conversion and code generation)

Table 1 (continued)

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feed-back	Monitoring	Input /Output	Other functions
[63], 2010	Modified commercial TDNCM4 controller with motion control chip and ARM	µC/OS-II	Commercial TDNCM40A application	ISO 6983	Yes(3 axis and spindle)	Commercial code	None	No	No	No	No	No
[4pt] [64], 2010	Commercial controller modified by own new developed MMC based on DSP and FPGA	Windows CE.net 32 bit embedded system	Commercial application modified with Bresenham algorithm for 2D simulation and DirectX3D graphic library for 3D simulation	ISO 6983	Yes(2 axis lathe)	G code interpreter	None	3D G code simulator	No	No	No	No
[4pt] [65], 2010	Commercial FANUC series + PC + motion control card + NC control board	MS Windows + NC operating system	Lab VIEW	ISO free Environment	Yes(3 axis motion control)	No	None	No	No	No	No	No
[4pt] [66], 2010	PC + micro controllers (PIC18F2550 and PIC16F677) + USB ports	Windows	Lab VIEW (Motion control) and MATLAB (Simulation)	ISO free Environment	Yes(3 axis motion control)	No	None	No	No	Yes (Motor drive current monitoring)	No	Electrical features monitoring through simulation on MATLAB
[4pt] [67], 2011	EtherMAC	Linux RTAI	OWL (Machine knowledge base system) and MATLAB Neural Networks (tool wear monitoring) C and PLC	ISO 14649-21	No (only frame-work)	STEP-NC interpreter with tool path generation based on OWL KB	None	No	Yes	Yes (Tool wear)	Yes	KB system based on OWL and Tool wear monitoring based on MATLAB (NN). No
[4pt] [68], 2012	PC + GT400-SV MMC + Card special CPU + PC's master slave dual control and Linux RTAI	Windows		ISO 6983	Yes(3 axis and spindle)	G code interpreter for B-Spline curve	None	No	No	No	No	
[4pt] [69], 2012	Red Hat Linux -200MHz machine	Linux	C++ (Interpreter) and MATLAB (Simulator)	ISO 6983	No (Cause only developed software module)	G-Code interpreter with tool path planning module	None	2DSimulator (MATLAB)	No	No	No	Simulation of interpreted code on MATLAB
[4pt] [70], 2012	PC + PCI-1243 MMC + ZETA Drive	Windows	VB	ISO free environment	Yes(3 axis stepper drives)		None	No	No	No	No	No
[4pt] [71], 2012	Prototype (2 printer, 1 scanner and 1 doorway) + PC + microcontroller	Windows	JAVA	ISO 14649-21	Yes(3 axis pro-and spindle) totype test bed	STEP-NC interpreter for drilling operations	None	No	No	No	No	No

Table 1 (continued)

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feedback	Monitoring	Input /Output	Other functions
[72], 2013	PC + Arduino Leaf Maple Micro controller	Linux and Windows	QT framework (HMI) CCC (G code conversion)	ISO 6983	Yes(2 axis CNC lathe)	G code interpreter for lathe that translate g code into NCC commands for Arduino	None	No	No	No	No	No
[73], 2013	Commercial XLTURN CNC with FANUC controller	Commercial machine environment	C++	Reads ISO 14649-AP203 and Convert to ISO 6983	Yes (2 axis CNC lathe)	Extract feature from STEP-NC file and generates G codes	In-Direct	No	No	No	No	No
[74], 2013	Commercial EMC2 or Linux CNC	Linux	C++, STEP-NC Modular, S Adopter	Read ISO 14649-21 and converts to ISO 6983	Yes(3 axis test bed)	Generate STEP-NC file (STEP-NC modular) and converts to G code (STEP-NC adopter)	In-Direct	No	No	No	No	No
[21], 2013	Commercial Stemen 840 D	Commercial machine environment	JAVA, Inter-operable CAD/CAM system	Reads ISO 6983 code and generates ISO 14649	Yes(Commercial 3 axis CNC)	Reads G code and generates STEP-NC file	In-Direct	No	No	No	No	No
[75], 2013	Commercial Siemens Sinumerik 840 Di + Server	Commercial machine environment	JAVA, ST Developer	ISO 14649	Yes(Commercial 3 axis CNC)	Generates STEP-NC file by utilizing EXPRESS to JAVA transformation via SDAI	In-Direct	2D Simulator	No	No	No	No
[20], 2014	Commercial CNC Cinemat Milacron "Sabre" milling machine with NUM controller + Linux CNC (EMC2) OAC + the legacy nC-12 controller from Fidia and the NUM 750 numeric controller	Commercial machine environment	Commercial machine environment	ISO 14649	Yes(Commercial 3 axis CNC)	None	In-Direct	No	No	No	No	No
[76], 2014	PC + motion control card + UMI	Windows Xp	LabVIEW, ST Developer, JAVA	ISO 14649	Yes(3 axis and spindle)	No(Uses ST developer and JAVA for this)	In-Direct	No	No	No	No	No
[77], 2014	PC + Arduino MC + Three H bridge and motor drives	Windows	MATLAB	No (Only motion control technique)	Yes (3 axis test bed)	None	None	No	No	No	No	Image processing technique for motion control

Table 1 (continued)

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feedback	Monitoring	Input /Output	Other functions
[78], 2014	Commercial FANUC LOLA pn101 4V2 and HMC500	Commercial Machine Environment	Commercial (CAD/CAM + STEP-NC Machine + operating software)	Reads generates ISO14649 and converts to ISO 6983	Commercial Units (3 or 2 axis)	Reads generates STEP-p21 or p28 file and converts to g code for FANUC via STEP-NC Machine software	In-Direct	Commercial STEP-NC Machine software	No	No	No	No
[79], 2014	HIT-CNC	Win32 and RTX	CAD/CAM for design, for C++ interpreter, HIT-CNC for operations	ISO 14649	Yes (3 axis milling)	STEP-NC	Interpreted (conceptual)	No	Yes	Yes	Yes	No
[80], 2015	PC (MMI + PLC+NCK) + PLC IO + sensors + servo drives and Ethernet card	Windows	Commercial CATIA + CAA (V5, C++),Soft PLC + Twin-CAT and EtherCAT	ISO 14649	Yes (3 axis milling)	Reads and generates STEP-21 or 28 files	Interpreted (conceptual)	No	No	No	Yes	No
[81], 2016	PC based CNC	Windows and Milling Commercial system	ARToolKit, Microsoft Visual C++, OpenGL	ISO 6983	None	G Code	None	Yes (AR Simulator)	No	No	No	Yes (Use of AR Technology)
[82], 2016	Linux CNC	Linux	Glade, JAVA	ISO 14649	Machine Robot	Yes(using generic Lin-cnc controller)	None	No	No	Yes(using open source MITConnect)	No	No
[83], 2016	Virtual Machining Model (STEP2M)	Commercial CNC	Java,Prime Face	ISO 14649	Yes (2 axis)	Yes(from STEPNC to NC part programme)	In-Direct	No	No	Yes (using open source MITConnect)	No	No
[84], 2016	Open Modular Architecture Controller (OMAC)	Commercial system	Based on CNC kernel and EXPRESS	ISO 14649	Yes	Yes	Interpreted	No	No	Yes (using open source MITConnect)	No	No
[85], 2016	Web based CAM	Windows	JAVA, Eclipse	ISO 14649	Yes	No	In-Direct	No	No	No	No	No
[86], 2017	Closed-loop machining based on STEP-NC	Windows	CATIA R20 Microsoft Visual Studio	ISO 14649	Yes	Yes	Interpreted	No	No	No	No	No
[46], 2017	PC + MMC + Denford NovaMill	Windows	LabVIEW	ISO 14649	Yes(3axis Milling)	Interprets STEP-part 21 file (face, hole, pocket)	Interpreted	Yes 3D Simulator based on LabVIEW	Yes	Yes(Live monitoring based on LabVIEW)	Yes	Yes (Interpretation, 3D simulator, live monitoring and automatic generation modules based on LabVIEW.

Table 1 (continued)

Developer	Commercial /PC based	Operating system	Developing platform	ISO data model	Motion control	Interpreter	STEP-NC programming approach	Data simulation	Data feedback	Monitoring	Input /Output	Other functions
[87], 2018	PC with online storage platforms	Commercial CNC system	OWL, JAVA, MapReduce	ISO 14649	No	No	In-Direct	No	No	No	No	No
[88], 2018	ASEA IRB6-S2 commercial controller	PTC Creo, STEP Machine	ASEA	ISO 14649	IDEF0 model robot	Yes	Interpreted	No	No	No	No	No
[89], 2019	Built in Grapetec CNC	GrapeSim commercial system	GrapeTec	ISO 14649	Yes(from Grapesim commercial simulation)	No	In-Direct	No	No	No	No	No
[90], 2019	GrapeSim	GrapeSim commercial system	GrapeTec, CATIA	ISO 14649	None (only simulation)	Yes (from STEP commercial)	None	No	No	Yes (using open source MTCconnect)	No	No
[91], 2019	Cloud environment STEP NC	Web server	MT Connect TCP/IP	ISO 14649	None	STEP-NC	None	No	No	No	No	No
[92], 2019	EtherMAC	Windows	EXPRESS	ISO 14649	Yes (2 axis)	Yes	Interpreted	No	No	No	No	No
[93], 2020	PC with NVIDIA Geforce GT 705 graphics card	Windows	C++	ISO 14649	None (Simulation Only)	No	None	No	No	No	No	No

4 Integrated CNC Systems

In order to facilitate the development of STEP-Compliant systems, the various areas need to be focused, as shown in Fig. 1. Based on this knowledge, many researchers had proposed different approaches for the development of modern CNC systems. The summaries of the developments from year 2003 to year 2020 are presented in Table 1.

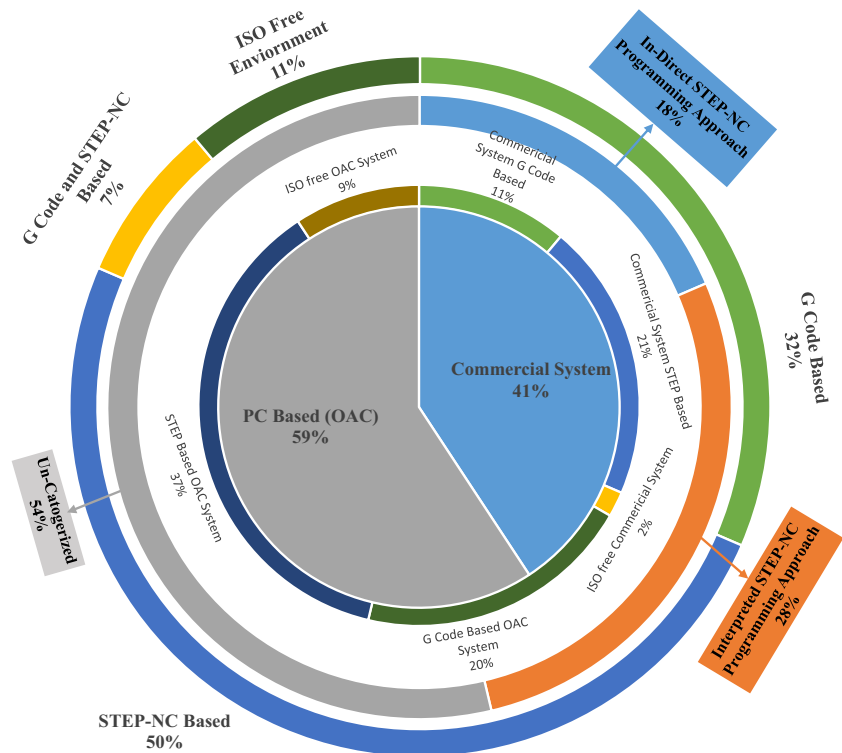
Several systems have been developed by using the commercial, PC-based, and integrated hardware configurations. Most of the techniques are created on Windows, Linux, commercial, and integrated operating systems. In this review, the developed methods are categorized into different fields such as ISO data interface model, STEP-NC programming approaches, and OAC technology. The integration of these technologies has been utilized for CNC system development. The yearly growth of this integration can be observed in Fig. 2. For many years, the OAC-based CNC systems are in the main focus compared to the commercial CNC systems. The development of OAC-based CNC system has been carried out on both ISO data interface models. The trend of this development is shown in Fig. 3, which indicates that STEP-NC-based CNC system development is higher than others. The STEP-NC has been implemented on commercial CNC and OAC platforms as

per its implementation techniques (Indirect and Interpreted) as shown in Fig. 4. This observation indicates that the current trend of CNC development is the STEP-NC based OAC CNC systems.

5 Conclusion

In the content of the paper, Computer Numeric Control (CNC), Computer-Aided Design (CAD), Computer-Aided Process Planning (CAPP), Computer-Aided Manufacturing (CAM), G codes (ISO 6983), STEP (ISO 1303), STEP-NC (ISO 14649), and Open Architecture Control (OAC) technologies are discussed from development to implementation. Over 50 embedded CNC systems are reviewed based on different categories, which are summarized in Fig. 5. From this review, it has been observed that there is much effort that had been carried out for the implementation of the STEP-NC data interface model by utilizing various techniques and tools. As discussed, the implementation was initiated on a commercial CNC system by the In-Direct STEP-NC programming approach, but that was not able to provide full features of the STEP-NC because it translates the STEP data into G codes. Secondly, commercial machine units were found to be of a closed nature. In order

Fig. 5 Summary of the reviewed works



to overcome these issues, the Interpreted STEP-NC programming approach implementation was suggested. But, this technique was a bit costly in implementation. Therefore, the utilization of open-architecture control technology is considered. This integration opens the door for the various techniques, which are mentioned in Table 1. However, all of these techniques have some limitations in terms of control, hardware, software, modern functionalities, and other factors. At present, there is no any STEP-CNC system has been commercialized because the aim of next-generation CNC had not been achieved yet. However, STEP-NC has successfully enabled lots of modern functionalities into the CNC, but there is more that is still to be achieved. This scenario indicates that the STEP-based CNC development is an active topic for research, in which various fields need to be addressed, such as an interpretation, simulation, hardware configuration, software configurations, optimization, monitoring, communication, safety, and e-manufacturing. To enable modern features in the CNC system, the STEP-NC data interface model is better than the G Code because it provides complete access to the design and manufacturing data. This data can be used in online and offline environments. However, for the implementation method, the Interpreted and Adaptive STEP-NC programming approaches are the main directions for future research as the In-Direct STEP-NC programming approach is not capable enough to achieve the aim of the future CNC. The content of the paper has clearly shown how the integration of various technologies can achieve these approaches. As in previous approaches, many researchers had utilized different integrated platforms in the system development, which are based on ISO 6983, ISO 14649, both and neutral data interface models. However, it is always a good idea to focus on the new ISO data interface model (ISO 14649 and ISO 10303-238) rather than ISO 6983 or neutral. These new data interface models are proved to be very beneficial as compared to old, which is clarified in the paper. These capabilities make them strong enough to address future challenges. These data interface models can be used with open architecture control technologies, micro-controllers, arduino, LabVIEW, MATLAB, Python, JAVA, C, and other wide range of hardware and software configurations for the development of low-cost CNC systems or prototypes.

Acknowledgements The authors would like to thank the Sustainable and Responsive Manufacturing (SUSREM) research group, and the Center of Research and Innovation Management (CRIM) of Universiti Teknikal Melaka (UTeM). The authors also acknowledge the support of the Advanced Manufacturing and Material Centre (AMMC) of Universiti Tun Hussein Onn Malaysia (UTHM), and Production Laboratory of Universiti Teknologi Malaysia (UTM).

Author contribution Kamran Latif and Anbia Adam contributed to the literature research. Kamran Latif, Yusri Yusof, and Aini Zuhra Abdul Kadir contributed to the data optimization, analysis, and structure. Kamran Latif contributed to the writing of the manuscript.

Funding This work is supported by the Ministry of Higher Education (MoHE) Malaysia under the Prototype Research Grant Scheme (PRGS) number PRGS/2020/FTKMP-COSSID/T00026.

Data availability All data generated or analyzed during this study are included in this published article and available at the corresponding author.

Declarations

Ethics approval This paper does not contain any studies with human participants or animals performed by any of the authors.

Conflict of interest The authors declare no competing interests.

References

- Liana SY, Hecker RL, Landers RG (2004) Machining process monitoring and control: the state-of-the-art. *Journal of manufacturing science and engineering* 126(2):297–310
- Safaieh M, Nassehi A, Newman S (2013) A novel methodology for cross-technology interoperability in CNC machining. *Robot Comput Integr Manuf* 29(3):79–87
- Newman S, Nassehi A, Xu X, Rosso R, Wang L, Yusof Y, Ali L, Liu R, Zheng L, Kumar S (2008) Strategic advantages of interoperability for global manufacturing using CNC technology. *Robot Comput Integr Manuf* 24(6):699–708
- Suh SH, Lee B, Chung D, Cheon S (2003) Architecture and implementation of a shop-floor programming system for STEP-compliant CNC. *Comput Aided Des* 35(12):1069–1083
- Xu X, He Q (2004) Striving for a total integration of CAD, CAPP, CAM and CNC. *Robot Comput Integr Manuf* 20(2):101–109
- Yusof Y, Kassim ND, Zamri Tan NZ (2011) The development of a new STEP-NC code generator (GEN-MILL). *Int J Comput Integr Manuf* 24(2):126–134
- Xu X, Newman S (2006) Making CNC machine tools more open, interoperable and intelligent—a review of the technologies. *Comput Ind* 57(2):141–152
- Calabrese F, Celentano G (2007) Design and realization of a STEP-NC compliant CNC embedded controller. In: *Emerging technologies and factory automation ETFA., Patras. IEEE*, pp 1010–1017
- Sääski J, Salonen T, Paro J (2005) Integration of CAD, CAM and NC with STEP-NC Espoo, VTT
- Shin SJ, Suh SH, Stroud I (2007) Reincarnation of G-code based part programs into STEP-NC for turning applications. *Comput Aided Des* 39(1):1–16
- Xu X, Wang H, Mao J, Newman S, Kramer T, Proctor F, Michaloski J (2005) STEP-compliant NC research: the search for intelligent CAD/CAPP/CAM/CNC integration. *Int J Prod Res* 43(17):3703–3743
- Krzic P, Stoic A, Kopac J (2009) STEP-NC: A new programming code for the CNC machines. *Strojniški vestnik* 55(6):406–417

13. Nassehi A (2007) The realisation of CAD/CAM/CNC interoperability in prismatic part manufacturing. *Diss Abstr Int* 68(4)
14. Wang H (2009) New control strategy for CNC machines via STEP-NC. (PhD) ResearchSpace@ Auckland
15. Kramer T, Xu X (2009) STEP in a Nutshell Advanced design and manufacturing based on STEP. Springer, Berlin, pp 1–22
16. Cai J, Weyrich M, Berger U (2005) Ontological machining process data modelling for powertrain production in extended enterprise. *J Adv Manuf Syst* 4(01):69–82
17. Hardwick M, Zhao YF, Proctor F, Nassehi A, Xu X, Venkatesh S, Odendahl D, Xu L, Hedlind M, Lundgren M (2013) A roadmap for STEP-NC-enabled interoperable manufacturing. *Int J Adv Manuf Technol* 68(5-8):1023–1037
18. Lee W, Bang Y, Ryou M, Kwon W, Jee H (2006) Development of a PC-based milling machine operated by STEP-NC in XML format. *Int J Comput Integr Manuf* 19(6):593–602
19. Hamilton K, Hascoet JY, Rauch M (2014) Implementing STEP-NC: Exploring possibilities for the future of advanced manufacturing Modern Mechanical Engineering. Springer, Berlin, pp 199–239
20. Rauch M, Laguionie R, Hascoët J. Y., Xu X (2009) Enhancing CNC manufacturing interoperability with STEP-NC. *J Mach Eng* 9(4):26–37
21. Zhang X, Nassehi A, Safaieh M, Newman S (2013) Process comprehension for shopfloor manufacturing knowledge reuse. *Int J Prod Res* 51(23-24):1–15
22. Ridwan F, Xu X, Liu G (2012) A framework for machining optimisation based on STEP-NC. *J Intell Manuf* 23(3):423–441
23. Consortium IS-N (2003) Technical Report 3 of IMS Project (97006) STEP-compliant data interface for numerical controls (STEP-NC) Report Period (Vol 1)
24. Muller P, Hyun Y (2001) ESPRIT Projekt EP29708 STEP-Compliant data interface of numerical controls (STEPNC) Final report, STEP-NC Consortium. pp 1–28
25. Cha JM, Suh SH, Hascoet JY, Stroud I (2014) A roadmap for implementing new manufacturing technology based on STEP-NC. *J Intell Manuf* 1–15
26. Ramesh R, Poo A (2009) Intelligent Ethernet based open architecture control system for machine tools. In: *Intelligent computing and intelligent systems ICIS*, Shanghai. IEEE, pp 612–616
27. Dong Y, Hu L, Rui Feng G, Jiangang Y, Pengfei X (2005) The research on real-time middleware for open architecture controller. In: *Embedded and real-time computing systems and applications*. IEEE, pp 80–83
28. Wu H, Zhang C, Li G, Wang B (2006) Windows XP embedded based open architecture computer numerical control system. In: *Mechatronic and embedded systems and applications*. IEEE, pp 1–6
29. Brecher C, Verl A, Lechler A, Servos M (2010) Open control systems: state of the art. *Prod Eng* 4(2-3):247–254
30. Maoyue L, Hongya F, Yuan L, Zhenyu H (2009) Research on reusable and configurable intelligent machining system. In: *Industrial electronics and applications*. IEEE, pp 3130–3133
31. Rauch M, Laguionie R, Hascoet JY, Suh SH (2012) An advanced STEP-NC controller for intelligent machining processes. *Robot Comput Integr Manuf* 28(3):375–384
32. Weck M, Wolf J, Kiritsis D (2001). In: *International intelligent manufacturing system forum. STEP-NC—The STEP compliant NC programming interface*
33. Newman S, Allen R, Rosso RJr (2003) CAD/CAM solutions for STEP-compliant CNC manufacture. *Int J Comput Integr Manuf* 16(7-8):590–597
34. Nassehi A, Newman S, Allen R (2006) The application of multi-agent systems for STEP-NC computer aided process planning of prismatic components. *Int J Mach Tools Manuf* 46(5):559–574
35. Wang H, Xu X, Des Tedford J (2007) An adaptable CNC system based on STEP-NC and function blocks. *Int J Prod Res* 45(17):3809–3829
36. Minhat M, Vyatkin V, Xu X, Wong S, Al Bayaa Z (2009) A novel open CNC architecture based on STEP-NC data model and IEC 61499 function blocks. *Robot Comput Integr Manuf* 25(3):560–569
37. Xu X (2006) Realization of STEP-NC enabled machining. *Robot Compu Integr Manuf* 22(2):144–153
38. Zhao YF, Habeeb S, Xu X (2009) Research into integrated design and manufacturing based on STEP. *Int J Adv Manuf Technol* 44(5-6):606–624
39. Weck M, Wolf J (2002) ISO 14649 provides information for sophisticated and flexible numerically controlled production. *Prod Eng [WGP-Annals]*
40. Storr A, Pritschow G, Heusinger S, Azotov A (2002) Workingstep planning for turning with STEP-NC: planning methods for user support. *IWF Zeitschrift für Wirtschaftlichen Fabrikbetrieb* 97(7-8):390
41. Suh SH, Cheon SU (2002) A framework for an intelligent CNC and data model. *Int J Adv Manuf Technol* 19(10):727–735
42. Denkena H, Tönshoff J, Selle A, Storr S, Heusinger S, Rogers G (2002) Offline-Berechnung der Zerspankräfte in der NC-Programmierung. Vorhersage der Zerspankräfte beim HSC-Schlichtfräsen
43. Erdős G, Xirouchakis P (2003) STEP-NC data model development for wire-EDM manufacturing IFAC
44. Richard J, Stark J (2002) Standardisation of the manufacturing process: the STEP-NC project. *IPLnet Workshop, Saas-Fee, I-tech, EIG. HES-SO*. pp 10–11
45. Laguionie R, Rauch M, Hascoët J. Y., Suh SH (2011) An extended manufacturing integrated system for feature-based manufacturing with STEP-NC. *Int J Comput Integr Manuf* 24(9):785–799
46. Latif K, Yusof Y, Nassehi A, Latif QBAI (2017) Development of a feature-based open soft-CNC system. *Int J Adv Manuf Technol* 89(1-4):1013–1024
47. Xiao W, Zheng L, Huan J, Lei P (2015) A complete CAD/CAM/CNC solution for STEP-compliant manufacturing. *Robot Comput Integr Manuf* 31:1–10
48. Suh SH, Kang SK, Chung DH, Stroud I (2008) *Theory and design of CNC systems*. Springer, Berlin
49. Zhang C, Wang H, Wang J (2003) An USB-based software CNC system. *J Mater Process Technol* 139(1):286–290
50. Xu X, Wang J (2004) Development of a G-code free, STEP-compliant CNC lathe. In: *ASME International mechanical engineering congress and exposition*. American Society of Mechanical Engineers, pp 75–82
51. Kramer TR, Proctor F, Xu X, Michaloski J (2006) Run-time interpretation of STEP-NC: implementation and performance. *Int J Comput Integr Manuf* 19(6):495–507
52. Ma XB, Han Z, Wang YZ, Fu H (2007) Development of a PC-based open architecture software-CNC system. *Chin J Aeronaut* 20(3):272–281
53. Qiang R (2007) Research of Software Open-CNC System, <http://www.paper.edu.cn>
54. Xiao S, Li D, Lai Y, Wan J, Feng S (2007) An open architecture numerical control system based on Windows CE. In: *Control and automation ICCA, Guangzhou*. IEEE, pp 1237–1240
55. Yuan L, Yong Zhang W, Hong Ya F (2008) An open architecture motion controller for CNC machine tools. In: *Systems and control in aerospace and astronautics ISSCAA*. IEEE, pp 1–4
56. Lan H, Liu R, Zhang C (2008) A multi-agent-based intelligent STEP-NC controller for CNC machine tools. *Int J Prod Res* 46(14):3887–3907

57. Maoyue L, Hongya F, Yuan L, Zhenyu H (2009) An intelligent controller based on constant cutting force for 5-axis milling. In: Information technology and computer science. IEEE, pp 237–241
58. Ekkachai K, Komin U, Chaopramualkul W, Tantaworrasilp A, Kwansud P, Seekhao P, Leelasawassuk T, Tanta-Ngai K, Tungpimolrut K (2009) Design and development of an open architecture CNC controller for milling machine retrofitting. In: ICCAS-SICE, Fukuoka. IEEE, pp 5629–5632
59. Yu D, Hu Y, Xu X, Huang Y, Du S (2009) An open CNC system based on component technology. IEEE Trans Autom Sci Eng 6(2):302–310
60. Morales Velazquez L, Romero Troncoso RdJ, Osornio Rios RA, Herrera Ruiz G, Cabal Yopez E (2010) Open architecture system based on a reconfigurable hardware software multi agent platform for CNC machines. J Syst Archit 56(9):407–418
61. Li P, Gao T, Wang J, Liu H (2010) Open architecture of CNC system research based on CAD graph-driven technology. Robot Comput Integr Manuf 26(6):720–724
62. Pabolu VK, Srinivas S (2010) Design and implementation of a three dimensional CNC machine. Int J Comput Sci Eng 2(8):2567–2570
63. Wang T, Liu Q, Wang L (2010) An RTOS-based embedded CNC system. In: Computer, mechatronics, control and electronic engineering (CMCE), TBD, Changchun, China. IEEE, pp 33–36
64. Xiao S (2010) An open-architecture embedded manufacturing control system. In: Measuring technology and mechatronics automation (ICMTMA). IEEE, pp 517–520
65. Weidong Y, Zhanbiao G (2010) An open CNC controller based on LabVIEW software. In: Computer application and system modeling (ICCASM), North University of China, Taiyuan, China. IEEE, pp V4-476–V474-479
66. Da Rocha P, Diogne de Silva e Souza R, De Lima Tostes ME (2010) Prototype CNC machine design. In: Industry applications (INDUSCON), Sao Paulo. IEEE, pp 1–5
67. Li P, Hu T, Zhang C (2011) A unified communication framework for intelligent integrated CNC on the shop floor. Procedia Eng 15:840–847
68. Xu XM, Li Y, Sun JH, Wang SG (2012) Research and development of open cnc system based on pc and motion controller. Procedia Eng 29:1845–1850
69. Chen L, Yu D, Zhang H, Geng C, Dong L (2012) Design and implement of a modularized CNC interpreter based on the integration of tool path planning module. In: Computer science and automation engineering (CSAE). IEEE, pp 613–616
70. Chunhao L, Lijin G, Jingdong L (2012) Research of motion control system based on PCI-1243. In: Digital manufacturing and automation (ICDMA). IEEE, pp 662–665
71. Pacheco NdO, Harbs E, Rosso RS Jr, Hounsell MdS, Ferreira JCE (2012) Application of the step-nc standard in a computer numerical controlled machining device. In: ABCM Symposium Series in Mechatronics, vol 5, pp 713–723
72. Khanna A, Kumar A, Bhatnagar A, Tyagi R, Srivastava S (2013) Low-cost production CNC system. In: Intelligent systems and control (ISCO), Coimbatore, Tamil Nadu, India. IEEE, pp 523–528
73. Sivakumar S, Dhanalakshmi V (2013) A feature-based system for CAD/CAM integration through STEP file for cylindrical parts. Indian J Eng Mater Sci 20(1):21–26
74. Gutierrez ME, Álvares AJ (2013) Development of a cnc router adherent to standard STEP-NC based on the controller advanced machine (EMC2)
75. Benavente JCT, Ferreira JCE, Goulart CM, Oliveira VGd (2014) A STEP-NC compliant system for the remote design and manufacture of mechanical components through the Internet. Int J Comput Integr Manuf 26(5):412–428
76. Elias D, Yusof Y, Minhat M (2014) An open STEP-NC controller via labview platform. Appl Mech Mater 660:873–877
77. Sarhan H (2014) A novel technique for controlling CNC systems. Control Theory and Inform 4(5):82–92
78. Živanović S, Glavonjić M (2014) Methodology for implementation scenarios for applying protocol STEP-NC. J Prod Eng 17(1):71–74
79. Po H, Hongya F, Zhenyu H, Dedong HA (2014) closed-loop and self-learning STEP-NC machining system. In: Advanced intelligent mechatronics (AIM). IEEE, pp 1598–1603
80. Xiao W, Zheng L, Huan J, Lei P (2015) A complete CAD/CAM/CNC solution for STEP-compliant manufacturing. Robot Comput Integr Manuf 31:1–10
81. Yap HJ, Pai YS, Chang S-W, Yap KS (2016) Development of an augmented reality-based G-code generator in a virtual Cnc milling simulation. Int J Comput Sci Eng (IJCSSE) 5(2):63–72
82. Álvares A. J., Paredes MEG, Ferreira JCE, Benavente JCT (2016) A web-based STEP-NC-compliant architecture for low cost 3D part manufacturing. Int J Manuf Res 11(1):1–27
83. Shin SJ, Woo J, Kim DB, Kumaraguru S, Rachuri S (2016) Developing a virtual machining model to generate MTConnect machine-monitoring data from STEP-NC. Int J Prod Res 54(15):4487–4505
84. Hu P, Han Z, Fu Y, Fu H (2016) Implementation of real-time machining process control based on fuzzy logic in a new STEP-NC compatible system. Math Probl Eng
85. Ferreira JCE, Benavente JCT, Inoue PHS (2016) A web based CAD/CAPP/CAM system compliant with the STEP NC standard to manufacture parts with general surfaces. J Braz Soc Mech Sci Eng 2016(39):155–176
86. Lei P, Zheng L, Xiao W, Li C, Wang D (2017) A closed-loop machining system for assembly interfaces of large-scale component based on extended STEP-NC. Int J Adv Manuf Technol 91(5–8):2499–2525
87. Ye Y, Hu T, Zhang C, Luo W (2018) Design and development of a CNC machining process knowledge base using cloud technology. Int J Adv Manuf Technol 94(9–12):3413–3425
88. Toquica JS, živanović S, Alvares AJ, Bonnard R (2018) A STEP-NC compliant robotic machining platform for advanced manufacturing. Int J Adv Manuf Technol 95(9–12):3839–3854
89. Liu Yazui, Zhao G, Zavalnyi O, Cao X, Cheng K, Xiao W (2019) STEP-compliant CAD/CNC system for feature-oriented machining. Comput-Aid Des Appl 16(2):358–368
90. Zhao G, Cao X, Xiao W, Zhu Y, Cheng K (2019) Digital twin for NC machining using complete process information expressed by STEP-NC standard. In: ACM International conference proceeding series
91. Afanasev S, Zhao G, Xiao W (2019) Towards cloud-based STEP-NC to enhance interoperability in global manufacturing. IOP Conf Ser Mater Sci Eng 658(1)
92. Gao W, Zhang C, Hu T, Ye Y (2019) An intelligent CNC controller using cloud knowledge base. Int J Adv Manuf Technol 102(1–4):213–223
93. Zhao G, Cao X, Xiao W, Liu Q, Jun MBG (2020) STEP-NC feature-oriented high-efficient CNC machining simulation. Int J Adv Manuf Technol 106(5–6):2363–2375

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.