## RESEARCH ARTICLE

# The Architecture, Methodology and Implementation of STEP-NC Compliant Closed-Loop Robot Machining System

**PO HU**[ID]**, JIE LI, JINGBO GUO, LIANPENG ZHANG**[ID]**, AND JIE FENG**

State Key Laboratory of Mechanical Behavior and System Safety of Traffic Engineering Structures, Shijiazhuang Tiedao University, Shijiazhuang, Hebei 050043, China

School of Mechanical Engineering, Shijiazhuang Tiedao University, Shijiazhuang, Hebei 050043, China

Corresponding author: Jingbo Guo (guojingbo66@163.com)

**ABSTRACT** Industrial robots are gradually being employed in machining processes, particularly the cutting process, owing to their flexibility, mobility, and economic efficiency. However, it is difficult to make the manufacturing process intelligent owing to the complexity of robot machining process information handling and programming. In this paper, the architecture of a STEP-NC compliant closed-loop robot machining system was designed, including its function model and information stream. A methodology based on STEP-NC was established to enable the analysis of high-level information directly and automatically generating robot program according to the actual machining conditions. The STEP-NC Application Activity Model (AAM) and Application Reference Model (ARM) of closed-loop robot machining system is built to integrate the machining process data, monitoring and inspection data, mechanical equipment data, machining status data and inspection result data within a unified data flow, making it possible to realize intelligent manufacturing and adaptively adjusting the robot machining process. The proposed closed-loop robot machining system was implemented based on an open STEP-NC interpreter that interprets the high-level information in STEP-NC directly to reduce machining robot programming time. An industrial camera was integrated with the robot for rawpiece positioning, then the STEP-NC interpreter can generate robot path rapidly according to the parameters of manufacturing features and position of rawpiece. The STEP-NC interpreter can generate a robot control program or communicate with a software controller using an application program interface, so it can be integrated with both existing industrial robot controllers and future open robot controllers. Finally, case studies are conducted for the functional verification of the proposed STEP-NC compliant closed-loop robot machining system.

**INDEX TERMS** Closed-loop machining, machining robot, open STEP-NC interpreter, STEP-NC standard.

## I. INTRODUCTION

Industrial robots are widely used in welding, loading, discharge, conveying, and painting. They have also been used in machining processes such as grinding, polishing, drilling, milling, and additive manufacturing in recent years. Unlike machining tools, industrial robots are more flexible, cost-effective, and can be installed on mobile platform [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Liu[ID].

Therefore, it is more suitable for machining large-scale and complex-shape workpieces in single and small batch production. However, low machining precision and complexity of robot programming are the main problems to be solved in robot machining [2]. Firstly, the relative position from part to machining robot is changeable, especially in mobile machining. In that case, inspection, calibration, and regeneration of robot program are necessary to reduce positioning error when a workpiece is newly installed [3]. Secondly, structure deformation and cutting chatter caused by

low structural stiffness of machining robot is another main source of machining error. Furthermore, there is a complex correlation between structural stiffness of machining robot and its pose, making it difficult to predict and compensate the deformation or chatter during machining process [4]. Therefore, to minimize the deformation of robot structure and ensure good machining quality, the parameters of robot should be considered synthetically at the stage of process planning. Moreover, dynamic properties such as cutting force and vibration in machining process should be recorded in real-time to monitor the machining status, and to get posture more suitable for machining or to reduce the chatter in robot machining [5], [6].

Industrial robot manufactures usually have their own program formats, making the programming work complicated [7]. Meanwhile, robot control programs contained only trajectory of end operator and values of input and output ports. No high-level information, such as geometrical model of parts, manufacturing features, technological data, and robot parameters can be integrated in robot control programs. Real-time status information in the machining of every part is not recorded and associated with high-level information for further analysis. In this context, data model contains both high-level information of robot machining process and its real-time status is needed.

Closed-loop machining (CLM) is a methodology for controlling the entire procedure from part design to machining, monitoring, and inspection based on integrating of complete product manufacturing information including both process plan data, inspection data, and real-time monitoring data to improve machining accuracy and lower the cost [8]. The first step of CLM is to integrate online inspection, real-time monitoring with machining. Then adjust machining process according to inspection results and machining status. STEP-NC is a new standard that describe machining process using object oriented method retaining all high-level information and its interrelationship, which is suitable for using in CLM to connect procedures of part design, process planning, machining, inspection, and monitoring by a unique information stream [9]. Therefore, many research works of CLM were based on STEP-NC standard, online inspection and real-time monitoring. Brecher *et al.* defined STEP-NC data model related to touch probing as draft of ISO 14649-16 and performed inspection tasks on CMM (Coordinate Measuring Machine) [10]. The inspection results are written into STEP-NC file as feedback to make the machining process closed-loop. Kumar *et al.* proposed a STEP-compliant machining process control framework, using STEP-NC to represent machining knowledge for further analysis [8]. They also built a STEP-NC model for representing machine tool information and used ISO 14649-16 for representing inspection information, then using the inspection results for process control [11]. Zhao *et al.* proposed a framework of closed-loop machining based on online inspection and developed a STEP-NC interpreter to extract information of machining, inspection, and feedback [12]. They also

proposed a STEP-NC data model for on-machine dimensional measurement and system structure for integrated process planning and feedback [13]. Wosnik *et al.* presented a structured STEP-NC model for describing machining process data and built a process chain framework to optimize machining process [14]. Ridwan *et al.* proposed an architecture for machining condition monitoring based on STEP-NC in order to optimize feed-rate of CNC machining [15]. Sivakumar *et al.* proposed a methodology for the inspection and feedback of cylindrical parts based on geometrical data based on STEP standard to integrate various stages of product lifecycle [16]. The architecture and implementation of closed-loop machining system for CNC machining is proposed by the first author of this paper to apply on-machine inspection and process control based on an open STEP-NC controller [17]. Lei *et al.* built a STEP-NC data model for aligning and laser tracker-measuring process to realize closed-loop machining of large-scale component [18]. Danjou *et al.* proposed a closed-loop manufacturing approach focus on machining knowledge feedback based on STEP-NC, which used ontology method to extract and capitalize machining data for guiding future process planning [19], [20]. There is no CLM system for robot machining at present.

STEP-NC has been used in industrial robot and robot machining mainly on programming. Solvang *et al.* built a CAM system, which generate robot machining program based on STEP-NC process information [21]. Xiao *et al.* defined a STEP-compliant Industrial Robot Data Model (IRDM) for representing information of robot such as kinematic, geometric and controller and being used in robot off-line programming system [22]. Minango and Ferreira proposed a method for post-process of milling robot consists of tool paths generation and kinematic algorithm [23]. Zivanovic *et al.* developed a RoboSTEP-NC module to generate G codes for robot machining based on STEP-NC file [24]. The robot controller system used in their research is based on LinuxCNC platform, which can interpret G codes directly [25]. Toqiuca *et al.* built a STEP-NC compliant robot machining platform using PTC Creo, STEP-NC Machine and machining robot with LinuxCNC controller [26], [27]. In their research, kinematic parameters of machining robot are modeled in XML file and input to the system along with STEP format geometric model. Slakovic *et al.* used Robot Language Converter (RLC) to extract cutter location trajectory from STEP-NC file and transfer it into robot programming language [28]. Alvares *et al.* discussed and implemented six architectures for applying STEP-NC in robot machining [29]. The key issue is to handle the information in STEP-NC file and then to simulate or machine on robot platform. However, the information flow of robot machining based on STEP-NC in existing research works is unidirectional and only part of information is used, so the advantage of STEP-NC is not fully realized.

To sum up, it is a potential way to manage integrated robot machining process information by combining CLM, STEP-NC with robot machining, and further more

to achieve intelligent manufacturing by comprehensive analysis of integrated robot machining information. However, building a framework of robot machining system that can perform machining tasks intelligently and adaptively based on closed-loop data chain of high-level information, such as robot parameters, online-inspection result and real-time machining status, is still a problem to be solved.

The main purpose of this study is to build a framework of machining robot system that executing machining tasks adaptively, along with a closed-loop data stream integrated geometrical data of part, process plan data of machining, inspection, and monitoring tasks, robot and tool information, and status data during robot machining process. A Closed-Loop Robot Machining System (CLRMS) based on STEP-NC data model is proposed, implemented and verified. The Application Activity Model (AAM) and Application Reference Model (ARM) of the proposed CLRMS are built using functional modeling and object-oriented approach respectively. An open STEP-NC interpreter supports online STEP-NC file interpreting is developed for the implementation of the proposed system. The interpreter gets position of rawpiece by an industrial camera and generates robot path instantly. In this way, programming time for machining robot is significantly reduced.

The rest of paper is organized as follows. Section II describes the AAM of STEP-NC compliant closed-loop robot machining system, including functional activities and information stream. Section III defines the ARM of integrated STEP-NC data model for closed-loop robot machining. Section IV presents an open STEP-NC interpreter, which interprets STEP-NC file with no specified tool path directly and using an industrial camera for workpiece positioning. Finally, the implementation and validation of the proposed closed-loop robot machining system is presented in section V.

## II. ARCHITECTURE AND METHODOLOGY OF CLOSED-LOOP ROBOT MACHINING SYSTEM

The model of CLRMS based on STEP-NC consists of AAM and ARM, the former defines functional activities of the process and the latter defines information needed for the functional activities [30]. In this section, AAM of CLRMS is defined to illustrate the architecture and information flow of the closed-loop robot machining process. The ARM of CLRMS will be defined and discussed in section III.

### A. OVERALL STRUCTURE AND INFORMATION STREAM OF CLRMS

The objective of CLRMS is controlling robot machining process based on machining status and inspection results. As show in Fig. 1, the robot machining process is divided into five procedures, namely part design, integrated process planning, machining robot path planning, robot machining and process control, and machining knowledge management. These procedures are not executed sequentially but in parallel and collaboratively. All the procedures can get information of machining status in real-time and control the process
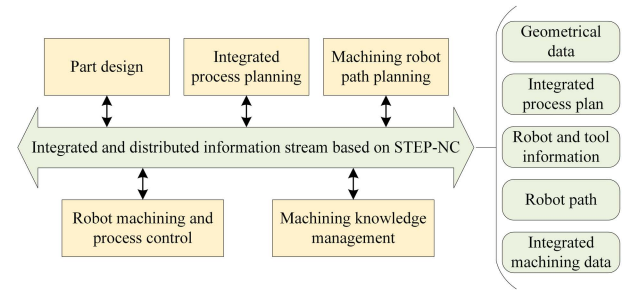


**FIGURE 1.** Closed-loop robot machining procedure and information stream.
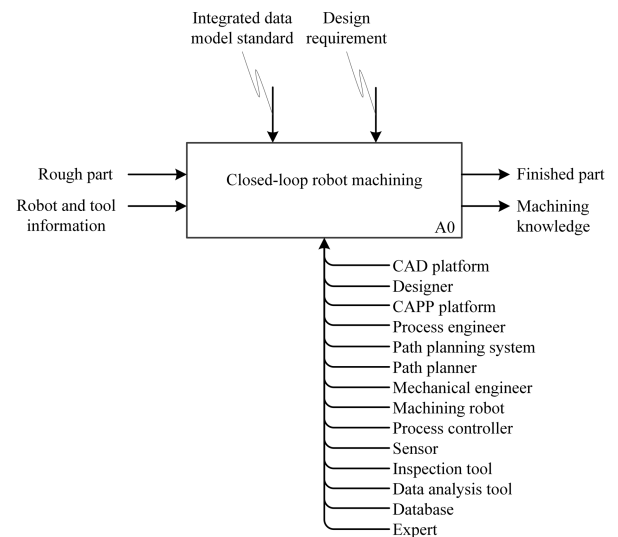


**FIGURE 2.** Top-level functional activity.

adaptively. An integrated and distributed information stream based on STEP-NC standard is used to represent all data in connection with machining process. Integrated means all information of geometrical data of parts, integrated process plan, robot and tool information, robot path and integrated machining data and their relationship are formatted in a consistent manner. Distributed means information collected during machining of every individual part should be stored separately to make the machining process traceable.

### B. FUNCTIONAL MODEL OF CLRMS

The AAM of closed-loop robot machining process is built by Integration DEFinition method 0 (IDEF0) for detailed analysis. The total process and procedures are represented using functional activities of IDEF0 diagram. Streams of input, output, control, and mechanism are represented using four groups of arrows around the functional activities. Fig. 2 is the top-level functional model of closed-loop robot machining. Inputs of activity A0 are rough part, and robot and tool information. Outputs of activity A0 are finished part and machining knowledge. Controls of activity A0 are design requirement and integrated data model standard compatible with STEP-NC. Mechanisms of activity A0 are the
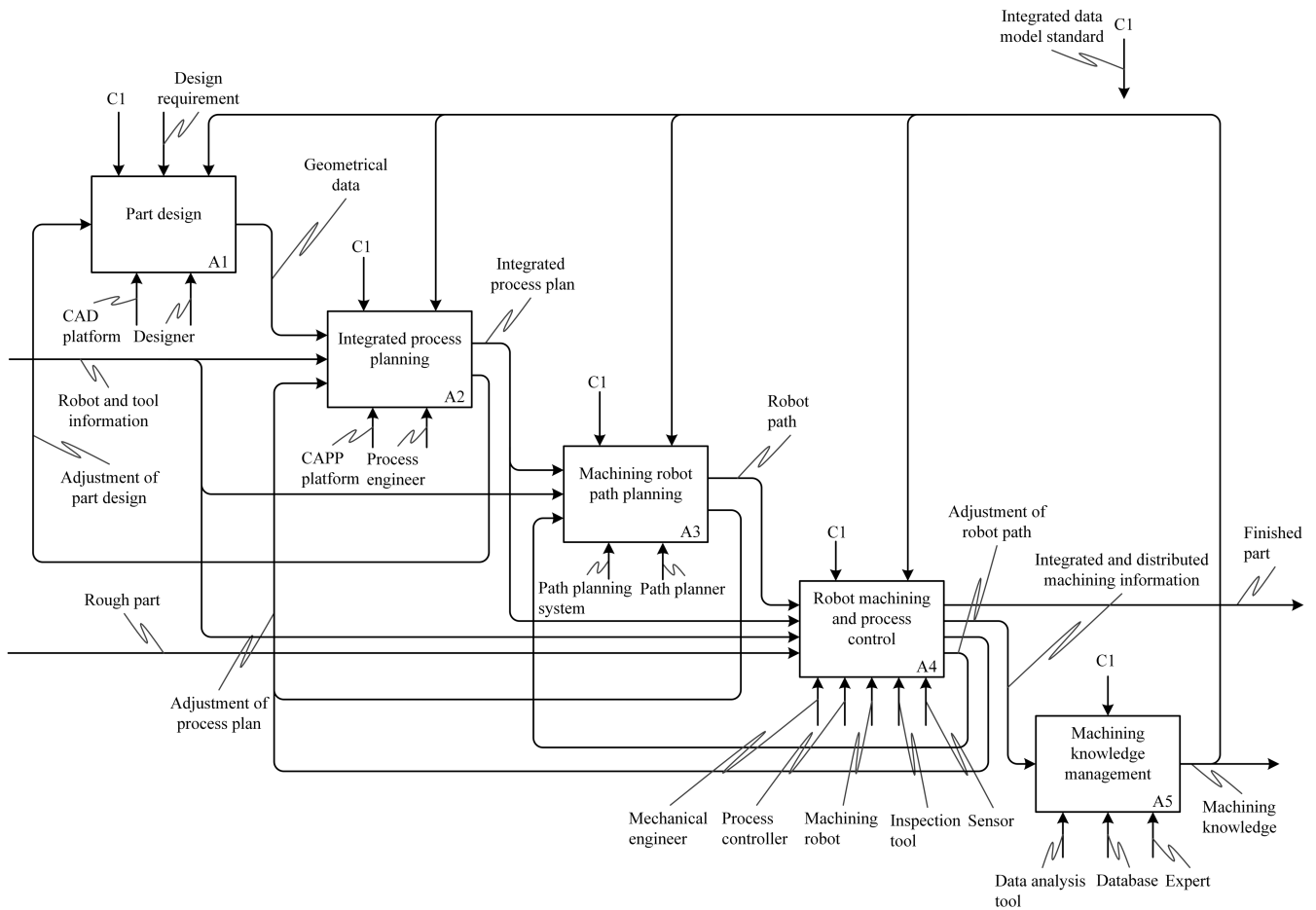
**FIGURE 3.** Functional activities and streams of CLRMS.

equipment, personnel, and other tools used in the robot machining process.

Functional activity A0 is further divided into five functional activities as shown in Fig. 3. They are part design (A1), integrated process planning (A2), machining robot path planning (A3), robot machining and process control (A4), and machining knowledge management (A5), corresponding to the five procedures of robot machining process. The five functional activities are connected with arrows representing streams of information or material used in these procedures. All information of robot machining process is collected and transferred in format of integrated data model standard to preserve and reuse machining knowledge. Every functional activity will analyze real-time information along with machining knowledge and adjust part design, process plan, robot path, and machining parameters adaptively.

### C. FUNCTIONAL ACTIVITIES IN CLRMS
#### 1) FUNCTIONAL ACTIVITY OF PART DESIGN
The function of activity A1 is to design the geometric parameters of part according to design requirement, usually conducted by designer on CAD platform. Machining knowledge

from activity A5 is considered as the part design will affect the manufacturability of parts. Output of this activity is geometrical model of both rough and finished part, which is represented in neutral format compatible with integrated data model standard. This model contained not only the 3D model such as STEP AP203 and STEP AP214 but also the material of part and tolerance of every feature. This will ensure the following activities to have comprehensive characteristic of the object to be machined. By this way activity A2 can make adjustment to geometrical data based on the actual situation in order to optimize the machining process. These adjustments will be inputted to activity A1 and manipulated by designer manually or by CAD platform software automatically.

#### 2) FUNCTIONAL ACTIVITY OF INTEGRATED PROCESS PLANNING
Activity A2 generates integrated process plan of robot machining contains machining task, online inspection, and machining status monitoring task. All information is represented with STEP-NC compliant data model and connected with geometrical data from activity A1. This procedure is divided into the following steps:

a) Identify and extract features to be machined and their tolerance items to be inspected by analyzing geometrical data of rough and finished parts.

b) Determine machining and inspection operations for every manufacturing feature on the basis of available robot and other equipment.

c) Set machining and inspection parameters for every operation according to the characteristic of robot and tool.

d) Select status parameters to be monitored or controlled during execution of machining operation.

e) Integrate all operations and features to create a machining, inspection, or monitoring workingstep.

f) Arrange the sequence of machining, inspection, and monitoring workingsteps.

g) Generate and output process plan of integrated robot machining tasks in format compatible with integrated data model standard.

Machining knowledge of previous machining processes are considered when generating new process plan. Adjustments such as process parameters, machining or inspection sequence, and add or remove of workingstep are made by activity A3 and A4 then sent back Activity A2.

### 3) FUNCTIONAL ACTIVITY OF MACHINING ROBOT PATH PLANNING

Process plan outputted from activity A2 is in STEP-NC data standard contains high-level information without tool path or motion control program. Then it is read and interpreted by STEP-NC file interpreter to plan robot path for controlling the robot. Activity A3 interprets STEP-NC file directly in online mode according to the real condition of machining robot and workpiece. The main function of activity A3 is divided into the following steps:

a) Read STEP-NC file and extract workingsteps from main workplan, then get machining, inspection, or monitoring information of every operation.

b) Generate machining robot path in work space according to the location of features and process routes of workingsteps.

c) Calculate machining robot path in joint space by inverse kinematic algorithm if the path contains complex curves.

d) Machining interference is checked according to the geometrical data of manufacturing features, workpiece, robot, and tool in STEP-NC file, along with the online positioning result in order to prevent collision during machining process.

e) Generate robot control program according to robot path if the robot controller only accepts programs in its own format.

### 4) FUNCTIONAL ACTIVITY OF ROBOT MACHINING AND PROCESS CONTROL

Function of activity A4 is executing robot machining process plan inputted from activity A2 adaptively. Machining status parameters, dimension error, form error, and position error are collected and attached with workingsteps and manufacturing features to generate the integrated and distributed machining information. Adjustment of machining process is

made by analyzing the machining information in real-time. The detailed function of activity A4 will be described in part D of section II.

### 5) FUNCTIONAL ACTIVITY OF MACHINING KNOWLEDGE MANAGEMENT

Function of activity A5 is storing, analyzing, and reusing all information collected during machining process, then generating and preserving machining knowledge, which contains both machining process information and its relationship. Artificial intelligence algorithms such as deep convolutional neural networks, Apriori algorithm, and genetic algorithm can be used in this procedure to obtain machining knowledge and to improve the machining quality [31], [32]. The machining knowledge is outputted to activity A1 to A4 as control streams to guide the machining process such as generating process plan, selecting machining parameters, and generating robot path.

### D. ROBOT MACHINING AND PROCESS CONTROL

Robot machining and process control (A4) is the core function of CLRMS. Its material input stream is rough part, which will be processed into finished part as output stream. During the machining process, activity A4 manipulates information from other activities to make adjustment of process plan, robot path, and process parameters in real-time according to machining status and inspection result, then sends the adjustment to other activities. It is divided into four collaborative functional activities as shown in figure 4, namely pre-machining inspection (A41), machining and monitoring (A42), online inspection (A43), and adjust machining process (A44).

Pre-machining inspection (A41) is needed when a rough part is installed on the fixture for the first time. Its purpose is to get the accurate relative position and orientation between machining robot and rough part. Activity A44 will adjusts the part to the position determined by process plan or recalculates the robot path and sends it back to activity A3. This procedure is necessary owing to machining robot is usually used in single and small batch production without fixture specially customized for the part, or in mobile machining of large-scale workpiece.

Function of activity A42 is getting robot path form activity A3 and executing the machining, inspection, and monitoring process determined in the integrated process plan. During machining, sensors are installed on robot, fixture or part to get machining status parameters such as force, torque, velocity, acceleration, and temperature. The collected data is outputted to other activities for further analysis.

An online inspection workingstep is added in machining process plan by activity A2 during machining process planning stage and conducted by activity A43 in the following situations:

1) The required precision of some manufacturing features is higher than the guaranteed precision of robot machining system.
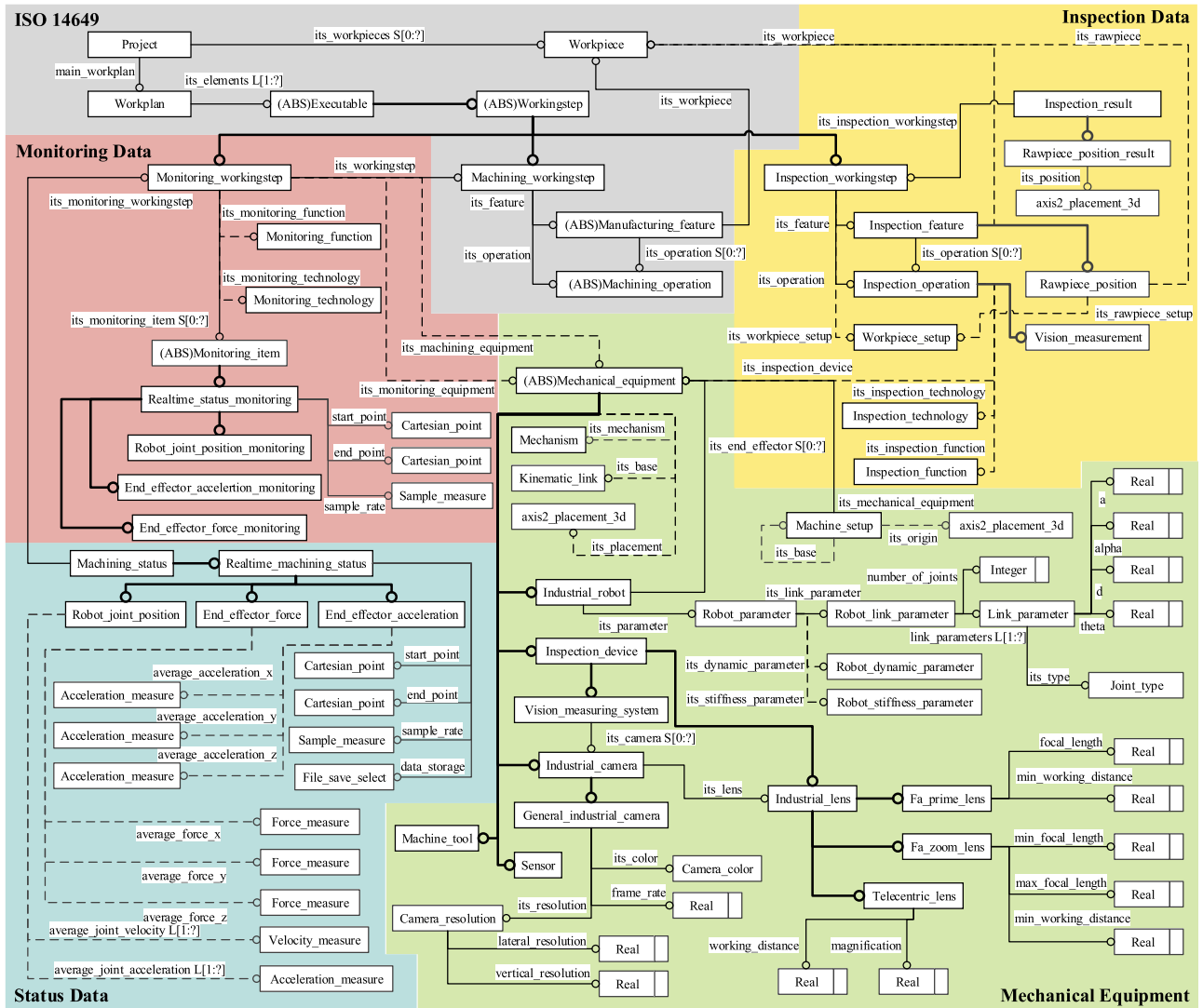
**FIGURE 4.** Functional activities and data streams of robot machining and process control.

2) The workpiece is re-installed during machining process so that the position of next manufacturing feature to be machined is changed.

3) The machining status parameters indicates that the error of previously finished manufacturing feature might be out of tolerance.

Inspection tasks of case 1) and 2) are scheduled to perform at some critical points of integrated process plan and written in STEP-NC file in advance. Inspection tasks of case 3) are written in STEP-NC file in forms of if statements and added to process plan in real-time by activity A43 if needed. Output streams of activity A43 are inspection results of both in-process and finished part.

Activity A44 has seven input streams from other activities covering all relevant information of machining process, such as integrated process plan, robot path, robot and tool information, inspection result of part, and machining status data. A process controller analyzes the information comprehensively to obtain the real condition of machining process and adjusts the machining process by optimizing cutting parameters or process plan in real-time. Take the procedure of cutting force control as an example to illustrate the function of activity A44 as follows.

1) Dynamometer is installed under the workpiece or on end effector of robot to monitor the cutting force during machining process. Multicomponent dynamometers that have been pre-encapsulated and calibrated by producer can be used directly, such as Kistler 9257B [33]. For self-designed cutting force measurement system, accurate calibration and layout planning of dynamometers should be conducted to assure the measurement accuracy [34].

2) When the cutting force is greater than the set value, which means deformation of robot and workpiece may be greater than the dimensional tolerance of the manufacturing feature. Then the process controller will adjust feedrate or cutting speed in real-time to reduce cutting force.

3) After the machining workingstep, A44 will send a request to A43 to add an online inspection workingstep to examine the actual error.

4) After the newly added inspection workingstep, A44 will send a request to A2 to add a new machining workingstep to fix the part if the error is repairable, or to abandon the part if the error cannot be fixed.

## III. STEP-NC DATA MODEL FOR CLOSED-LOOP ROBOT MACHINING

### A. INTEGRATED ROBOT MACHINING DATA MODEL BASED ON STEP-NC STANDARD

Data model integrating complete machining process information is the basis of closed-loop robot machining. In this section, the ARM of integrated robot machining data is built based on STEP-NC standard, which uses high-level information to describe machining process and supports bidirectional data transmission between any sections of the robot manufacturing system. Several extensions are made using the description method defined in STEP-NC standard to cover the whole closed-loop robot machining process. The overall data model structure contains both original STEP-NC standard and its extensions are illustrated in Fig. 5. All extensions are defined by EXPRESS and EXPRESS-G method and can be integrated seamlessly with other STEP-NC data model [35].

**FIGURE 5.** STEP-NC data model for closed-loop robot machining.

As discussed in section II, the information stream should contain the following types of information:

### 1) GEOMETRICAL DATA

In conventional manufacturing system, geometrical data means 3D model of finished part in universal data format such as STEP and IGES. This does not meet the requirement of CLRMS as the manufacturing features and their tolerances are ignored. In ISO-14649, geometrical data is represented as manufacturing features, such as planar_face, closed_pocket, round_hole, and so on, while the location, shape, dimension, and tolerance of manufacturing features are described by their properties [36]. In this context, data model in published STEP-NC standards are sufficient for CLRMS, so that this part of data model is directly referenced from ISO-14649 part 10 [36].

### 2) ROBOT MACHINING PROCESS PLAN

This part of data model is the core data of current STEP-NC standard, which describes machining process by using

entity project and its properties. The most significant property of project is its main_workplan representing by using entity workplan and its property its_elements, which contains one or several executables such as workingstep, nc_function, and program_structure. At present, most machining process information of milling and turning are already defined in ISO-14649 part 11 and 12, and can be directly referenced to build robot machining process plan data model [37], [38]. However, data model for machining status monitoring and online inspection are not defined in ISO-14649 or other compliant STEP standards. Therefore, a group of entities are newly defined and integrated with existing ISO-14649 data model.

### 3) ROBOT AND TOOL INFORMATION

This part of data model describes the characteristic of robot, tool, inspection equipment, sensor, and other accessories of industrial robot. ISO-14649 part 111 and 121 defines data model for milling cutting tools and turning cutting tools, which can be directly referenced [39], [40]. However, other

equipment used in CLRMS are not included in STEP-NC standard. In that case, data model of industrial robot, inspection device, industrial camera, machine tool, and sensor are defined and integrated with existing ISO-14649 data model.

### 4) ROBOT PATH

In CLRMS, robot path generated by path planning system is not only used to control the robot, but also used to calculate the posture and speed of robot according predict potential problems such as deformation, excessive joint velocity and singularity. So, it is necessary for robot machining process controller to read and store robot path especially position and velocity of every joint in STEP-NC files. Therefore, entity robot_joint_positon is defined and integrated in STEP-NC data model.

### 5) REAL-TIME MACHINING DATA

Real-time machining data such as cutting force, cutting speed, federate, and cutting accelerate are collected during robot machining process. As introduced above, position, velocity, and acceleration of joints contained in entity robot_joint_positon are also included, along with joint torque and force. Therefore, entities for representing these data are defined and integrated in STEP-NC data model.

### B. STEP-NC DATA MODEL FOR ONLINE INSPECTION AND REAL-TIME MONITORING

Inspection and monitoring are necessary to improve the quality of finished parts and efficiency of machining process in robot machining owing to low stiffness and accuracy of robot. Three types of workingstep are defined in ISO-14649 part 10 namely machining_workingstep, rapid_movement and touch_probing [36]. However, entity touch_probing describes process of workpiece probing, workpiece complete probing and, tool probing by low-level information without inspection result. Furthermore, there is no data model for real-time robot machining status monitoring. In that case, two types of workingstep namely inspection_workingstep and monitoring_workingstep are newly defined by using EXPRESS method, along with entity inspection_result, which is used to preserve results of inspection_workingstep.

Entity inspection_workingstep contains information of online inspection, such as inspection feature, operation and position of workpiece. The EXPRESS definition is shown in Fig. 6. Entities related to inspection workingstep are also defined as shown in Fig. 7. Entity inspection_feature describes what to be inspected, while inspection_operation describes its inspection method. The results are recorded in entity inspection_result, which is linked to inspection workingstep. Entity rawpiece_position derived from inspection_feature, entity vision_measurement derived from inspection_operation, and entity rawpiece_position_result derived from inspection_result are defined to describe workpiece positioning process using machine vision method.

Entity monitoring_workingstep describes the real-time monitoring process during robot machining, including the



```
ENTITY inspection_workingstep
   SUBTYPE OF workingstep;
   its_feature:              inspection_feature;
   its_operation:            inspection_operation;
   its_workpiece_setup:      OPTIONAL workpiece_setup;
END_ENTITY;
```

**FIGURE 6.** Definition of inspection_workingstep.



```
ENTITY inspection_feature
   its_id:            identifier;
   its_workpiece:     OPTIONAL workpiece;
   its_operations:    SET [0:?] OF inspection_operation;
END_ENTITY;

ENTITY rawpiece_position
   SUBTYPE OF inspection_feature;
   its_rawpiece:        workpiece;
   its_rawpiece_setup:  OPTIONAL workpiece_setup;
END_ENTITY;

ENTITY inspection_operation
   its_id:                    identifier;
   its_inspection_device:     OPTIONAL mechanical_equipment;
   its_inspection_technology: OPTIONAL inspection_technology;
   its_inspection_functions:  OPTIONAL inspection_functions;
END_ENTITY;

ENTITY vision_measurement
   SUBTYPE OF inspection_operation;
END_ENTITY;

ENTITY inspection_result
   its_inspection_workingstep: inspection_workingstep;
END_ENTITY;

ENTITY rawpiece_position_result
   SUBTYPE OF inspection_result;
   its_position:  axis2_placement_3d;
END_ENTITY;
```

**FIGURE 7.** Definition of online inspection.



```
ENTITY monitoring_workingstep
   SUBTYPE OF workingstep;
   its_workingstep:           workingstep;
   its_machining_equipment:   OPTIONAL mechanical_equipment;
   its_monitoring_equipment:  OPTIONAL mechanical_equipment;
   its_monitoring_item:       SET [1:?] OF monitoring_item;
   its_monitoring_technology: OPTIONAL monitoring_technology;
   its_monitoring_functions:  OPTIONAL monitoring_functions;
END_ENTITY;
```

**FIGURE 8.** Definition of monitoring_workingstep.

corresponding machining workingstep, items to be monitored, equipment, technology, and functions. The EXPRESS definition is shown in Fig. 8. Entity monitoring_item is an abstract super type of realtime_status_monitoring, more items would be defined base on this super type in the future. In this paper, three real-time monitoring items as shown in Fig. 9 are defined as subtype of realtime_status_monitoring, namely robot_joint_position_monitoring, end_effector_acceleration_monitoring, and end_effector_force_monitoring.

### C. STEP-NC DATA MODEL FOR MECHANICAL EQUIPMENT IN CLRMS

Process control of robot machining system is more complex than machine tool, so that an abstract super type mechanical_equipment and several subtypes are defined to describe

```
ENTITY monitoring_item
   ABSTRACT SUPERTYPE OF (ONEOF
(realtime_status_monitoring));
END_ENTITY;

ENTITY realtime_status_monitoring
   SUBTYPE OF monitoring_item;
   start_point:        OPTIONAL cartesian_point;
   end_point:          OPTIONAL cartesian_point;
   sample_rate:        OPTIONAL sample_measure;
END_ENTITY;

ENTITY robot_joint_position_monitoring
   SUBTYPE OF realtime_status_monitoring;
   monitor_joint:       LIST [1:?] OF BOOLEAN;
END_ENTITY;

ENTITY end_effector_accelertion_monitoring
   SUBTYPE OF realtime_status_monitoring;
   monitor_accelertion_x:       OPTIONAL BOOLEAN;
   monitor_accelertion_y:       OPTIONAL BOOLEAN;
   monitor_accelertion_z:       OPTIONAL BOOLEAN;
END_ENTITY;

ENTITY end_effector_force_monitoring
   SUBTYPE OF realtime_status_monitoring;
   monitor_force_x:     OPTIONAL BOOLEAN;
   monitor_force_y:     OPTIONAL BOOLEAN;
   monitor_force_z:     OPTIONAL BOOLEAN;
END_ENTITY;
```

**FIGURE 9.** Definition of monitoring_item.

```
ENTITY mechanical_equipment
   ABSTRACT SUPERTYPE OF (ONEOF (machine_tool,
industrial_robot, inspection_device, sensor));
   its_id:            label;
   its_vendor_id:     label;
   its_product_id:    label;
   its_mechanism:     OPTIONAL mechanism;
   its_base:          OPTIONAL kinematic_link;
   its_placement:     OPTIONAL axis2_placement_3d;
END_ENTITY;
```

**FIGURE 10.** Definition of mechanical_equipment.

information of various types of equipment, which should be considered in both process planning stage and machining stage. The EXPRESS definition is shown in Fig. 10. If the equipment is installed separately, property its_base should not be defined. If the equipment is installed on another equipment, property its_base should be defined. At the meantime, property its_placement should also be defined to describe the position of the equipment in coordinate system of kinematic_link defined by property its_base.

A four-level hierarchical coordinate systems namely machine coordinate system (MCS), setup coordinate system (SCS), workpiece coordinate system (WCS), and feature coordinate system (FCS) are defined in ISO-14649 Part10 to determine the location of workpiece and manufacturing feature [36]. In this system, fixture, workpiece, and cutting tool are usually installed on the table of machine tool so that MCS can be treated as base coordinate system. But machining robot has no fixed table while the robot itself can be mobile. Furthermore, other equipment such as inspection devices and sensors may not be installed on the robot structure. In this

context, the four-level hierarchical coordinate systems in ISO-14649 are insufficient to describe the relative position of equipment in CLRMS. In this paper, a multi branch frame of coordinate systems for robot machining are designed as show in Fig. 11. The Setup-Workpiece-Feature chain coordinate systems is no longer described in MCS but in a newly defined global coordinate system (GCS), in which machine coordinate system (MCS) and its link coordinate systems (LCS) of more than one equipment can be defined in separate chains. MCS of the equipment can also be defined in MCS or LCS of another equipment on which it is installed. For example, the MCS of end effector is usually defined in LCS of the last joint of industrial robot.

Entity machine_setup which describes MCS of mechanical equipment is defined as shown in Fig. 12. Property mechanical_equipment describes which equipment the MCS is related to. Property its_origin defined the MCS of mechanical equipment in GCS or in MCS of another mechanical equipment, depends on the related mechanical_equipment is installed separately or on another equipment. This entity should not be used if the mechanical equipment is installed on a movable part of another equipment.

Entity mechanism and entity kinematic_link are referenced from ISO-10303 Part 105 to describe kinematic structure of machining robot [33]. However, these two entities cannot describe D-H parameters of machining robot. In that case, new data model of industrial robot integrating D-H parameters, dynamic parameters, and stiffness parameters is defined as shown in Fig. 13. Entity link_parameter describes D-H parameters of industrial robot using the first four properties to represent distance and rotation angle between adjacent joints. Property its_type represents type of joint, which can be revolute or prismatic.

Data model for machine vision system is defined as shown in Fig. 14. Entity inspection_device is super type of all inspection devices such as vision measuring system, industrial camera, and lens. Entity vision_measuring_system has one or more industrial cameras is used to perform vision measurement tasks in robot machining. Entity industrial_camera and industrial_lens and their sub types are used to describe parameters of camera and lens in vision measuring system.

### D. STEP-NC DATA MODEL FOR ROBOT MACHINING STATUS

Entity machining_status and its subtype realtime_machining_status is defined for recording real-time machining status data in robot machining process. The EXPRESS definition is shown in Fig. 15. Its property linked machining status data with corresponding monitoring workingstep. Time domain data of machining status is saved as separate data file owing to its large data volume. Property data_storage indicates the storage location of data file. Only characteristic values of real-time machining status data such as average force, average acceleration, and average velocity are saved in STEP-NC file. Three entities are defined for recording force and
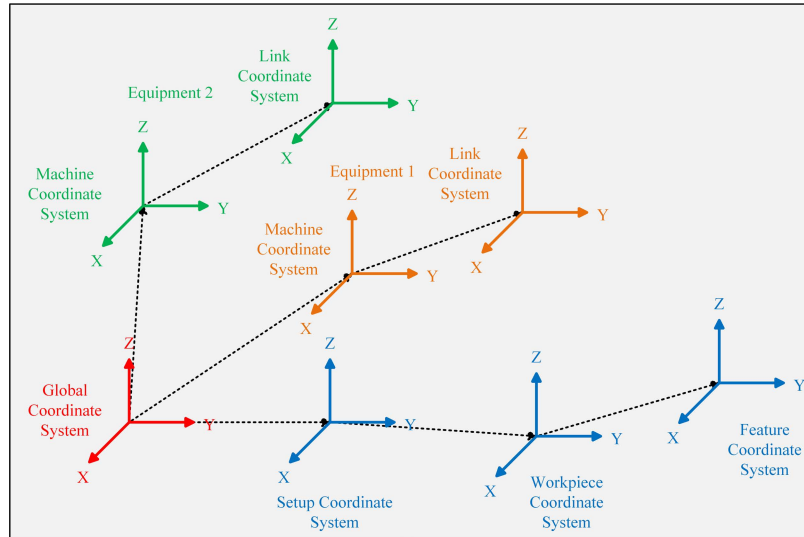
**FIGURE 11. Coordinate systems in CLRMS.**

```
ENTITY machine_setup
    its_mechanical_equipment:  mechanical_equipment;
    its_origin:           OPTIONAL axis2_placement_3d;
    its_base:             OPTIONAL machine_setup;
END_ENTITY;
```

**FIGURE 12. Definition of machine_setup.**

```
ENTITY industrial_robot
    SUBTYPE OF mechanical_equipment;
    its_parameter:        robot_parameter;
    its_end_effector:     SET [0:?] OF mechanical_equipment;
END_ENTITY;

ENTITY robot_parameter
    its_link_parameter:              robot_link_parameter;
    its_dynamic_parameter:           OPTIONAL
                                     robot_dynamic_parameter;
    its_stiffness_parameter:         OPTIONAL
                                     robot_stiffness_parameter;
END_ENTITY;

ENTITY robot_link_parameter
    number_of_joints:   INTEGER;
    link_parameters:    LIST [1:?] OF link_parameter;
END_ENTITY;

ENTITY link_parameter
    a:                  REAL;
    alpha:              REAL;
    d:                  REAL;
    theta:              REAL;
    its_type:           joint_type;
END_ENTITY;

TYPE joint_type = ENUMERATION OF (revolute, prismatic);
END_TYPE;
```

**FIGURE 13. Definition of industrial robot and its parameters.**

```
ENTITY inspection_device
    SUBTYPE OF mechanical_equipment;
END_ENTITY;

ENTITY vision_measuring_system
    SUBTYPE OF inspection_device;
    its_camera:    SET [1:?] OF industrial_camera;
END_ENTITY;

ENTITY industrial_camera
    SUBTYPE OF inspection_device;
    its_lens:  industrial_lens;
END_ENTITY;

ENTITY general_industrial_camera
    SUBTYPE OF industrial_camera;
    its_color:          camera_color;
    its_resolution:     camera_resolution;
    frame_rate:         REAL;
END_ENTITY;

ENTITY camera_resolution
    lateral_resolution:   REAL
    vertical_resolution:  REAL;
END_ENTITY;

ENTITY industrial_lens
    SUBTYPE OF inspection_device;
END_ENTITY;
```

**FIGURE 14. Definition of inspection device.**

acceleration of robot end effector, along with position of joints in real-time during machining process.

## IV. OPEN STEP-NC INTERPRETER AND ONLINE INSPECTION

### A. OPEN STEP-NC INTERPRETER FOR CLRMS

The overall architecture of the proposed CLRMS is presented in Fig. 16. Software platform consist of CAD software platform, CAPP software platform, open STEP-NC interpreter, and machining knowledge management system. Hardware platform consist of machining robot, machine vision system and hardware interface. Programming interface connects software with hardware. CAD software platform corresponds to functional activity A1 in section II generates geometrical model of machined part and transfers it to CAPP software platform. Any CAD software with function of geometry modeling can be used in CLRMS to accomplish this task. CAPP software platform corresponds to functional activity A2 in section II generates machining process plan integrates geometrical information of machined part, mechanical equipment in robot machining system, machining workingstep, inspection workingstep, and monitoring workingstep

```
ENTITY machining_status
    its_monitoring_workingstep:    monitoring_workingstep;
END_ENTITY;

ENTITY realtime_machining_status
    SUBTYPE OF machining_status;
    start_point:          cartesian_point;
    end_point:            cartesian_point;
    sample_rate:          sample_measure;
    data_storage:         file_save_select;
END_ENTITY;

ENTITY end_effector_force
    SUBTYPE OF realtime_machining_status;
    average_force_x:   OPTIONAL force_measure;
    average_force_y:   OPTIONAL force_measure;
    average_force_z:   OPTIONAL force_measure;
END_ENTITY;

ENTITY end_effector_acceleration
    SUBTYPE OF realtime_machining_status;
    average_acceleration_x:        OPTIONAL
                                   acceleration_measure;
    average_acceleration_y:        OPTIONAL
                                   acceleration_measure;
    average_acceleration_z:        OPTIONAL
                                   acceleration_measure;
END_ENTITY;

ENTITY robot_joint_position
    SUBTYPE OF realtime_machining_status;
    average_joint_velocity:        OPTIONAL LIST [1:?]
                                   OF velocity_measure
    average_joint_acceleration:    OPTIONAL LIST [1:?]
                                   OF acceleration_measure;
END_ENTITY;
```

**FIGURE 15.** Definition of machining status.

in a unified data stream by using data model described in section III. Machining knowledge management system is a database storing and analyzing the information of robot machining process.

The implementation work of this paper is focused on activity A3 and A4 described in section II. Activity A3 can be implemented as an individual system or integrated with activity A4. In this paper, an open STEP-NC interpreter integrating activity A3 and A4 is proposed and developed for robot machining to form a high-level machining robot controller. The proposed STEP-NC interpreter can read the STEP-NC file and control the machining process adaptively. The software kernel of the proposed open STEP-NC interpreter consists of five software modules as described below.

1) Human machine interface (HMI) module is an input and output interface. Its main function is to read and edit STEP-NC file, display vision captured by camera and other information of robot machining process.

2) STEP-NC interpreting module is the most important part of open STEP-NC interpreter. Core functions such as parsing STEP-NC file and mapping it to internal data format, checking syntax and logic errors, and generating tool path are conducted by this module.

3) Robot path planning module deals with kinematics calculation and velocity planning problems to generate robot path in Cartesian space or joint space. This module has two types of implementation mode, namely integrated mode and standalone mode. Integrated mode means the STEP-NC interpreter is integrated with an open robot controller with software interfaces and can control the servo drivers directly. Standalone mode means the robot controller is a closed-system that can only accept program of native language. A postprocessor specially developed for the robot controller is needed if this module works in the latter mode.

4) Online inspection module communicates with machine vision system through hardware interface such as USB port and Ethernet port to capture image of workpiece. Then the images are processed to measure the shape, dimension, position, and surface roughness. Inspection results are saved in STEP-NC file and analyzed for adjusting robot machining process.

5) Real-time monitoring module communicates with sensors through hardware interface such as USB port, serial port, and data acquisition card to record real-time robot machining status data. The collected data is analyzed by using artificial intelligence algorithms to determine the robot machining condition and adjust machining parameters adaptively in real-time to get better machining quality.

## B. ONLINE WORKPIECE POSITIONING BASED ON MACHINE VISION

The proposed STEP-NC interpreter can generate tool path in FCS of every manufacturing feature. Then the robot path is generated based on tool path in FCS and relative position and orientation between coordinate systems in robot machining system. However, the location of workpiece is usually not fixed in robot machining, so that online workpiece positioning is needed when a new workpiece is to be machined. In this paper, machine vision method is used to get the relative position between workpiece and machining robot.

The coordinate systems of CLRMS in this paper is shown in Fig. 17. A rectangle shape workpiece clamped on a flat plier which is parallel to the base of machining robot is took as an example to illustrate the process of online workpiece positioning. The top left corner, bottom left corner, and bottom right corner of the workpiece, namely P1, P2, and P3 are selected as target points for online workpiece positioning.

There are two branches of coordinate systems in the implementation platform of this paper. The first branch is machining robot and its end effectors. The second branch is fixture and workpiece. In the first branch, MSC of machining robot $\{M_R\}$ is coincide with GCS $\{G\}$. LCS of 6 joints and ending tool $\{L_T\}$ are linked in series. Industrial camera for online workpiece positioning and motorized spindle with cutting tool for machining are installed on the ending tool of robot. Therefore, MCS of both mechanical equipment $\{M_C\}$ and $\{M_M\}$ are defined in $\{L_T\}$. In the second branch, SCS $\{S\}$ is set at the fixture on platform and WCS $\{W\}$ is coincide with $\{S\}$. FCS $\{F\}$ of every manufacturing feature is defined
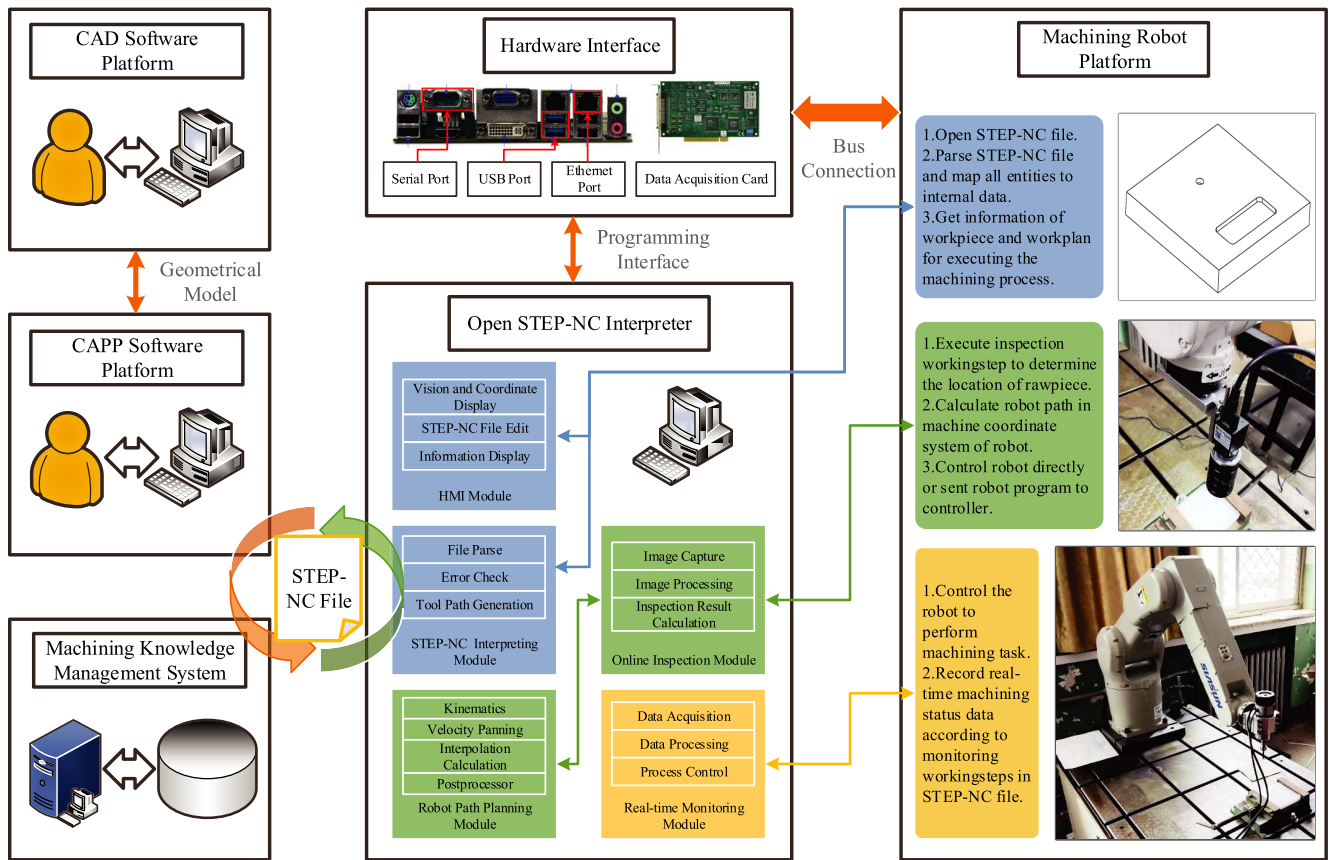
**FIGURE 16.** Architecture of closed-loop robot machining system based on STEP-NC.
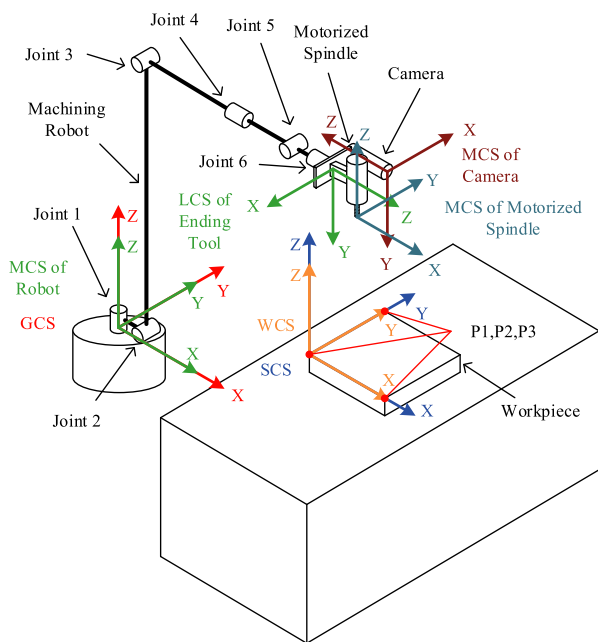


**FIGURE 17.** Coordinate systems of machining robot and workpiece.

in {W} by entities in STEP-NC file. The relative location and orientation between {S} and {G} should be determined before generating robot path.

The industrial camera (MV-SUA2000M-T, MindVision) and lens (MV-LD-16-10M-J, MindVision) are firstly calibrated using the Zhang's method to get the intrinsic parameters [42]. The camera is black-and-white with the resolution of 5488×3672. In total, 20 images of a checkerboard with 12×9 squares from different angles are captured for calibration. Part of the images are shown in Fig. 18. The calibration results are

$$f_x = 7023.0002, f_y = 7023.2098, u_0 = 2757.6177,$$
$$v_0 = = 1767.1936, k_1 = -0.07221, k_2 = 0.04745,$$
$$p_1 = -7.8641 \times 10^{-4}, p_2 = -4.9523 \times 10^{-4}.$$

The method of online workpiece positioning is shown in Fig. 19. Compared with deep learning method that needs training of a large amount of samples, template matching algorithm has low computational complexity and good robustness and is widely used in feature recognition and localization [43]. The characteristics of rectangle shape workpiece used in this paper is clear and distinguishable. Furthermore, the shape and dimension of workpiece and the machining environment is usually known in advance, and the axis of camera is perpendicular to the upper planner of workpiece during positioning process, which makes the characteristic more markedly and leads to more accurate measurement results. In this context, it is practicable to make a template
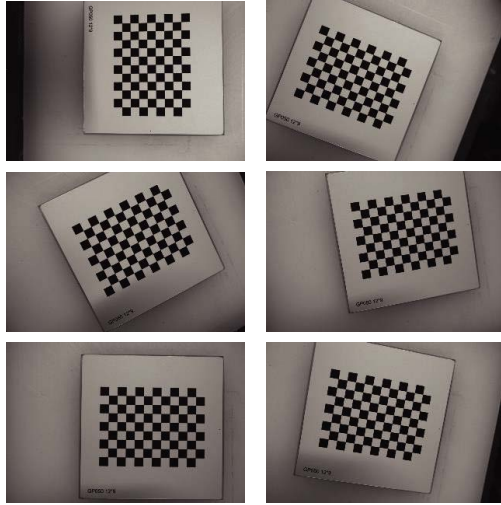
**FIGURE 18.** Part of images used for camera calibration.

**TABLE 1.** Recognition results of template matching algorithm.

|  | P1 (100 images) | P2 (100 images) | P3 (100 images) |
|---|---|---|---|
| **TPL1** | 100 matched | 0 matched | 0 matched |
| **TPL2** | 0 matched | 100 matched | 0 matched |
| **TPL3** | 0 matched | 0 matched | 100 matched |

that is accurately matched with the three target points in Fig. 17 and use template matching algorithm to get the location of feature-point correctly [44].

The template matching algorithm is carried out as follows.

1) Make three templates for P1, P2, P3 respectively, namely TPL1, TPL2, TPL3, as bitmap files in advance.

2) Capture the image of one target point.

3) Transformed the image to gray-scale image.

4) Transformed the gray-scale image to binary image.

5) Use the pre-made template to recognize and estimate the X and Y coordinate values of target point in the image.

To validate the validity of the template matching algorithm, 300 images of the three target points are captured, 100 for each. Then use the template matching algorithm to recognize the target point in every image. Normalized squared difference method is used to find the optimum matching position, and the threshold value is set to less than 0.2. The recognition results are shown in Table 1, which indicate that the recognition rate is 100% when using template correspond to the target point for recognition, and no target point is recognized when using template correspond to another target point for recognition. In this context, template matching algorithm is suitable for online positioning of simple rectangle shape workpiece.

The procedure of online workpiece positioning is as follows.

1) Install the workpiece parallel to the base platform of machining robot. Then move the robot to make {$M_C$} parallel with {G}. In that case, the axis of camera is perpendicular to

the base platform. Adjust the camera in Z direction of {G} by moving the robot until it is at appropriate height to take the picture of one corner point of the workpiece clearly and completely. Estimate the pixel distance by use of calibration checkerboard mentioned above, and the result is 0.02893mm per pixel.

2) Move the camera above the three target points sequentially. At every target point, capture the image and calculated the offset values between target point and the center of image using template matching algorithm. Move the robot according to the offset values until the two points are aligned in the image coordinate. The alignment error is ±1 pixel due to the movement precision of robot, which is 0.05786mm in positioning error.

3) Records the rotation angle of robot joints and calculate the position of camera axis in XY plane using forward kinematics solution of industrial robot.

## C. ROBOT PATH GENERATION

As described in part B of section IV, the target point coincides with the camera axis at XY plane of {$M_C$} in the workpiece positioning procedure. In that case, target point position in {$M_C$} is represented as

$$^{M_C}p_{tp} = \begin{bmatrix} 0 & 0 & ^{M_C}z_{tp} & 1 \end{bmatrix}^T \tag{1}$$

and in {W} is represented as

$$^{W}p_{tp} = \begin{bmatrix} ^{W}x_{tp} & ^{W}y_{tp} & ^{W}z_{tp} & 1 \end{bmatrix}^T \tag{2}$$

The target point position is transformed to {G} in two branches of coordinate systems as

$$^{G}_{M_R}T\,^{M_R}_{L_T}T\,^{L_T}_{M_C}T\,^{M_C}p_{tp} = ^{G}_{S}T\,^{S}_{W}T\,^{W}p_{tp} \tag{3}$$

where $^{G}_{M_R}T$ is transformation matrix from {$M_R$} to {G}; $^{M_R}_{L_T}T$ is transformation matrix from {$L_T$} to {$M_R$}; $^{L_T}_{M_C}T$ is transformation matrix from {$M_C$} to {$L_T$}; $^{G}_{S}T$ is transformation matrix from {S} to {G}; $^{S}_{W}T$ is transformation matrix from {W} to {S}.

As described above, {$M_R$} coincide with {G}, and {W} coincide with {S}, so that (3) can be reduced to

$$^{M_R}_{L_T}T\,^{L_T}_{M_C}T\,^{M_C}p_{tp} = ^{G}_{S}T\,^{W}p_{tp} \tag{4}$$

According to the relative position and orientation between {$M_C$} and {$L_T$} in Fig. 17, $^{L_T}_{M_C}T$ can be represented as

$$^{L_T}_{M_C}T = \begin{bmatrix} -1 & 0 & 0 & a_{M_C} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & d_{M_C} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

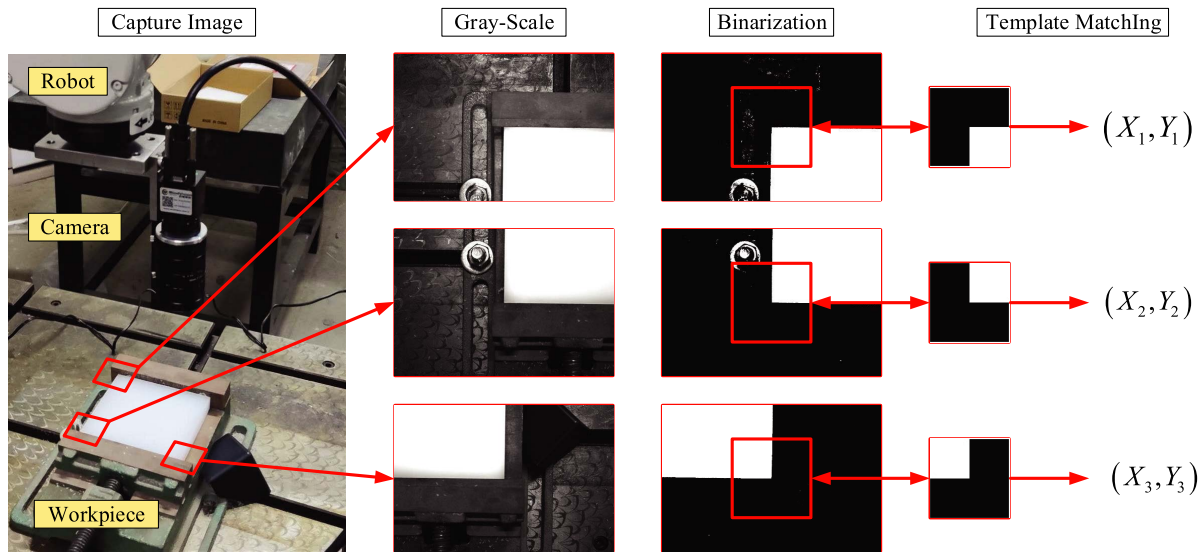where $a_{M_C}$ and $d_{M_C}$ are the X and Z coordinates of {$M_C$} origin in {$L_T$}.

**FIGURE 19.** The framework of online workpiece positioning method.

In vision measurement process, $\{M_C\}$ and $\{M_R\}$ are at the same orientation, so that $^{M_R}_{L_T}T$ can be represented as

$$^{M_R}_{L_T}T = \begin{bmatrix} -1 & 0 & 0 & ^Gx_{T_0} \\ 0 & 1 & 0 & ^Gy_{T_0} \\ 0 & 0 & -1 & ^Gz_{T_0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Axis $Z_S$ is parallel with axis $Z_G$ and $^G_ST$ can be represented as

$$^G_ST = \begin{bmatrix} \cos\theta_S & -\sin\theta_S & 0 & ^Gx_{S_0} \\ \sin\theta_S & \cos\theta_S & 0 & ^Gy_{S_0} \\ 0 & 0 & 1 & ^Gz_{S_0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

where $\theta_S$ is rotational angle from $\{G\}$ to $\{S\}$ around axis $Z_G$.

According to (1)-(7), (3) can be represented as

$$\begin{bmatrix} 1 & 0 & 0 & ^Gx_{T_0} - a_{M_C} \\ 0 & 1 & 0 & ^Gy_{T_0} \\ 0 & 0 & 1 & ^Gz_{T_0} - d_{M_C} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ ^{M_C}z_{tp} \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta_S & -\sin\theta_S & 0 & ^Gx_{S_0} \\ \sin\theta_S & \cos\theta_S & 0 & ^Gy_{S_0} \\ 0 & 0 & 1 & ^Gz_{S_0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} ^Wx_{tp} \\ ^Wy_{tp} \\ ^Wz_{tp} \\ 1 \end{bmatrix} \quad (8)$$

Take the first two rows of (8) as

$$\begin{cases} ^Wx_{tp}\cos\theta_S - ^Wy_{tp}\sin\theta_S + ^Gx_{S_0} = ^{M_R}x_{T_0} - a_{M_c} \\ ^Wx_{tp}\sin\theta_S + ^Wy_{tp}\cos\theta_S + ^Gy_{S_0} = ^{M_R}y_{T_0} \end{cases} \quad (9)$$

The coordinates of the three target points in $\{W\}$ are

$$\left(0, ^Wy_{tp1}, 0\right), (0, 0, 0), \left(^Wx_{tp3}, 0, 0\right).$$

The coordinates of $\{L_T\}$ origin in $\{M_R\}$ when camera is aligned with target point can be calculated using D-H method and represented as

$$\left(^{M_R}x_{T_01}, ^{M_R}y_{T_01}, ^{M_R}z_{T_01}\right),$$
$$\left(^{M_R}x_{T_02}, ^{M_R}y_{T_02}, ^{M_R}z_{T_02}\right),$$
$$\left(^{M_R}x_{T_03}, ^{M_R}y_{T_03}, ^{M_R}z_{T_03}\right).$$

Put the three groups of coordinates into (9) and get

$$\begin{cases} \theta_{S1} = \arctan 2(^{M_R}x_{T_02} - ^{M_R}x_{T_01}, ^{M_R}y_{T_01} - ^{M_R}y_{T_02}) \\ \theta_{S2} = \arctan 2(^{M_R}y_{T_03} - ^{M_R}y_{T_02}, ^{M_R}x_{T_03} - ^{M_R}x_{T_02}) \\ ^Gx_{S_0} = ^{M_R}x_{T_02} - a_{M_C} \\ ^Gy_{S_0} = ^{M_R}y_{T_02} \\ \theta_S = (\theta_{S1} + \theta_{S2})/2 \end{cases} \quad (10)$$

The cutter tip is coinciding with $\{M_M\}$ origin so its coordinate in $\{M_M\}$ is

$$^{M_M}p_{ct} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T \quad (11)$$

and in $\{W\}$ is represented as

$$^Wp_{ct} = \begin{bmatrix} ^Wx_{ct} & ^Wy_{ct} & ^Wz_{ct} & 1 \end{bmatrix}^T \quad (12)$$

The coordinates of cutter tip can be transformed to $\{G\}$ in two branches of coordinate systems as

$$^G_{M_R}T\,^{M_R}_{L_T}T\,^{L_T}_{M_M}T\,^{M_M}p_{ct} = ^G_ST\,^S_WT\,^Wp_{ct} \quad (13)$$

and reduced to

$$\begin{bmatrix} 1 & 0 & 0 & ^{M_R}x_{T_0} + a_{M_M} \\ 0 & 1 & 0 & ^{M_R}y_{T_0} \\ 0 & 0 & 1 & ^{M_R}z_{T_0} - d_{M_M} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$
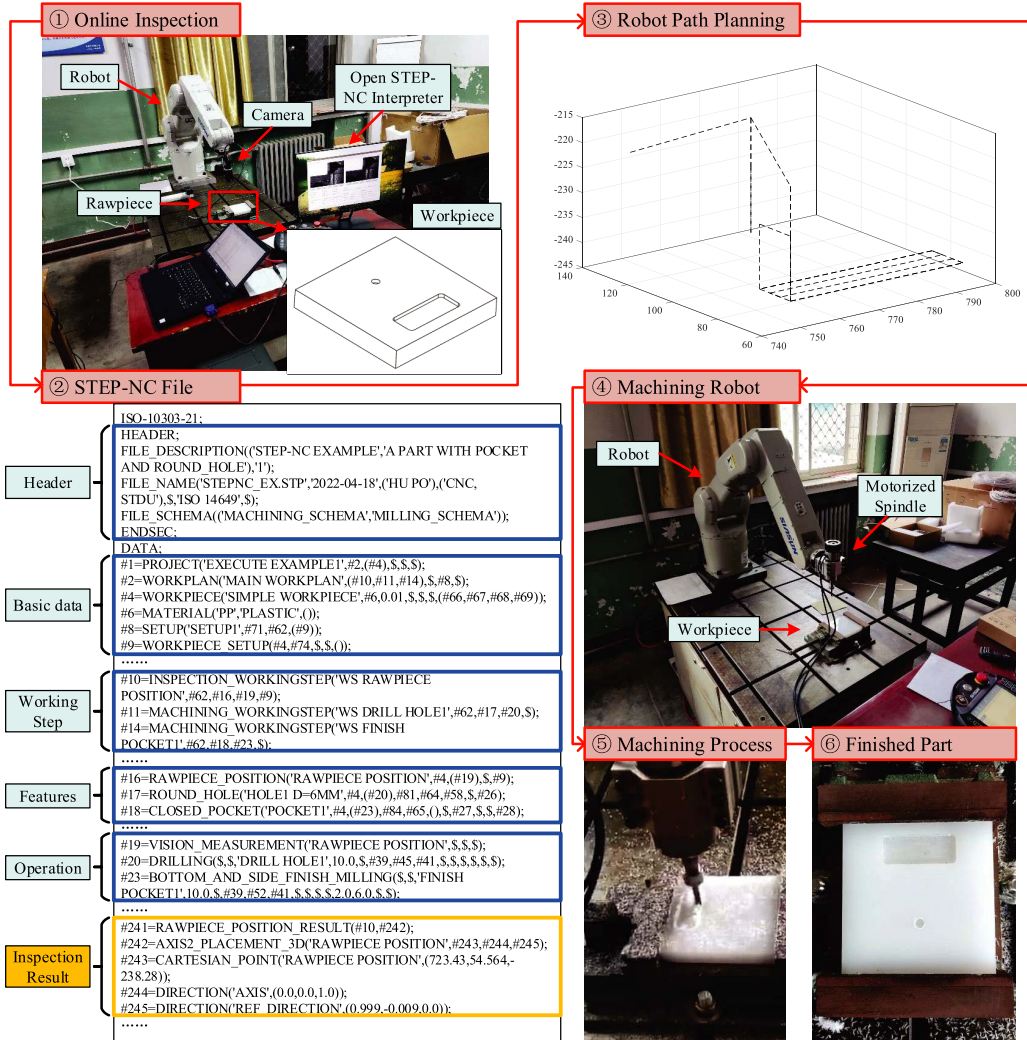
**FIGURE 20.** Implementation of online inspection and machining in CLRMS.

$$
= \begin{bmatrix} \cos\theta_S & -\sin\theta_S & 0 & {}^G x_{S_0} \\ \sin\theta_S & \cos\theta_S & 0 & {}^G y_{S_0} \\ 0 & 0 & 1 & {}^G z_{S_0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^W x_{ct} \\ {}^W y_{ct} \\ {}^W z_{ct} \\ 1 \end{bmatrix} \quad (14)
$$

where $d_{M_M}$ and $a_{M_M}$ are the Y and Z coordinates of $\{M_M\}$ origin in $\{L_T\}$.

Take the first three rows of (14) as

$$
\begin{cases} {}^{M_R}x_{T_0} = {}^W x_{ct}\cos\theta_S - {}^W y_{ct}\sin\theta_S + {}^G x_{S_0} - a_{M_M} \\ {}^{M_R}y_{T_0} = {}^W x_{ct}\sin\theta_S + {}^W y_{ct}\cos\theta_S + {}^G y_{S_0} \\ {}^{M_R}z_{T_0} = {}^W z_{ct} + {}^G z_{S_0} + d_{M_M} \end{cases} \quad (15)
$$

Then let the bottom of cutter touch the upper surface of workpiece, on which the Z coordinate of $\{W\}$ is zero. Then read robot joint position and calculate the Z coordinate of $\{L_T\}$ origin in $\{M_R\}$ by using D-H method, which is represented as ${}^{M_R}z_{T_04}$.

Put ${}^{M_R}z_{T_04}$ into (15) and get

$$
\begin{cases} {}^{M_R}x_{T_0} = {}^W x_{ct}\cos\theta_S - {}^W y_{ct}\sin\theta_S + {}^G x_{S_0} - a_{M_M} \\ {}^{M_R}y_{T_0} = {}^W x_{ct}\sin\theta_S + {}^W y_{ct}\cos\theta_S + {}^G y_{S_0} \\ {}^{M_R}z_{T_0} = {}^W z_{ct} + {}^{M_R}z_{T_04} \end{cases} \quad (16)
$$

Tool paths for approaching, cutting, and retracting every manufacturing feature are planned and generated in FCS. Transformation matrix from $\{F\}$ to $\{W\}$, namely ${}^W_F T$ is calculated by reading properties from entities in STEP-NC file. The tool path is firstly transformed to $\{W\}$ by ${}^W_F T$, then use (16) to calculate the robot path.

## V. IMPLEMENTATION AND VALIDATION

The proposed STEP-NC compliant closed-loop robot machining system and methodology is implemented and validated as shown in Fig. 20. The experimental platform consists of SIASUN SR7CL industrial robot, industrial camera, industrial lens, ER11-48V motorized spindle with $\phi$6mm

end milling cutter, and Latitude 3470 laptop with Intel Core i5-6200U CPU and NVIDIA Geforce 920M graphics card. The open STEP-NC interpreter is developed on QT and Visual Studio using C++. The controller of SIASUN SR7CL industrial robot is a closed-system that cannot integrated with STEP-NC interpreter directly. Therefore, the robot path planning module of STEP-NC interpreter works in standalone mode. The STEP-NC interpreter developed in this paper can be used with industrial robot from other manufactured by updating the postprocessor to support more kinds of robot program language. A rectangle shape workpiece with one round hole and one closed pocket on upper surface is designed to be positioned and machined on CLRMS. Real-time monitoring is currently not implemented in this prototype platform and will be considered in future research works. The key entities of STEP-NC file are show in part 2 of Fig. 20. The main workplan contains one inspection workingstep and two machining workingsteps along with corresponding features and operations.

The open STEP-NC interpreter reads and interprets STEP-NC file and executes the three workingsteps in sequence. Firstly, an inspection workingstep is executed to determine the position and orientation of SCS in GCS by using the approach described in part B and C of section IV. Industrial camera and lens are installed on the ending tool of robot to conduct the inspection task. Line #241 to #245 are added at the end of STEP-NC file to record the inspection results. Entity rawpiece_position_result is defined in part B of section III is used to link the inspection results with corresponding inspection workingstep.

Tool path for machining each manufacturing feature is planned in FCS and transformed to MCS of robot by using (16) in part C of section IV. The robot path is shown in part 3 of Fig. 20. Then the postprocessor generates robot control program, which is sent to robot controller through its input interface. The motorized spindle with end milling cutter is installed on ending tool of robot to conduct the machining task. The camera is temporarily dismounted during machining to avoid interference. The machining process and finished part is shown in part 5 and 6 of Fig. 20.

Experimental result indicates that the proposed STEP-NC compliant closed-loop robot machining system with open STEP-NC interpreter can execute integrated robot machining process plan contains workingstep for rawpiece positioning and machining appropriately. The positioning results can also be recorded and integrated with other entities by using the STEP-NC data model designed in section III. The process of robot path planning for a STEP-NC manufacturing feature takes only a few seconds, which greatly reduced the machining robot programming time compared with teaching and off-line programming method, especially in small batch manufacturing.

## VI. CONCLUSION

As described above, the architecture and methodology of STEP-NC compliant closed-loop robot machining system

are proposed in this paper to realize integrated bidirectional data stream and simplify robot programming. The AAM of CLRMS is built by using IDEF0 method. There are five functional activities in CLRMS namely part design (A1), integrated process planning (A2), machining robot path planning (A3), robot machining and process control (A4), and machining knowledge management (A5). The ARM of STEP-NC data for CLRMS is defined by using both EXPRESS-G and EXPRESS method to represent integrated robot machining process plan, inspection data, monitoring data, machining status data, and mechanical equipment. The newly defined STEP-NC data model is integrated with existing data model in ISO 14649. A framework of multi branch coordinate systems is defined in STPE-NC data model to describe the relative position and orientation between mechanical equipment and workpiece in CLRMS. Functional activity A3 and A4 are implemented based on an open STEP-NC interpreter, which is developed using C++ to realize online interpreting of STEP-NC file contains entities of the proposed integrated STEP-NC data model. An industrial camera is integrated with the STEP-NC interpreter to determine the position of workpiece using template matching algorithm. Tool path in FCS is planned and transformed to robot path in MCS of machining robot rapidly, which remarkably reduced the programming time of robot machining. Implementation platform of CLRMS is built, on which a case study of inspection and machining of a rectangle shape plastic workpiece is conducted. The feasibility and validity of the proposed framework and methodology of CLRMS is verified by the experimental result.

## VII. FUTURE RESEARCH

Future research works should be carried out to enhance and optimize the proposed system. Firstly, the positional accuracy of industrial robot is usually limited, which should be calibrated for compensation. However, the commonly used laser tracking system is costly and time-consuming. In that case, high efficiency machining robot accuracy calibration approach based on machine vision will be investigated. Furthermore, template matching algorithm may not be suitable if the machining environment is complex and variable. More adaptable machine vision algorithm based on deep learning is needed for online positioning and inspection of workpiece. Secondly, real-time monitoring of machining status and control is necessary to improve the stability of robot machining process. The placement of sensors, real-time data processing algorithm, and real-time adaptive control algorithm will be studied to reduce deformation and chatter in robot machining. The data mining method for extracting the relationship between robot parameters, processing parameters, and real-time status information should also be investigated to predict and recognize the deformation and chatter of robot.

## REFERENCES

[1] B. Tao, X. Zhao, S. Yan, and H. Ding, "Kinematic modeling and control of mobile robot for large-scale workpiece machining," *Proc. Inst. Mech. Eng. B, J. Eng. Manuf.*, vol. 236, nos. 1–2, pp. 29–38, Jan. 2022.

[2] W. Ji and L. Wang, "Industrial robotic machining: A review," *Int. J. Adv. Manuf. Technol.*, vol. 103, nos. 1–4, pp. 1239–1255, Jul. 2019.

[3] F. Leali, A. Vergnano, F. Pini, M. Pellicciari, and G. Berselli, "A work-cell calibration method for enhancing accuracy in robot machining of aerospace parts," *Int. J. Adv. Manuf. Technol.*, vol. 85, nos. 1–4, pp. 47–55, Jul. 2016.

[4] L. Yuan, Z. Pan, D. Ding, S. Sun, and W. Li, "A review on chatter in robotic machining process regarding both regenerative and mode coupling mechanism," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 5, pp. 2240–2251, Oct. 2018.

[5] H. N. Huynh, H. Assadi, E. Rivière-Lorphèvre, O. Verlinden, and K. Ahmadi, "Modelling the dynamics of industrial robots for milling operations," *Robot. Comput.-Integr. Manuf.*, vol. 61, Feb. 2020, Art. no. 101852.

[6] P. Mesmer, M. Neubauer, A. Lechler, and A. Verl, "Drive-based vibration damping control for robot machining," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 564–571, Apr. 2020.

[7] Z. Pan, J. Polden, N. Larkin, S. Van Duin, and J. Norrish, "Recent progress on programming methods for industrial robots," *Robot. Comput. Integr. Manuf.*, vol. 28, no. 2, pp. 87–94, 2012.

[8] S. Kumar, A. Nassehi, S. T. Newman, R. D. Allen, and M. K. Tiwari, "Process control in CNC manufacturing for discrete components: A STEP-NC compliant framework," *Robot. Comput.-Integr. Manuf.*, vol. 23, no. 6, pp. 667–676, Dec. 2007.

[9] X. Zhang, R. Liu, A. Nassehi, and S. T. Newman, "A STEP-compliant process planning system for CNC turning operations," *Robot. Comput.-Integr. Manuf.*, vol. 27, no. 2, pp. 349–356, Apr. 2011.

[10] C. Brecher, M. Vitr, and J. Wolf, "Closed-loop CAPP/CAM/CNC process chain based on STEP and STEP-NC inspection tasks," *Int. J. Comput. Integr. Manuf.*, vol. 19, no. 6, pp. 570–580, Sep. 2006.

[11] S. Kumar, S. T. Newman, A. Nassehi, P. Vichare, and M. K. Tiwari, "An information model for process control on machine tools," in *Proc. 6th Cirp-Sponsored Int. Conf. Digit. Enterp. Technol.*, vol. 66, 2010, pp. 1565–1582.

[12] F. Zhao, X. Xu, and S. Xie, "STEP-NC enabled on-line inspection in support of closed-loop machining," *Robot. Comput.-Integr. Manuf.*, vol. 24, no. 2, pp. 200–216, Apr. 2008.

[13] Y. F. Zhao and X. Xu, "Enabling cognitive manufacturing through automated on-machine measurement planning and feedback," *Adv. Eng. Informat.*, vol. 24, no. 3, pp. 269–284, 2010.

[14] M. Wosnik, H. Rüdele, and P. Klemm, "Process-informed machining objects for closed-loop process planning," *Int. J. Comput. Integr. Manuf.*, vol. 22, no. 11, pp. 1054–1064, Nov. 2009.

[15] F. Ridwan and X. Xu, "Advanced CNC system with in-process feed-rate optimisation," *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 3, pp. 12–20, Jun. 2013.

[16] S. Sivakumar and V. Dhanalakshmi, "An approach towards the integration of CAD/CAM/CAI through STEP file using feature extraction for cylindrical parts," *Int. J. Comput. Integr. Manuf.*, vol. 26, no. 6, pp. 561–570, Jun. 2013.

[17] P. Hu, Z. Y. Han, H. Y. Fu, and D. D. Han, "Architecture and implementation of closed-loop machining system based on open STEP-NC controller," *Int. J. Adv. Manuf. Technol.*, vol. 83, nos. 5–8, pp. 1361–1375, 2016.

[18] 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," *Int. J. Adv. Manuf. Technol.*, vol. 91, nos. 5–8, pp. 2499–2525, Jul. 2017.

[19] C. Danjou, J. Le Duigou, and B. Eynard, "Closed-loop manufacturing, a STEP-NC process for data feedback: A case study," *Proc. CIRP*, vol. 41, pp. 852–857, Jan. 2016.

[20] C. Danjou, J. Le Duigou, and B. Eynard, "Manufacturing knowledge management based on STEP-NC standard: A closed-loop manufacturing approach," *Int. J. Comput. Integr. Manuf.*, vol. 30, no. 9, pp. 995–1009, Sep. 2017.

[21] B. Solvang, L. K. Refsahl, and G. Sziebig, "STEP-NC based industrial robot CAM system," *IFAC Proc. Volumes*, vol. 42, no. 16, pp. 245–250, 2009.

[22] W. Xiao, J. Huan, and S. Dong, "A STEP-compliant industrial robot data model for robot off-line programming systems," *Robot. Comput.-Integr. Manuf.*, vol. 30, no. 2, pp. 114–123, Apr. 2014.

[23] S. N. R. Minango and J. C. E. Ferreira, "Combining the STEP-NC standard and forward and inverse kinematics methods for generating manufacturing tool paths for serial and hybrid robots," *Int. J. Comput. Integr. Manuf.*, vol. 30, no. 11, pp. 1203–1223, Nov. 2017.

[24] S. Zivanovic, N. Slavkovic, and D. Milutinovic, "An approach for applying STEP-NC in robot machining," *Robot. Comput.-Integr. Manuf.*, vol. 49, pp. 361–373, Feb. 2018.

[25] D. Milutinovic, M. Glavonjic, N. Slavkovic, Z. Dimic, S. Zivanovic, B. Kokotovic, and L. Tanovic, "Reconfigurable robotic machining system controlled and programmed in a machine tool manner," *Int. J. Adv. Manuf. Technol.*, vol. 53, nos. 9–12, pp. 1217–1229, Apr. 2011.

[26] J. S. Toquica, S. Živanović, A. J. Alvares, and R. Bonnard, "A STEP-NC compliant robotic machining platform for advanced manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 95, nos. 9–12, pp. 3839–3854, Apr. 2018.

[27] J. S. Toquica, S. Zivanovic, R. Bonnard, E. Rodriguez, A. J. Alvares, and J. C. E. Ferreira, "STEP-NC-based machining architecture applied to industrial robots," *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 41, no. 8, pp. 1–18, Aug. 2019.

[28] N. Slavkovic, S. Zivanovic, and D. Milutinovic, "An indirect method of industrial robot programming for machining tasks based on STEP-NC," *Int. J. Comput. Integr. Manuf.*, vol. 32, no. 1, pp. 43–57, Jan. 2019.

[29] A. J. Alvares, E. Rodriguez, C. I. R. Jaimes, J. S. Toquica, and J. C. E. Ferreira, "STEP-NC architectures for industrial robotic machining: Review, implementation and validation," *IEEE Access*, vol. 8, pp. 152592–152610, 2020.

[30] X. W. Xu, H. Wang, J. Mao, S. T. Newman, T. R. Kramer, F. M. Proctor, and J. L. Michaloski, "STEP-compliant NC research: The search for intelligent CAD/CAPP/CAM/CNC integration," *Int. J. Prod. Res.*, vol. 43, no. 17, pp. 3703–3743, Sep. 2005.

[31] Y. Zhao, Q. Liu, W. Xu, H. Yuan, and P. Lou, "An ontology self-learning approach for CNC machine capability information integration and representation in cloud manufacturing," *J. Ind. Inf. Integr.*, vol. 25, Jan. 2022, Art. no. 100300.

[32] W. Gao, C. Zhang, T. Hu, and Y. Ye, "An intelligent CNC controller using cloud knowledge base," *Int. J. Adv. Manuf. Technol.*, vol. 102, nos. 1–4, pp. 213–223, May 2019.

[33] Z. Han, H. Jin, Y. Fu, and H. Fu, "Cutting deflection control of the blade based on real-time feedrate scheduling in open modular architecture CNC system," *Int. J. Adv. Manuf. Technol.*, vol. 90, nos. 9–12, pp. 2567–2579, Jun. 2017.

[34] J. Zhang, Y. Tian, Z. J. Ren, Y. Han, and Z. Y. Jia, "Rotation calibration method for thrust based on force error analysis," *Exp. Techn.*, vol. 44, no. 1, pp. 85–98, Feb. 2020.

[35] *Industrial Automation Systems and Integration—Product Data Representation and Exchange Part 11: Implementation Methods: The EXPRESS Language Reference Manual*, ISO Standard 10303, 2004.

[36] *Data Model for Computerized Numerical Controllers Part 10: General Process Data*, ISO Standard 14649, 2004.

[37] *Data Model for Computerized Numerical Controllers Part 11: Process Data for Milling*, ISO Standard 14649, 2002.

[38] *Data Model for Computerized Numerical Controllers Part 12: Process Data for Turning*, ISO Standard 14649, 2003.

[39] *Data Model for Computerized Numerical Controllers Part 111: Tools for Milling*, ISO Standard 14649, 2002.

[40] *Data Model for Computerized Numerical Controllers Part 121: Tools for Turning Machines*, ISO Standard 14649, 2003.

[41] *Industrial Automation Systems and Integration—Product Data Representation and Exchange Part 105: Integrated Application Resource: Kinematics*, ISO Standard 10303, 1996.

[42] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[43] J. Cai and T. Lei, "An autonomous positioning method of tube-to-tubesheet welding robot based on coordinate transformation and template matching," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 787–794, Apr. 2021.

[44] J. Xu, X. He, and W. Ji, "Mechanical system and template-matching-based position-measuring method for automatic spool positioning and loading in welding wire winding," *Appl. Sci.*, vol. 10, no. 11, pp. 1–14, 2020.

**PO HU** was born in Shijiazhuang, Hebei, China, in 1986. He received the B.S. degree in mechanical design manufacture and automation from Yanshan University, Qinhuangdao, Hebei, in 2009, and the M.S. and Ph.D. degrees in mechanical manufacture and automation from the Harbin Institute of Technology, Harbin, Heilongjiang, China, in 2011 and 2016, respectively. He is currently working as a Lecturer with the School of Mechanical Engineering, Shijiazhuang Tiedao University. His research interests include intelligent manufacturing, machining robot, open architecture CNC, and intelligent control of large engineering machinery.

**JIE LI** was born in 1979. He received the Ph.D. degree from the Chinese Academy of Agricultural Mechanization Sciences, in 2007. He is currently working as a Professor with the School of Mechanical Engineering, Shijiazhuang Tiedao University, China. He has published over 40 technical papers. His research interests include mechanical structure fatigue and material failure analysis.

**JINGBO GUO** was born in Shijiazhuang, Hebei, China, in 1966. He received the Ph.D. degree from the Beijing University of Technology, in 2006. He is currently working as a Professor with the School of Mechanical Engineering, Shijiazhuang Tiedao University, China. His research interests include design theory and method, reliability and failure diagnosis of shield machine, TBM, shield cutter, remanufacturing of shield machine, and TBM.

**LIANPENG ZHANG** received the B.S. degree from Yanshan University, in 2010, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, in 2012 and 2017, respectively. He is currently a Lecturer with the School of Mechanical Engineering, Shijiazhuang Tiedao University. His research interests include electro-hydraulic servo control, parallel robot, and robotic design and control application.

**JIE FENG** received the B.S. degree from Yanshan University, in 2010, and the M.S. and Ph.D. degrees from the Harbin Institute of Technology, in 2012 and 2017, respectively. She is currently a Lecturer with the School of Mechanical Engineering, Shijiazhuang Tiedao University. Her research interests include electro-hydraulic servo control, mechanical engineering and material processing, and forming technology.

· · ·