Implementation of MTConnect in machine monitoring system for CNCs

Siyi Chen
Advanced Manufacturing Institute
Chongqing University of Posts and Telecommunications
Chongqing, China
siyichen903@163.com

Chao Yin
State Key Laboratory of Mechanical Transmission
Chongqing University
Chongqing, China
ych925@cqu.edu.cn

Xiaobin Li
School of Economics and Business Administration
Chongqing University
Chongqing, China
xiaobin_lee@cqu.edu.cn

Abstract**—**In current manufacturing shop floor, the diversity of CNCs' interfaces and communication protocols makes it extremely challenging for shop managers to collect the device data and monitor their states. To allow the existing machine tools to be easily connected and monitored by a "plug and play" way, this paper proposes a machine monitoring system for CNCs based on MTConnect, and describes the implementation of MTConnect protocol in the proposed system in detail. The implementation includes establishing information model for the CNCs according to MTConnect standard, collecting and processing data from CNCs by MTConnect Adapter, establishing communication with MTConnect Agent, and implementing data interaction operations between MTConnect Agent and monitoring application using MTConnect HTTP requests. Finally, the implemented system is illustrated with the CNC VMC-1060L in a realistic application, and the results show the practical feasibility of this MTConnect based system for CNCs monitoring.

Keywords—Machine monitoring; MTConnect; System; CNC

I. Introduction

During the last few decades, the rapid advancement of Information and Communication Technologies (ICT) has been reshaped the manufacturing industry in all aspects. New manufacturing methods have been put forward constantly and aroused great interest of scholars and manufacturers all around the world, like the intelligent manufacturing, Cloud Manufacturing (CMfg) [1], and Cyber-Physical System (CPS) [2]. However, as the essential part to realize these good versions for manufacturing, the connection of devices at shop floor and the data acquisition of different machines are still issues for some shop managers, and even basic machine monitoring is out of reach because of the proprietary interfaces and communication protocols supported by different types of machines.

To address these issues, various data acquisition and machine monitoring solutions are proposed by domestic and overseas scholars. For instance, an open architecture using internet for data acquisition and monitoring in machine tools with CNC is proposed in [3]. To realize remote monitoring and management for machine tools, a method of information integration of Numerical Control (NC) equipment in [4] is proposed, which includes interconnection architecture, information model for devices and process of information interchange. Ben et al. uses a mobile phone unit as a communication method between customers and machine tool manufacturers to realize the remote monitoring and maintenance system [5]. As for the studies abroad, a universal data access server using OPC and XML technologies is proposed to realize the modern distributed data acquisition and monitoring system in [6]. [7] presents a data acquisition method by analyzing the information requirements of active and a system architecture of effective data acquisition was proposed.

The approaches mentioned above have been successfully used in manufacturing shop floor, however, with the lack of a "plug-and-play" standard for different manufacturing equipment, the data acquisition for monitoring system is still difficult, time-consuming and expensive. Most shops and plants are already familiar with situations in which collecting data and monitoring machines requires creating or adding separate, unique Adapters or interfaces to for every machine-to-machine connection, every software application and every network connection.

Fortunately, the emerging MTConnect standard makes it possible for the monitoring system to accurately and consistently collect data from any MTConnect compatible machine. Since MTConnect was first introduced in 2008, it is becoming a widely accepted Internet communication standard for factory automation in recent years. Overseas manufacturers have successively announced their support for MTConnect standard, and it has become the defacto standard among equipment providers, software vendors and system integrator.

While MTConnect is commonly pre-installed on brands such as Boeing, FANUC and Okuma, it seems still a new thing to domestic equipment suppliers. Most existing machine tools in china are not MTConnect compatible, and very few works

2572-6609/17 $31.00 © 2017 IEEE
DOI 10.1109/ES.2017.19

have been done to implement MTConnect for these CNCs. [8] designs a general interface of information collection based on MTConnect architecture, and designed the data items for the monitoring machine. [9] analyzes the structure of CNC engraving machine and establishes MTConnect device model. Therefore, making current CNCs MTConnect compatible shows a great significance for domestic manufacturers and shop managers to realize factory integration and automation.

Hence, with the vision of achieving a MTConnect standardized communication platform for CNCs, this paper has particularly focused on implementing the MTConnect for the monitoring system that would allow the existing CNCs to be accessed and monitored from anywhere over the Internet. In the next section, we give a brief introduction of MTConnect standard and relate it to our work. Afterward, the implementation of MTConnect for the CNC machine monitoring system is presented as follows. First, the structure and mechanism of the proposed system for CNCs monitoring are explained in Section 3. Then, the steps to implement the MTConnect for CNCs monitoring are detailed in Section 4. Finally, the practical feasibility of our proposal is illustrated in a realistic application in Section 5. Section 6 concludes the paper.

II. MTCONNECT INTRODUCE

MTConnect is an open and royalty-free communication protocol that commits to promote the interoperability between manufacturing devices and software from different suppliers. The principal goal of MTConnect is to create a "universal machine language" that is understandable to all machines as well as the users.It provides connectivity and the capability to monitor and collect data from the entire production floor: machines, cells, devices, and processes. This capability opens the door for true computer-integrated intelligent manufacturing. The most notable components of MTConnect are the MTConnect Adapter and the MTConnect Agent. Abbreviations and Acronyms

A. MTConnect Adapter

MTConnect Adapter is the bridge between Agent and shop floor devices or sensors. It enables existing shop floor devices "to speak MTConnect," which means that the data produced by shop floor equipment will be translated into a format that can be processed by MTConnect Agent. Adapter can be a simple software application or a combination of software and hardware, which depends on the requirement of monitoring system. Depending on different interfaces exposed by machine Tools, PLCs, sensors and Human Input Devices, connections between Adapters and devices can be over serial ports and network ports, or the Application Programming Interface(API) offered by the equipment provider (which is relatively easy to create the Adapter).

It is worth to note that the Adapter is not actually covered in the MTConnect standard. As a separate application provided on the device side of the implementation, it is still a very vital part of MTConnect implementation as it delivers the ability to communicate with different types of machines using various protocols and makes it possible to use one generic Agent for all MTConnect compatible devices.

B. MTConnect Agent

The Agent takes over where the Adapter leaves off. While Adapter collects data from devices and transmits the data to Agent, the Agent collects data from Adapter and transmits the XML format message to the client in the manufacturing cloud across the network.

One main job of the Agent is to format the data into an MTConnect Standard XML stream. This is the format readable by humans as well as machines. This simplicity makes it easy to create data that can be shared by different applications. The Agent must also provide adequate buffer storage to hold data until applications request it. The buffer size should suit the needs of monitoring. Another important job for the Agent is to provide an interface to the network so data can be accessed by outside applications. MTConnect Agent is simply Web server that responds to HTTP requests from applications with the appropriate HTTP response. Like the Adapter, the Agent is a software program, but users have several options for where the Agent is installed. It might run on the same control unit computer where the Adapter is installed.

While most studies in this area focus on the development of the MTConnect Agent, the main purpose of this paper is the specific implementation of the communication between Adapter and Agent, as well as the request/response mechanism between Agent and the client application, which are the key part for the MTConnect implementation for CNC machine tolls monitoring system.

III. DESIGN OF THE MONITORING SYSTEM BASED ON MTCONNECT TECHNOLOGY

By using MTConnect technology, the monitoring system is designed to be applicable for both traditional CNCs and new CNCs which can support Internet communication protocols such as OPC-UA or MTConnect. The system architecture is shown in Fig.1.

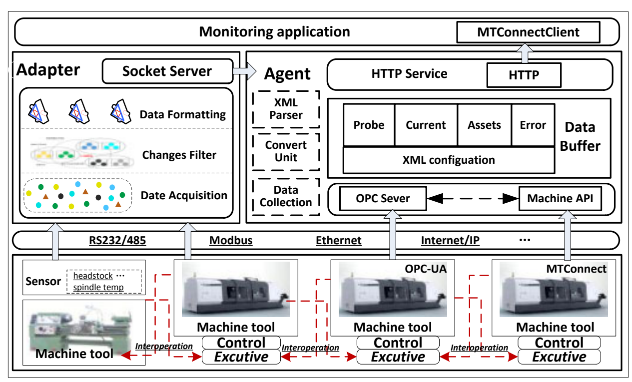

Fig. 1. Architecture of CNCs monitoring system based on MTConnect

A. System architecture and components

In the proposed system, the CNC machine tool's numerical control system is comprised of two components: the upper control system on a remote networked PC, and the Industrial Ethernet based motion controller embed in the CNC machine tool itself. The upper control system and the motion controller are connected by the real-time Ethernet.

The Agent is integrated into the upper control system by the software interface, running as an independent thread. Each Agent can connect directly to one or more CNCs which are MTConnect-compatible. The numerical control system collects the raw data from CNCs, and the Agent acquires the required data from the machine API by function calling. For CNCs that are not MTConnect-compatible, the Agent needs the Adapter to convert the data into MTConnect format, and then get the data from Adapter by establishing the socket on TCP/IP. The basic function of Agent is described previously.

The Adapter is developed as a hardware which is physically separated from the CNC machine tool and the PC control system. The connection between Adapter and CNCs can be established on serial peripheral interface or Ethernet port, and the data can be collected by RS232/485, USB, or APIs. For the machine data which cannot be offered by its numerical control system, some sensors need to be added to collect data for Adapter. For example, to gather information of the spindle, the vibration sensor and spindle temp sensor can be installed on CNCs and then connected to corresponding Adapters. The functions of the Adapter is designed as follows: getting current state information of CNCs; formatting the data into the MTConnect Standard messages; establishing communication with Agent and sending the data as text by socket.

The application refers to the actual user or demand side of the data collected by MTConnect Agent, such as the remote monitoring software. The monitoring application contains the MTConnect Client. The function of the Client is to work with Agent to query and provide data for monitoring application. The Client and Agent are connected by the Internet network, and the communication between Agent and Client are based on MTConnect data model and MTConnect HTTP protocol. For example, by using the MTConnect standard dataitems designed for CNCs and the *probe, current and sample* requests of MTConnect, the Client can easily get the status data from shop floor machines and realize machine monitoring and factory automation.

B. System operation mechanism

Fig. 2 illustrates the data flow of the proposed system from underlying device to application on the Internet.

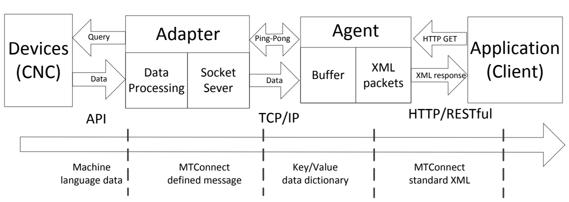

Fig. 2. Data flow from the device to the client

Firstly, the CNC machine tool is connected to the Adapter by the interfaces it exposed as TCP/IP, RS-232, RS-485, and so on. Once the machine is ready to output its data, the Adapter is responsible for communicating with the CNC machine tool (controller) and sends commands to collect machine state information. Then the raw data in machine language collected by Adapter would be translated into MTConnect-defined message based on text, which can be understandable by the Agent. Then data sent by Adapter would be stored in the store buffer of Agent with a Key/Value data dictionary. The data transmission between Adapter and Agent is based on socket using TCP/IP protocol. On the one hand, the Adapter periodically detects the value changes of monitored data items and only sends changed data to the Agent. By default, the Adapter samples the machine state every 1ms.

On the other hand, it responds to Agent's requests to send the data from the machine. The data sent by Adapter would be stored in the buffer storage in the Agent. Meanwhile, a HTTP server with RESTful interface is created to wait for the client's requests. When a certain request (probe, current or sample) is made to the Agent, it converts the required data into a standard XML format based on the MT-Connect Standard and sends the data with the appropriate HTTP response.

IV. Implementation of the Proposed System

A. Establishing information model of the CNC machine tool based on MTConnect standard

The first step to implement MTConnect for the proposed system is to establish the information model of the CNC machine tool according to its logical structure. The information model of the monitored CNC machine tool is also defined by MTConnect standards. In order to meet the interoperability requirements of heterogeneous devices, it uses unified terminology to describe devices so that different machine tools can be described by a standardized information model. The devices connected to a MTConnect agent are called Devices (device set) and can contain serveral different devices, called Device. Device usually represents a machine tool or other device that has its own attributes such as identification numbers and names. Device consists of several Components (component set). Component is an abstract type, including types as Axes, Controller, Door, Sensors, and Systems. Each specific component can also have sub components, and their names and attributes are described by Dataitems. All dataitems are semantically and contextually identified, which describe the corresponding information that is capable to be provided to Agent and applications from the machine component.

Following MTConnect standard, the CNC machine tool of the proposed system is defined as a Device in a Devices, which consists of several components, the Axes, the Controller, the Door, the Sensor and the System. Each Component has its Sub-Component, for example, the Axes consists of three linear feed shafts X, Y, Z, and a rotary C. The dataitems below each component describe the main attributes of the component. According to the monitoring requirements of our system, the information which needs to be collected includes the position and load of linear X, Y, Z; the speed of spindle, the mode and load of rotary, the status and mode of controller, the operation situation of path, and the condition of coolant, hydraulic and electric system. The corresponding Dataitems are created for the monitored information, as shown in table I, and the parameter variable and storage address of the information are matched for each Dataitem following MTConnect. The information model of the CNC machine tool is built as shown in Fig.3, taken the CNC machine center VMC-1060L as an example.

TABLE I. Dataitems of CNC machine center VMC-1060L

Component	Sub-Component	Component Name	Dataitem
Axes	Rotary	C	Spindle_Speed
	Linear	X, Y, Z	Load
			Rotary_Mode
			Load
			Position
Controller		Warning	Message
		Estop	EmergencyStop
	Path	PathFeedrate	Path_Position
		ControllerMode	Mode
		Execution	execution
		ToolId	tool_id
System	Electric	Power	Power_State

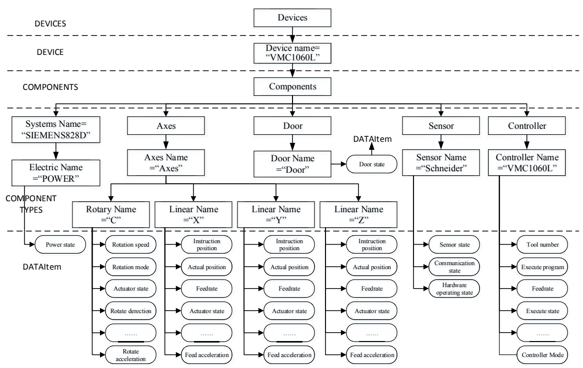

Fig. 3. Information model of CNC machine center VMC-1060L

B. Processing data by MTConnect Adapter

The information collected form CNCs such as the execution state, control mode, spindle speed, etc., always has different data types and needs to be translated into MTConnect format. According to MTConnect standard, the format of the message that Adapter returned to Agent is simple: it is always beginning with a time stamp, and subsequently the name and value of the dataitem. According to the different type of dataitems, there are four kinds of messages: Event, Sample, Condition, and Message. For example, Event describes the changes of a series of logical states, such as running state, mode of operation, and current program name. It describes the state of a device or component, which is asynchronous and has discrete value.The format of the Event message is like: Timestamp|name|value|name|value.

To transform the collected data into the defined format, a Abstract Factory pattern is used here to create an universal interface to convert the different data format into MTConnect-defined message for the Agent. The parent class defines a unified interface for the names, values, and conversion functions of dataitems, and the subclass inherit the parent class so as to ensure the consistency of data conversion interface. The class view of the Adapter design is showed in Fig. 4.

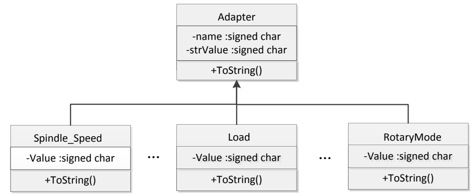

Fig. 4. Class view of Adapter designed by abstract factory pattern

C. Establishing communication with MTConnet Agent by Socket

As introduced previously, while MTConnect specifies the communication process from the Agent to the application, it does not give the way to transmit data from Adapter to Agent. Hence, a socket server based on TCP is designed to realize the communication and data transmission between Adapter and Agent. In our implementation, the socket development utilizes the C# Framework. Since the Framework can provide the communication protocols and detection of changed dataitems, the rest work to develop the program are as follows: declaring communication parameter variables and data types, including server IP address and port; determining the dataitems for each monitored parameter (which we have done in section 3.1) and the addresses to receive and send collected data; gathering data periodically; responding to the Agent's data request and sending the changed values during each PLC scan cycle.

Firstly, the main program calls the function from socket library to create a socket and complete port binding. Then it starts to listen for the corresponding port, and accept the connection request send by Agent. After establishing the connection between Adapter and Agent, the data can be read and write via the socket. On the one hand, the socket server would listen and response to the data request from the Agent, and use iRequestType() to decide the dataitems and datatype that required by the Agent. Then it calls the data processing function to translate the machine data into MTConnect-defined messages, and send it to the Agent. On the other hand, the server would check for updates of the values of dataitems during each PLC scanning period, and only the changed value will be sent. The main program design is shown in Fig.5.

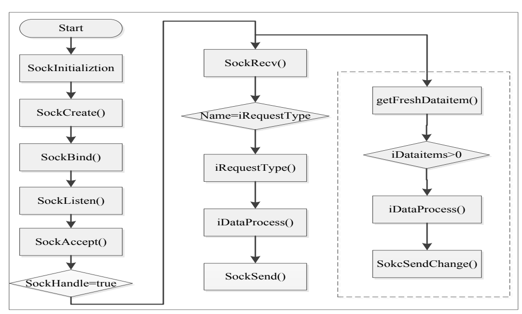

Fig. 5. The design of main program for socket server

D. Implementing data interaction between the Agent and application using MTConnect HTTP requests

The data interaction process of the monitoring system is realized by the request/response mechanism between the Agent and the client, which is the most important part of MTConnect implementation for CNCs monitoring. To collect the data from Agent, the application makes an HTTP GET request to Agent, and the Agent responds in XML with appropriate HTTP response. The query requests of the application include four types as *probe* (device query), *current* (numerical value query), *sample* (sampling) and Asset, and the result of the query is described by MTConnect XML document as MTConnectDevice, MTConnectStreams, MTConnectError, MTConnectAssets. The MTConnectDevice describes the machine structure and dataitem that can be collected from the monitored devices; the MTConnectStreams records the series of events, sampling value and status value obtained from the device and components; the MTConnectAssets is used to describe attachments which are related to the manufacturing process but not the part of the component. The request and its corresponding MTConnect document are shown in Table Ⅱ. Using the *probe, current, sample* operations of MTConnect, the workflow of the application and Agent in data interaction is shown as follows, which is illustrated in Fig. 6.

- Firstly, the Agent connects to a name service server (LDAP) to convert the name of the monitored device to the URL (Universal Resource Identifier), and register its own URI information. Then the Agent begin to collect data from the device/Adapter, and store the information.
- The application uses LDAP to send a request for the location of the device, and the LDAP server returns the URI information of the device.
- Using the URI acquired by LDAP, the application sends the *probe* request to the Agent, and the Agent returns the device information model and dataitems information.
- The application sends the *current* request to the Agent and get the current state of the device and components and the order number of current data. Then the application can sample and query the Agent at a certain frequency as needed, and get a stream of changed data from a certain time point.

Following the steps suggested above and based on the information model established in section 3.1, the *probe, current, sample* are send to Agent successively to collect data from the monitored CNC machining center VMC-1060L.

TABLE II. MTCONNECT REQUEST TYPE FOR AGENT

Request type	Request content	Return document
Probe	Devices, components, and dataitems	MTConnectDevices
Assets	Key/Value store of device information	MTConncetAssets
Sample	Stream of changes from a certain point	MTConncetStreams
Current	Current values of dataitems	MTConncetStreams

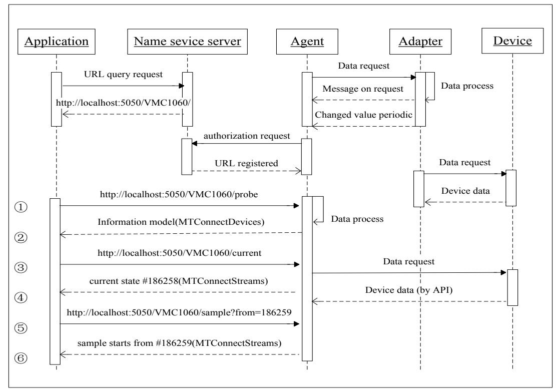

Fig. 6. Interaction progress of data acquisition and monitoring

The response to *probe* request is the MTConnectDevices document of VMC-1060L, which shows the physical attributes and dataitems information of the CNC machine center, as depicted in Fig. 7. The XML document is partial folded to make it more readable. The top tag <device> refers to the CNC machine center VML-1060L. The dataitem 'AVAILABILITY' is necessary to show whether the device is available to provide data for the monitoring application. The dataitems of Linear X are shown in detail, while the XML description for other components of Axes is omitted. The Controller describes the control unit of the machine center, and its component Path shows related information of the processing route. The System has the components as Coolant, Electric and Hydraulic, and this document only shows the dataitems of Electric. Because of the length limitation, the XML response of the *current* and *sample* request is not shown here.


```
MTConnectDevices xmlns:m="urn:mtconnect.org:MTConnectDevices:1.3"
xmlns="urn:mtconnect.org:MTConnectDevices:1.3" xmlns: xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi: schema Location="urn:mtconnect.org:MTConnectDevices:1.3 /
schemas/MTConnectDevices_1.3.xsd">
  <Header creationTime="2017-03-29T02:31:14Z" sender="mtcagent" instanceId="1490681234"
version="1.3.0.9" assetBufferSize="1024" assetCount="0" bufferSize="65536"/>
  <Devices>
    <Device id="dev" iso841Class="6" name="VMC-1060L" sampleInterval="10" uuid="000">
      <Description manufacturer="SystemInsights"/>
      <DataItems>
          <DataItem category="EVENT" id="avail" type="AVAILABILITY"/>
          <DataItem category="EVENT" id="dev_asset_chg" type="ASSET_CHANGED"/>
          <DataItem category="EVENT" id="dev_asset_rem" type="ASSET_REMOVED"/>
      </DataItems>
        <Components>
          <Axes id="ax" name="Axes">
            <Components>
              <Rotary id="c1" name="C">…</Rotary>
              <Linear id="x1" name="X">
                <DataItems>
                  <DataItem category="SAMPLE" id="x2" name="Xact"
nativeUnits="MILLIMETER" subType="ACTUAL" type="POSITION" />
                  <DataItem category="SAMPLE" id="x3" name="Xcom"
nativeUnits="MILLIMETER" subType="COMMANDED" type="POSITION" />
                  <DataItem category="SAMPLE" id="n3" name="Xload"
nativeUnits="PERCENT" type="LOAD" units="PERCENT"/>
                  <DataItem category="CONDITION" id="Xloadc" type="LOAD"/>
                  <DataItem category="CONDITION" id="Xsystem" type="SYSTEM"/>
                </DataItems>
              </Linear>
              <Linear id="y1" name="Y">…</Linear>
              <Linear id="z1" name="Z">…</Linear>
            </Components>
          </Axes>
          <Controller id="cn1" name="controller">
            <DataItems>
              <DataItem category="EVENT" id="msg" type="MESSAGE"/>
              <DataItem category="EVENT" id="estop" type="EMERGENCY_STOP"/>
              <DataItem category="CONDITION" id="clp" type="LOGIC_PROGRAM"/>
              <DataItem category="CONDITION" id="motion" type="MOTION_PROGRAM"/>
              <DataItem category="CONDITION" id="system" type="SYSTEM"/>
            </DataItems>
            <Components>
              <Path id="pth" name="path">…</Path>
            </Components>
          </Controller>
          <Systems id="systems" name="systems">
            <Components>
            <Electric id="el" name="electric">
              <DataItems>
                <DataItem category="EVENT" id="p2" name="power" type="POWER_STATE"/>
              </DataItems>
            </Electric>
            <Coolant id="cool" name="coolant">…</Coolant>
            <Hydraulic id="hsys" name="hydrolic">…</Hydraulic>
          </Components>
        </Systems>
      </Components>
    </Device>
  </Devices>
</MTConnectDevices>
```


Fig. 7. XML response for *probe* request

V. APPLICATION VERIFICATION

The implementation of the proposed system based on MTConnect is applied in a CNC machining center VMC-

1060L. The maximum travel of the CNC machining center with the X/Y/Z axis is1000/600/900 mm; X/Y/Z axis feed speed is 24/24/18m/min; the highest speed of spindle is 8000rpm; the main motor power is 11kw, the capacity of the cutter is 24, the positioning accuracy is 0.005mm.

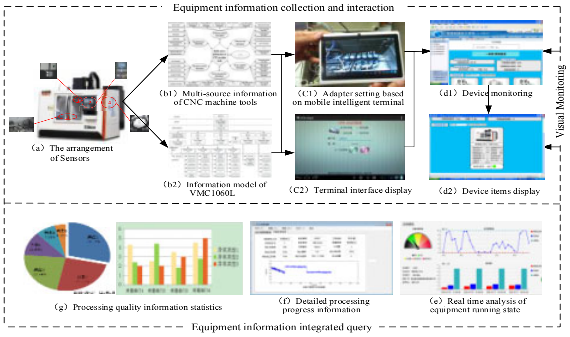

Fig. 8. The experimental verification

- The arrangement of Sensors. According to monitoring requirements of the system, the corresponding sensors are arranged to collect data which cannot obtained from the NC system. The infrared temperature sensor, data collector, production quality detector and tool detector are arranged respectively at the position ①, ②, ③, ④as shown in Fig.8a, to work with the numerical system to collect the raw data of the CNC machine tool.
- Adapter based on mobile intelligent terminal. In this implementation, the adapter is embedded into the intelligent interactive terminal, which provides "one-stop" service function for data collection and machine monitoring system. The mobile intelligent terminal software is developed in Android platform which can be connected with enterprise LAN by the wireless network.
- Agent development based on MTConnect.NET Agent SDK. This Agent development tool provides the interfaces and components for Agent programming with MTConnectAgentCore.dll, which can be embedded directly in the upper control system of CNCs. Using Agent IMachineAPI interface, the Agent can directly read data in the NC system, and Idata and IhttpServer interface can realize the communication and data interaction with monitoring application.
- Information visualization of the CNC machine tool. The monitoring system provides a visualization module for the information of CNC machining center, which is developed in our previous work. The information of the CNC machining center VMC860L can be visible in the application, including the X/Y/Z axis real-time value and feed speed, spindle speed and steering, tool number, real-time power, the current program number, control mode and safety gate status, as shown in 8e, 8f, 8g.

VI. CONCLUSIONS AND FUTURE WORK

The monitoring for CNCs using MTConnect is a promising aspect for real intelligent manufacturing. This paper proposed a MTConnect based monitoring system for CNCs, and described the implementation of MTConnect protocol for the proposed system in detail. The implementation includes establishing information model for the CNC machine tool, processing data from CNC machines tools by Adapter, establishing communication with MTConnect Agent, and implementing data interaction operations between Agent and monitoring application using MTConnect HTTP requests. Finally, the practical feasibility of this proposal is illustrated with the CNC VMC-1060L in a realistic application of manufacturing shop monitoring.

As MTConnect standard is a read-only protocol which can only collect and deliver monitoring data at the present version, the real-time control of the manufacturing operations and processes using MTConnect need to be studied in future work. In addition, because the CSMA/CD mechanism of the Ethernet, the response time also needs to be tested to verify the effect of this protocol. In the future, we will launch more experiments to test the system and work on expanding the current MTConnect to enable real-time control of manufacturing processes.

ACKNOWLEDGMENT

This work was supported by the Science and Technology Program of Guangdong Province (No. 2015A010103022), and Post-Doctoral Funding Project of Chongqing (No. Xm2016008).

REFERENCES

[1] Bo-Hu Li, Lin Zhang, Lei Ren, et al, “Typical characteristics, technologies and applications of cloud manufacturing,” Computer Integrated Manufacturing Systems, 2012, pp.1345-1356.

[2] Monostori L, “Cyber-physical Production Systems: Roots, Expectations and R&D Challenges,” Procedia CIRP, 2014, pp. 9-13.

[3] F. Ferraz Jr · R.T, Coelho, “Data acquisition and monitoring in machine tools with CNC of open architecture using internet, ” Int J Adv Manuf Technol (2005) 26, pp. 90–97 .

[4] Xiao-Hu Lu, Dong Yu, Yi Hu, Zhuang Yao, “Numerical control system information integration method based on Agent,” Journal of Jilin University, June 2015, pp.1980-1986.

[5] Ben Edrington, Bingyan Zhao, Adam Hansel, “Machine monitoring system based on MTConnect technology,” Proceedings of the 3rd International Conference in Through-life Engineering Services, Procedia CIRP,2014(22), pp.92 – 97.

[6] Dae-Seung Yoo, Vu Van Tan, and Myeong-Jae Yi, “A Universal Data Access Server for Distributed Data Acquisition and Monitoring Systems,” ICIC,2009,5754:762–773.

[7] Kwangmyeong Kim, Injun Choi, “Active data acquisition for proactive exception handling in manufacturing,” Adv Manuf Technol, 2009(43), pp.365–378.

[8] Lu Lin，Yu Dong，Hu Yi, “Research of the Open CNC Controller Network Interface Technology,” Journal of Chinese Computer Systems, 2015，Vol.36, No.3.

[9] Riliang Liu, Chenrui Zhang, Ke Wang, Jiang Yu, “Networked monitoring technology of numerical control machine tools based on MTConnect,” Computer Integrated Manufacturing System, May 2013, Vol.19, No.5.

[10] Xiaoqing Frank Liu, S M Nahian Al Sunny, et al, “Implementation of MTConnect for open source 3D printer in cyber physical manufacturing cloud,” Proceedings of the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, IDETC/CIE 2016-60378.