Manufacturing Blockchain of Things for the Configuration of a Data- and Knowledge-Driven Digital Twin Manufacturing Cell

Chao Zhang, Guanghui Zhou, *Member, IEEE*, Han Li, and Yan Cao

***Abstract*—Configuring intelligent manufacturing systems (IMSs) is significant for manufacturing enterprises to take a step toward Industry 4.0. However, most current IMS is configured based on the Industrial Internet of Things (IIoT) with a centralized architecture, which results in poor flexibility to handle manufacturing disturbances and limits capacity to support security solutions. To solve the above issues, this article combines IIoT with the permissioned blockchain and proposes a novel manufacturing blockchain of things (MBCoT) architecture for the configuration of a secure, traceable, and decentralized IMS. Then, hardware infrastructures and software-defined components of MBCoT are designed to provide an insight into the industrial implementation of IMS. Furthermore, the consensus-oriented transaction logic of MBCoT is presented based on a crash fault-tolerant protocol, which empowers MBCoT with a strong but resource-efficient encryption mechanism to support the autonomous manufacturing process. Finally, the implementation of an MBCoT prototype system and its application examples justify that the proposed approach is practical and sound. The evaluation experiment demonstrates that MBCoT equips IMS with a secure, traceable, stable, and decentralized operating environment while achieving competitive throughput and latency performance.**

***Index Terms*—Blockchain, Industrial Internet of Things (IIoT), Industry 4.0, intelligent manufacturing system (IMS), manufacturing blockchain of things (MBCoT).**

I. Introduction

In THE context of Industry 4.0, intelligent manufacturing has been regarded as the key for manufacturing enterprises to establish competitive advantages in a dynamic and global market [1], [2]. The intelligent manufacturing system (IMS) takes advantages of in-depth integration of new-generation information technologies, such as Industrial Internet of Things (IIoT), big data analytics, digital twin, and knowledge engineering, within the manufacturing industry to achieve a flexible, smart, and autonomous manufacturing process in order to maximize the product quality and throughput, while reducing cost [3], [4]. Therefore, configuring IMS is significant for manufacturing enterprises to take a step toward Industry 4.0. IMS configuration not only involves the establishment of ubiquitous connections for manufacturing devices (MDs) to the cyber system but also needs to bring together the software-defined functional nodes, such as data analytics tools and knowledge-based systems, within the cyber system to support autonomous operations and self-optimization of IMS during the manufacturing process [5].

Currently, IMS is typically deployed based on IIoT with a centralized architecture [6], which collects data from devices to a central cloud for intensive analytics. This architecture suffers from the following main drawbacks: 1) IIoT does build connections for devices to the cyber system, but could not break the communication barrier between functional nodes within the cyber system since they may be untrusted nodes; 2) stochastic errors or latency in such a centralized network may significantly influence the real-time performance of IMS; and 3) an open IMS could boost the competitiveness of manufacturing enterprises as customers nowadays prefer to participate in the manufacturing process, while the privacy and security of the open IMS in the current architecture are hard to be guaranteed. On the other hand, blockchain technology along with its secure, decentralized computing capacity is guiding another paradigm shift in the financial industry. Nowadays, blockchain is also introduced into IIoT to construct a secure, traceable, and decentralized IIoT network, generally applied in supply chain management [7], [8]. Naturally, the convergence of blockchain and IIoT may potentially overcome the current drawbacks of IIoT-based IMS. Nevertheless, configuring IMS based on blockchain and IIoT is still made difficult by the following challenging issues. First, IMS is more of a concept than a mature system, whose definition and participants remain unclear. Second, the integration of blockchain and IIoT is still in its infancy. How to handle the complex data flow, information flow, and knowledge flow in IMS while guaranteeing its privacy and security is still challenging.

To overcome the above drawbacks of IIoT-based IMS and figure out the challenging issues for applying blockchain in IMS, this article combines IIoT with the permissioned blockchain and proposes a novel manufacturing blockchain of things (MBCoT) architecture for the configuration of a secure,

Manuscript received January 7, 2020; revised June 4, 2020; accepted June 19, 2020. Date of publication June 30, 2020; date of current version December 11, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 51975463; in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-057; and in part by the Fundamental Research Funds for the Central Universities under Grant xzy022019066. *(Corresponding author: Guanghui Zhou.)*

Chao Zhang, Guanghui Zhou, and Han Li are with the School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China, and also with the State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710054, China (e-mail: superzc@stu.xjtu.edu.cn; ghzhou@mail.xjtu.edu.cn; 244068893@qq.com).

Yan Cao is with the School of Mechatronic Engineering, Xi'an Technological University, Xi'an 710021, China (e-mail: jantonyz@163.com).

Digital Object Identifier 10.1109/JIOT.2020.3005729

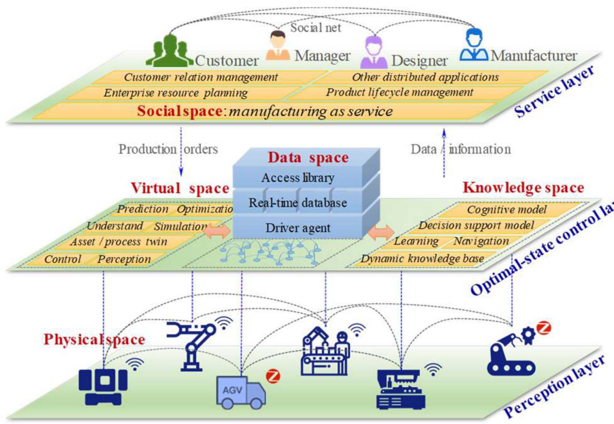


Fig. 1. Three-layer and 5-D framework for DK-DTMC.

traceable, and decentralized IMS. To this end, the article first defines a data-driven and knowledge-driven digital twin manufacturing cell (DK-DTMC) as a reference model for IMS. Then, MBCoT is explained based on its system architecture, hardware infrastructure, software-defined components, including MBCoT network, ledger model, and smart contracts, and consensus-oriented transaction logic. Finally, an MBCoT prototype system is implemented and its application examples and evaluation experiment demonstrate the feasibility and effectiveness of MBCoT for DK-DTMC configuration.

The remainder of the article is organized as follows. Section II introduces the research background and motivation behind the article. In Section III, we propose a novel MBCoT architecture for DK-DTMC configuration. Section IV constructs and evaluates an MBCoT prototype system to show the effectiveness of the approach. The conclusion is found in Section V.

II. Background

This section presents the research background behind the article, including a brief introduction of DK-DTMC and current trends for blockchain and IIoT integration. On that basis, the research motivation of the article is summarized.

A. DK-DTMC

In the context of Industry 4.0 [9], we introduced a new kind of IMS named DK-DTMC in the previous works [10], [11], which has powerful learning and cognitive capacities that could support autonomous operations of the manufacturing process. Specifically, as shown in Fig. 1, DK-DTMC is defined based on a three-layer and 5-D framework.

The *perception layer* perceives the real-time status of manufacturing resources, such as work-in-process (WIP) and smart MDs in physical space (PS) by a sensor network. In addition, the perceived data and real-time control orders are published and subscribed by smart gateways.

The *optimal-state control layer* makes PS operate at its optimal state [12] with the cooperation of data space (DS), virtual space (VS), and knowledge space (KS). Here, DS parses and collects real-time data from PS via adopters embedded in the driver agent. It also provides interfaces for other spaces to access data through an access library [13]. VS contains virtual manufacturing resources, such as the virtual WIP, virtual MDs, and virtual machining process, which are the software-defined mappings of physical resources that could simulate and understand the performance of DK-DTMC. KS integrates dynamic knowledge bases (DKB) and knowledge models (KMs) that act as the brain of DK-DTMC to handle various manufacturing problems in PS or VS [14], [15].

The *service layer* packages manufacturing capacities and resources as services to serve customers, managers, designers, manufacturers, etc., in social space (SS) [16]. SS integrates a variety of service systems [such as customer relation management (CRM) and enterprise resource planning (ERP)] and distributed applications (DAPPs), which bridge the gap between the supply of DK-DTMC and demand of customers.

DK-DTMC is a typical IMS that could be a perfect substitution of the traditional human-oriented manufacturing system. However, the implementation and configuration of DK-DTMC are still made difficult by the following undressed issues: 1) *security* is the primary issue since there are massive data, information, and knowledge transmitted between and accessed by different users in different spaces; 2) *distributed collaboration* is another important issue as there are many users with different roles in different spaces collaborating with each other to enable the autonomous operations of DK-DTMC; and 3) *traceability* is also an important issue as customers and companies nowadays prefer to track the manufacturing process in a service-oriented manufacturing paradigm.

B. Blockchain, IIoT, and Their Integration

A blockchain is defined as an immutable, traceable, decentralized, and secure database for recording transactions made on and maintained within a distributed network of mutually untrusting peers [17], [18]. Blockchains have emerged with Bitcoin that is an innovative peer-to-peer (P2P) electronic cash system and, nowadays, is regarded as a promising technology to run trusted exchanges within a distributed P2P network in the digital world [19], [20]. Generally, there are four building blocks for a blockchain, including the distributed ledger, smart contracts, consensus, and cryptography. In a blockchain network, every peer maintains a copy of the ledger [21] that records all transactions taking place on the network. Smarts contracts [22] are used to provide controlled access to the ledger. The consensus [23] defines a standard process of keeping the ledger transactions synchronized across the network. Cryptography [24] binds the network transactions and ledger data with a very strong crypto mechanism that makes these transactions and data hard to be tracked or tampered by unauthorized users. There are two kinds of blockchain platforms, including permissionless blockchains and permissioned blockchains. In permissionless blockchains [25], such as Bitcoin and Ethereum, anyone can participate in the P2P network without a specific identity. On the other hand, permissioned blockchains [26], such as Hyperledger Fabric, only permit a set of known, identified peers participating in the P2P network. According to the current experiences [17], [27], permissioned blockchains usually achieve higher throughput and lower latency than permissionless blockchains. The reason is that consensus, such as Proof of Works (PoWs) [28] in permissionless blockchains is complex and costly to handle the Sybil attack [29]. This makes permissioned blockchains more applicable for time-sensitive enterprise-level applications.

IIoT is a computing concept describing ubiquitous network connectivity for machine-to-machine (M2M) and industrial communication with innovative sensor technologies [30], [31]. IIoT paves the way for better perceiving and understanding the industrial process, thereby enabling smart and efficient production [32]. Nowadays, IIoT is gradually adopted by manufacturing enterprises to improve production performance through smart and remote management [33]. However, there are several bottlenecks for IIoT-based IMS configuration. Security and privacy are the primary concern in IIoT. Nevertheless, traditional protection mechanisms fail to secure IIoT systems due to the limited computing capacity at edge devices [34]. In addition, since the IIoT network is built on a centralized architecture, its deployment might be costly for enterprises as expensive central servers and ordinary maintenance are needed. What is more, stochastic latency or disturbances may significantly influence the real-time performance of IIoT-based IMS [35].

It is a natural way to integrate IIoT with the blockchain to build a secure, distributed, and stable blockchain IIoT network [36]. Actually, the integration of blockchain and IIoT has recently attracted the interest of stakeholders across industry and academia [37]. Bai *et al.* [38] presented a lightweighted blockchain-based platform for the configuration of a secure and trustable IIoT to support decentralized manufacturing applications. Liu *et al.* [39] proposed a lightweight blockchain system that was resource efficient and applicable for power-constrained IIoT scenarios via a green consensus mechanism. Rathee *et al.* [40] introduced a security hybrid IIoT framework using blockchain to ensure a secure product shipment, trace workers' locations, and product documentation. He *et al.* [41] developed a blockchain-based software status monitoring system to detect and handle malicious behaviors of IIoT devices. Huang *et al.* [42] proposed a blockchain system with a credit-based consensus mechanism for IIoT to guarantee system security and transaction efficiency. Wan *et al.* [43] introduced a conceptual model to help design the blockchain- and IIoT-based smart factory. Lu *et al.* [44] designed a blockchain-empowered secure data sharing architecture by integrating federated learning in the consensus process.

The above works show that the convergence of blockchain and IIoT may potentially overcome the current drawbacks of IIoT-based IMS. However, the integration of IIoT and blockchain is still in its infancy. How to integrate blockchain and IIoT for DK-DTMC configuration is still challenging due to the complex data flow, information flow, and knowledge flow in DK-DTMC.

C. Motivation

Through the above analyses we have the following three observations: 1) it is significant to develop an IMS, namely, DK-DTMC, in the context of Industry 4.0; 2) the development and configuration of IMS faces the challenges, such as security, distributed collaboration, traceability, etc.; and 3) integrating IIoT with the blockchain is a benefit to construct a secure, traceable, and decentralized network system for IMS, which, however, is still made difficult by the complex data flow, information flow, and knowledge flow in DK-DTMC. Consequently, we combine IIoT with the permissioned blockchain and propose a MBCoT architecture for the configuration of DK-DTMC. In addition, we define hardware infrastructures, software-defined components, and consensus-oriented transaction logic of MBCoT for supporting the autonomous operations of DK-DTMC. The implementation of an MBCoT prototype system and its application examples and evaluation experiment demonstrate that MBCoT could equip IMS with a secure, traceable, stable, and decentralized operating environment, while achieving competitive throughput and latency performance.

III. MBCoT Architecture for DK-DTMC Configuration

This section first defines the concept and system architecture of MBCoT for DK-DTMC configuration. Then, hardware instruments, software-defined components, and consensus-oriented transaction logic of MBCoT are designed.

A. Definition and System Architecture

Definition 1: MBCoT is the fusion of blockchain technology and IIoT, where IIoT builds ubiquitous connections for physical manufacturing resources to the cyber system while blockchain connects software-defined functional nodes of IMS and runs P2P transactions within the cyber system in a secure, traceable, and decentralized way.

Specifically, for DK-DTMC configuration, MBCoT is formalized as

$$\text{MBCoT} ::= \{\text{IIoT}_{\text{PS}}, \text{DS}, \text{VS}, \text{KS}, \text{SS}\} \bowtie B_{\text{IIoT}-\text{DS}-\text{VS}-\text{KS}-\text{SS}} \tag{1}$$

where IIoT_{PS} connects physical resources in PS to the cyber system, PS, DS, VS, KS, and SS correspond to each of five dimensions of DK-DTMC, respectively, and $B_{\text{IIoT}-\text{DS}-\text{VS}-\text{KS}-\text{SS}}$ defines a P2P blockchain network and its transaction logic among these dimensions to support autonomous operations of DK-DTMC.

Based on the above definition, we design a system architecture of MBCoT for DK-DTMC configuration, as shown in Fig. 2. In this architecture, we reconstruct the three-layer and 5-D framework (in functional view) of DK-DTMC into four layers in the configured view, including the device layer, edge layer, cloud layer, and user layer. Each device is connected to its device twin via IIoT, which serves as the transaction listening node to perceive the current or predict future disturbances, such as tool wear and device failure, during the manufacturing process as transaction proposals that are published to the blockchain, while subscribing validated transaction responses from the blockchain to solve the disturbances. Brokers, KM, and computing units at edge collaborate with the process twin,

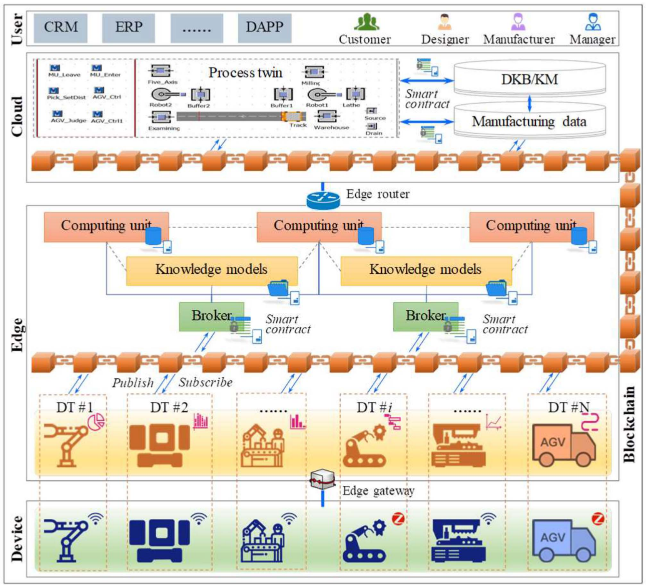


Fig. 2. System architecture of MBCoT for KD-DTMC configuration.

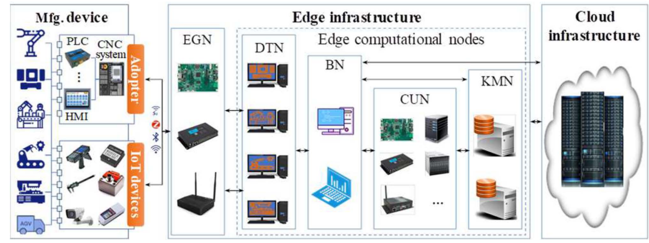


Fig. 3. Hardware infrastructures of MBCoT.

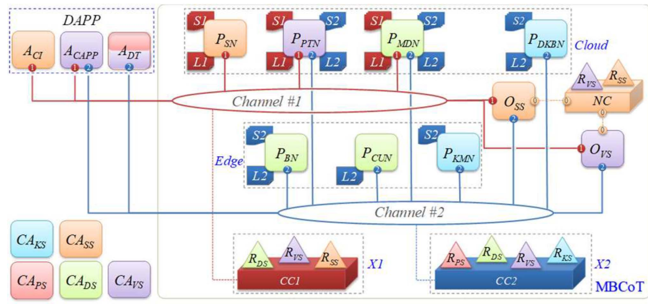


Fig. 4. MBCoT network.

DKB, and KM, manufacturing data at cloud based on smart contracts to realize local or global optimal-state control of DK-DTMC. Here, a broker first subscribes a disturbance of a specific device and decides whether handle it at edge or cloud depending on its time sensitivity and complexity, while solving it by distributing appropriate computing unit and KM based on smart contracts, to locally optimize that device. Meanwhile, the process twin collects many of these disturbances to understand the current manufacturing performance of DK-DTMC via a simulation process and then invokes appropriate KM based on smart contracts to globally optimize the performance of DK-DTMC. In addition, device–edge–cloud resources are packaged as manufacturing services based on the blockchain to serve upper level users in a service-oriented manufacturing manner.

The proposed MBCoT architecture could not only build ubiquitous connections for physical devices to the cyber system via IIoT but also define an autonomous operating mechanism within the cyber system based on the blockchain and smart contracts to support the real-time local or global optimization of DK-DTMC. In addition, attributing to the interconnection of DK-DTMC via MBCoT, DK-DTMC could be taken as a whole to provide smart manufacturing services. This may help manufacturing enterprises better survive from the changing marketplace.

B. Hardware Design

This section introduces the hardware infrastructures of MBCoT, including MDs, edge infrastructures, and cloud infrastructures (as shown in Fig. 3).

MDs include machine tools, industrial robots, automatic guided vehicle (AGV), cutting tools, WIP, etc. The real-time status of MDs is perceived by adopters and IoT devices. Adopters are software components written to parse data collected from CNC systems with different communication protocols. IoT devices are used to collect real-time production data, which are wirelessly connected to edge infrastructures and powered by batteries or energy harvesters.

Edge infrastructures include two kinds of edge nodes, namely, edge gateway nodes (EGNs) and edge computational nodes (ECNs). EGN acts as the entry points for ECN, which preprocesses the perceived data by data filtration and aggregation and forwards them to ECN while receiving comments or control orders from ECN or cloud and transmitting them to the corresponding devices. ECN consists of device twin nodes (DTNs), broker nodes (BNs), computing unit nodes (CUNs), and KM nodes (KMNs). DTN analyzes the preprocessed data via virtual models to find or predict manufacturing disturbances and forward the results to BN. BN assigns the appropriate CUN and KM to make in-situ prompt decisions for time-sensitive tasks to solve the disturbances, or transmits the required data for computing-intensive tasks to the cloud infrastructures for further analysis.

Cloud infrastructures provide powerful computing and storage capacity. DK-DTMC extends resources from the cloud to execute the process twin or KM to optimize the manufacturing performance when tasks are latency tolerant but computing intensive or ECN becomes overloaded. Cloud infrastructures also handle manufacturing service transactions in SS.

C. Software-Defined MBCoT Components

This section introduces the software-defined components of MBCoT based on the permissioned blockchain, which include the MBCoT network, ledger model, and smart contracts.

1) MBCoT Network: As shown in Fig. 4, the MBCoT network is a technical infrastructure that provides ledger (*L1*, *L2*) and smart contract (*S1*, *S2*) services to support autonomous operations of DK-DTMC. Each dimension of DK-DTMC acts as an organization $R \subseteq R = \{R_{\mathrm{PS}}, R_{\mathrm{DS}}, R_{\mathrm{VS}}, R_{\mathrm{KS}}, R_{\mathrm{SS}}\}$ to participate in the network, where its permissions are identified by X.509 certificates [45] in a certificate authority $CA \subseteq$

$CA = \{CA_{\mathrm{PS}}, CA_{\mathrm{DS}}, CA_{\mathrm{VS}}, CA_{\mathrm{KS}}, CA_{\mathrm{SS}}\}$. For example, CA_{PS} dispenses X.509 certificates that can be used to identify participants as belonging to R_{PS}. In addition, R_{SS} and R_{VS} are the network administrators that define policy rules specified in network configuration *NC* to govern the network via ordering service nodes, namely, O_{SS} and O_{VS}.

Specifically, R_{DS}, R_{VS}, and R_{SS} come together as a consortium *X1* to form a P2P social network on channel #1 (C_1) for upper level manufacturing services, where C_1 is governed according to the policy rules specified in channel configuration CC1 and formalized as

$$\boldsymbol{C_1} ::= \{A_{\mathrm{CI}}, \ldots, A_{\mathrm{CAPP}}\} \xleftrightarrow{S1} \{P_{\mathrm{SN}}, P_{\mathrm{PTN}}, P_{\mathrm{MDN}}\} \xleftrightarrow{L1} \{O_{\mathrm{SS}}, O_{\mathrm{VS}}\} \tag{2}$$

where C_1 consists of a set of social nodes $P_{\mathrm{SN}} \in R_{\mathrm{SS}}$, a set of process twin nodes $P_{\mathrm{PTN}} \in R_{\mathrm{VS}}$, and a set of manufacturing data nodes $P_{\mathrm{MDN}} \in R_{\mathrm{DS}}$. DAPP, such as the customer interaction application A_{CI} and computer-aided process planning application A_{CAPP}, are permitted to connect to C_1 based on CA_{SS} and CA_{VS}. In addition, a smart contract set *S1* is used to generate business transactions among DAPP and peer nodes, which are subsequently batched into blocks by O_{SS} or O_{VS} and distributed to every peer node connected to C_1 where they are immutably recorded on their copy of the ledger *L1*.

Similarly, R_{PS}, R_{DS}, R_{VS}, and R_{KS} come together as a consortium *X2* to form a P2P optimal-state control network on channel #2 (C_2) for lower level autonomous operations of DK-DTMC. C_2 is governed by CC2 and formalized as

$$\boldsymbol{C_2} ::= \{A_{\mathrm{CAPP}}, \ldots, A_{\mathrm{DT}}\} \xleftrightarrow{S2} \{P_{\mathrm{BN}}, \ldots, P_{\mathrm{DKBN}}\} \xleftrightarrow{L2} \{O_{\mathrm{SS}}, O_{\mathrm{VS}}\} \tag{3}$$

where C_2 consists of a set of BNs P_{BN}, CUNs P_{CUN} and manufacturing data nodes P_{MDN} belonging to R_{DS}, a set of process twin nodes $P_{\mathrm{PTN}} \in R_{\mathrm{VS}}$, a set of KMNs P_{KMN}, and DKB nodes P_{DKBN} belonging to R_{KS}. DAPP, such as A_{CAPP} and A_{DT} are permitted to connect to C_2 based on CA_{VS} and CA_{PS}. *S2* is used to generate optimal-state control transactions among DAPP and peer nodes, which are subsequently batched into blocks by O_{SS} or O_{VS} and distributed to every peer node to update the copy of the ledger *L2*.

2) Ledger Model: The ledger represents objects, namely, manufacturing orders (MOs) on C_1 and MDs on C_2, by a set of facts about the states of the objects, and immutably records transactions which update the states of objects. A ledger is held and maintained by distributed peer nodes within the MBCoT network. The ledger model contains two distinct pieces—a *word state* that holds a cache of the current value of states (as shown in the top of Fig. 5), and a *blockchain* that immutably records the history of all transactions within the network (as shown in the bottom of Fig. 5) that resulted in the current word state. Here, the *word state* is implemented as a database based on LevelDB or CouchDB for the efficient storage and retrieval of states. The *blockchain* is structured as a sequential log of interlinked blocks, where each block collects a sequence of transactions and is appended to the chain based on its hash value. A block is defined as a triple $B_i = \{H_i, D_i, M_i\}$.

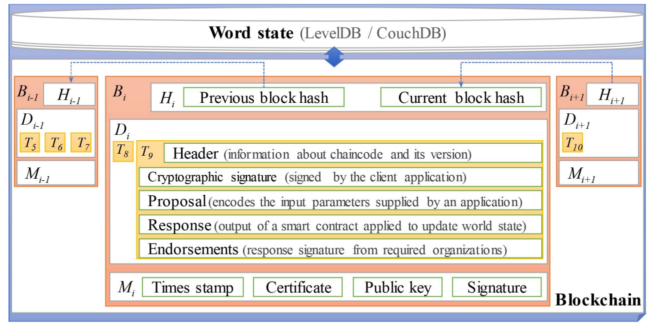


Fig. 5. Ledger model for MBCoT.

A block header H_i contains the block number, current block hash, and previous block hash that are written when a block is created by cryptographically hashing the block data D_i, to make every block be inextricably linked to its neighbor. D_i collects a sequence of transactions written by O_{SS} or O_{VS}, where each transaction t_i consists of a header, a cryptographic signature signed by DAPP, a proposal supplied by DAPP as the transaction input, responses as the transaction output, and endorsements collecting response signature from the required organizations. The block metadata M_i contains a timestamp indicating the time when the block was written, as well as the certificate, public key, and signature of the block writer. Since *blockchain B* is the standard for a ledger model (only the transaction content is different based on the specific word state and transaction logic), we customize word states W_1 and W_2 capturing a set of facts about the current states of an MO for *L1* on C_1 and an MD for *L2* on C_2. W_1 and W_2 are defined as

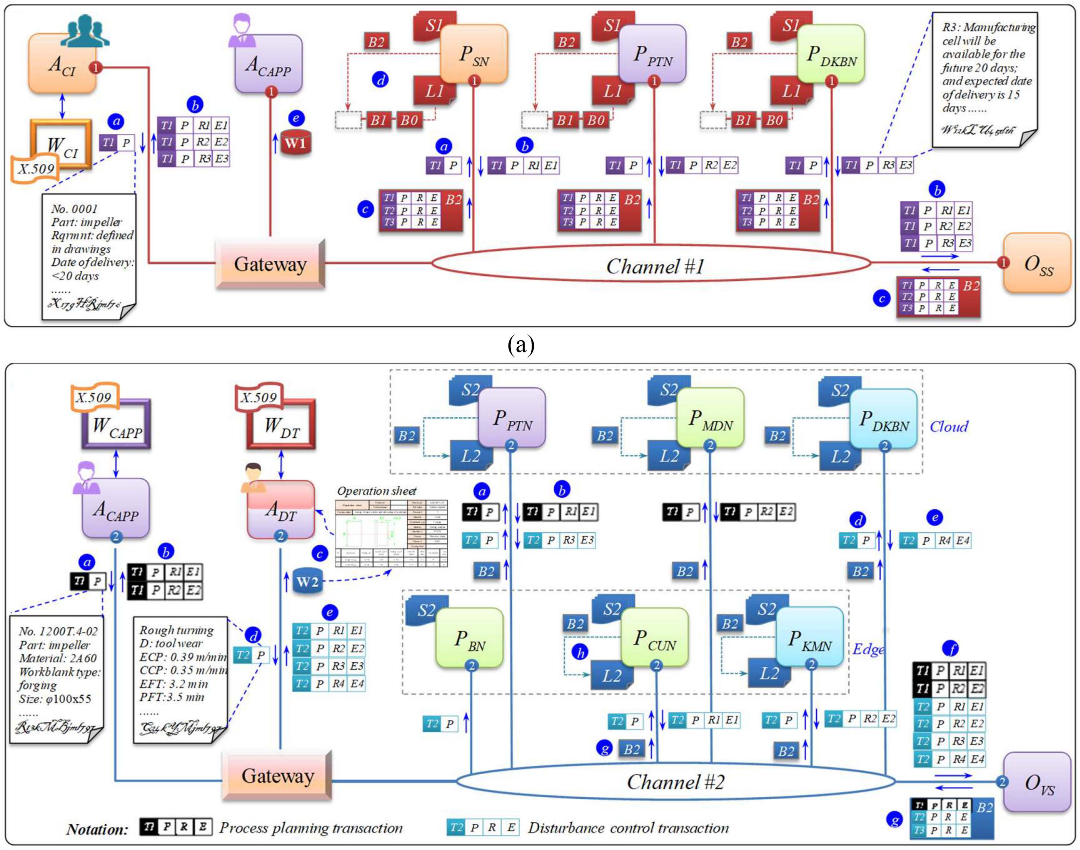
$$\boldsymbol{W_1} = \left\{\left\{\cup_i \left\langle K_i^{\mathrm{MO}}, V_i^{\mathrm{MO}}\right\rangle\right\}, \left\langle K^{\mathrm{MC}}, V^{\mathrm{MC}}\right\rangle\right\} \tag{4}$$

$$\boldsymbol{W_2} = \cup_j \left\langle K_j^{\mathrm{MD}}, V_j^{\mathrm{MD}}\right\rangle \tag{5}$$

where W_1 contains two kinds of key–value $\langle K, V\rangle$ pairs, namely, $\langle K_i^{\mathrm{MO}}, V_i^{\mathrm{MO}}\rangle$ and $\langle K^{\mathrm{MC}}, V^{\mathrm{MC}}\rangle$, capturing the current states of the ith MO and the whole manufacturing cell (MC), respectively. Here, K_i^{MO} indicates the order number of MO_i, and V_i^{MO} represents a set of properties for MO_i, including order content, order requirements, date of delivery, current state, etc. K^{MC} indicates the identifier of MC, and V^{MC} represents a set of properties of MC, including its basic information, utilization, capability, current status, etc. Each $\langle K_j^{\mathrm{MD}}, V_j^{\mathrm{MD}}\rangle$ in $\boldsymbol{W}_2$ represents the current states of the jth MD, where K_j^{MD} indicates the device identifier of MD_i and V_j^{MD} consists of a set of properties about MD_i, including device information, device utilization, device capability, current status, etc. In addition, each validated transaction results in changing values of a key–value pair to update the ledger.

3) Smart Contracts: Smart contracts define a series of rules between different dimensions of DK-DTMC in executable codes to facilitate the autonomous operations of DK-DTMC. To this end, two kinds of smart contract sets, namely, *S1* and *S2*, are defined for C_1 and C_2, respectively.

S1 consists of five smart contracts to govern the upper level manufacturing services in SS, including the value evaluation contract (VEC), capacity evaluation contract (CEC), availability evaluation contract (AEC), state update contract (SUC), and



(a)

(b)

Fig. 6. Consensus-oriented transaction logic. (a) Social transaction logic on C_1. (b) Optimal-state control transaction logic on C_2.

state query contract (SQC). For a new MO proposal supplied by a customer, VEC, CEC, and AEC evaluate the business value of the order, capacity, and availability of MC, and decide to accept the order only when all three smart contracts output positive responses. During the order manufacturing process, SUC updates the states of MO and MC with the process twin, which could be accessed by customers based on SQC.

S2 also contains five smart contracts to govern the lower level optimal-state control of DK-DTMC, including the process plan validation contract (PPVC), process plan optimization contract (PPOC), disturbance decision support contract (DDSC), disturbance handle contract (DHC), and disturbance evaluation contract (DEC). A_{CAPP} generates a theoretical process plan as transaction proposal to $\boldsymbol{C}_2$, where PPVC first validates and transforms the plan into the practical operations by invoking $\boldsymbol{W}_2$ to check the feasibility and availability of machine tools, cutting tools, and cutting parameters. Then, PPOC outputs the optimal plan, which utilizes PT to visually simulate, analyze, and optimize the plan considering manufacturing time and cost. The details of the above process could be found in [14]. The optimized plan and detailed operations are taken as the input for lower level optimal-state control of DK-DTMC, where each A_{DT} perceives the status of MD to understand the current disturbance or predict a potential future disturbance that is published as the transaction proposal to $\boldsymbol{C}_2$, while subscribing transaction responses to solve the disturbance based on DDSC, DHC, and DEC. DDSC determines whether to handle the disturbance at edge or cloud. DHC decides which KM at which CUN is used, and generates responses to locally optimize that DM. DEC subscribes many of these disturbances to evaluate the current and optimize the future performance of DK-DTMC with the process twin and appropriate KM.

D. Consensus-Oriented Transaction Logic

The consensus is the process of synchronizing transactions across the MBCoT network, where a consensus algorithm embedded in O_{SS} and O_{VS} is used to ensure that the required peer nodes have reached an agreement on the content and order of transactions. MBCoT employs a permissioned blockchain as its base architecture, which operates with a set of known, identified, and often vetted peer nodes. Therefore, MBCoT could use a simple crash fault-tolerant (CFT) consensus protocol to govern the transaction process. CFT does not require costly mining like PoW, thus achieving higher throughput and lower latency. Based on CFT, Fig. 6 defines two consensus-oriented transaction logics for MBCoT, namely, social transaction logic on C_1 and optimal-state control transaction logic on C_2.

As shown in Fig. 6(a), a social transaction starts with a transaction *T1* proposal *P*, which describes the information and requirements of MO supplied by a customer via A_{CI}. A_{CI} selects an identifier from a wallet W_{CI} identifying the role and right of the customer to participate on $\boldsymbol{C}_1$ through the gateway.

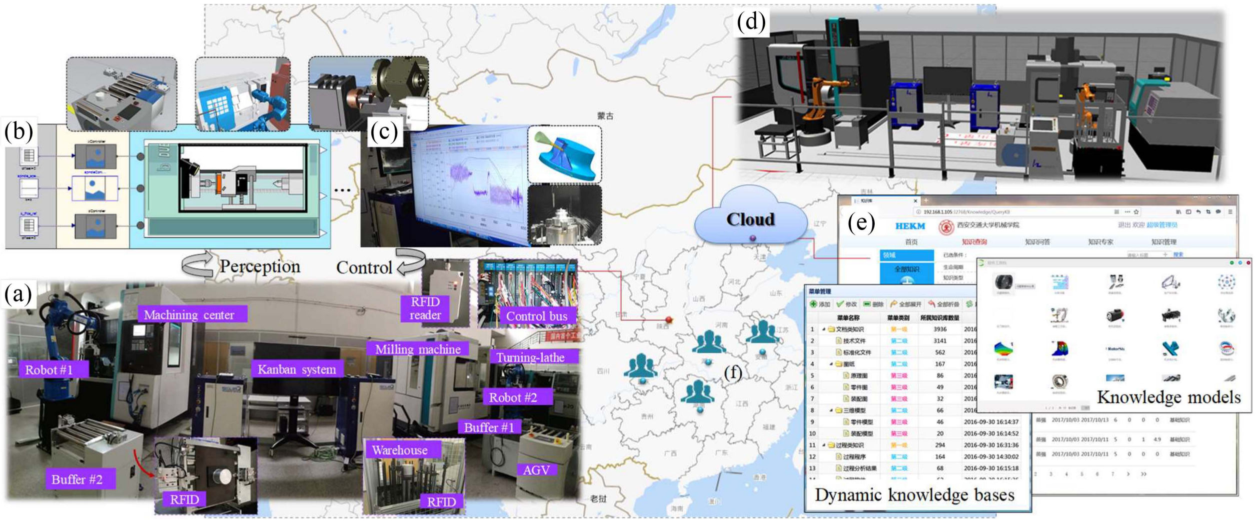


Fig. 7. Scenario of the testbed. (a) Mfg. and IoT devices; (b) turning tool twin and (c) DT-based real-time disturbance understanding and prediction at edge; (d) process twin and (e) DKB and KM at cloud; (f) customers at different places.

A_{CI} sends P signed by the customer based on its private key to peer nodes P_{SN}, P_{PTN}, and P_{MDN}. P_{SN} evaluates the business value of P by executing VEC in *S1* and generates response *R1* with the endorsement *E1* signed based on the private key of P_{SN}. P_{PTN} and P_{MDN} evaluate the capacity and availability of DK-DTMC by executing CEC and AEC, which generates *R2* with *E2* and *R3* with *E3*, respectively. A_{CI} receives endorsed responses and decides to continue or terminate *T1* at this time. If all the endorsed responses and feedback of A_{CI} are positive, a new transaction *T1* is generated and passed to the ordering service. O_{SS} and O_{VS} collect many these transactions, packages them into a new block, and distributes it to peer nodes. Peer nodes validate and process the block, which results in a new block being added to *L1* on P_{SN}, P_{PTN}, and P_{MDN} at the same way, leading to an immutable ledger. Once this process is complete, each peer node may inform the connected DAPP, such as A_{CI} and A_{CAPP} that *T1* has been processed. During the order manufacturing process, customers of A_{CI} could query the status of the order by executing SQC on P_{MDN} or visualize its manufacturing process by executing SQC on P_{PTN}.

Fig. 6(b) shows a similar transaction logic for the optimal-state control of DK-DTMC, whose key stages are as follows. A_{CAPP} takes the informed complete social transaction as the input to generate theoretical plans as transaction *T1* proposal *P*, which is sent to P_{MDN} and P_{PTN}. P_{MDN} validates and transforms the plan into practical operations by executing PPVC and generates *R1* with *E1*. Then, P_{PTN} visually simulates, analyzes, and optimizes the plan considering manufacturing time and cost by executing PPOC and generates *R2* with *E2*. Once *T1* is generated, P_{PTN} or P_{MDN} may inform A_{DT} with the optimized plan and operations to guide the manufacturing process in PS. During the manufacturing process, each A_{DT} generates a disturbance as the transaction *T2* proposal *P* sent to P_{BN}. P_{BN} determines whether to handle the disturbance at edge or cloud by executing DDSC and generates *R1* with *E1*. If *R1* indicates handling at edge, P_{BN} further decides which KM at which CUN is used by executing DHC, where P_{CUN} and P_{KMN} generate R2 with E2 and R3 with E3 respectively. If not, P_{DKBN} invokes and executes an appropriate KM based on DHC, and generates R4 with E4. A_{DT} is informed with these endorsed responses to locally optimize that MD. In parallel, P_{PTN} subscribes many of these disturbances to evaluate the current and optimize the future performance of DK-DTMC by executing DEC. In addition, O_{SS} and O_{VS} collect these transactions and package them into a new block to update *L2*.

IV. Prototype Implementation and Evaluation

Hyperledger Fabric is an opensource enterprise-grade-permissioned blockchain with identifiable participants, high transaction throughput performance, and low latency of transaction confirmation while keeping privacy and confidentiality of transactions and data pertaining to enterprise business. To this end, the Fabric is employed to construct the prototype system in this article.

A. Prerequisites of the Testbed

Fig. 7 illustrates the prerequisites of the testbed, which involves MC, DT deployed at the edge, process twin, DKB, KM, and distributed customers. Fig. 7(a) shows the layout and building blocks of the MC, which consists of three machine tools, two 6-axis articulated robots, two buffers, a warehouse, an AGV, and a Kanban system. In addition, IoT devices are used to perceive the real-time status of MC. For example, position status of cutting tools and WIP is tracked by RFID; sensors, such as Hall sensors, liquid-level sensors, and multispeed revolvers, deployed on the machine tools and robots perceive the linear displacement, angular displacement, velocity, pressure, and temperature of these devices. Fig. 7(b) shows an example of DT, namely, turning tool twin, that is constructed using Modelica. As shown in Fig. 7(c), DT is a complete mapping of its corresponding device that could understand the current or predict the future disturbances based on the real-time data perceived by IoT devices, while locally optimizing the performance of that device with the appropriate KM. PT [Fig. 7(d)], DKB and KM [Fig. 7(d)] are

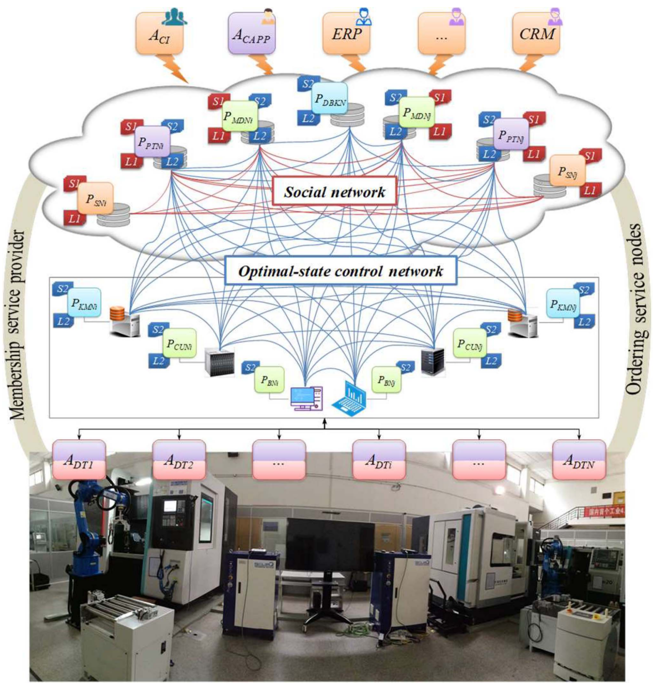
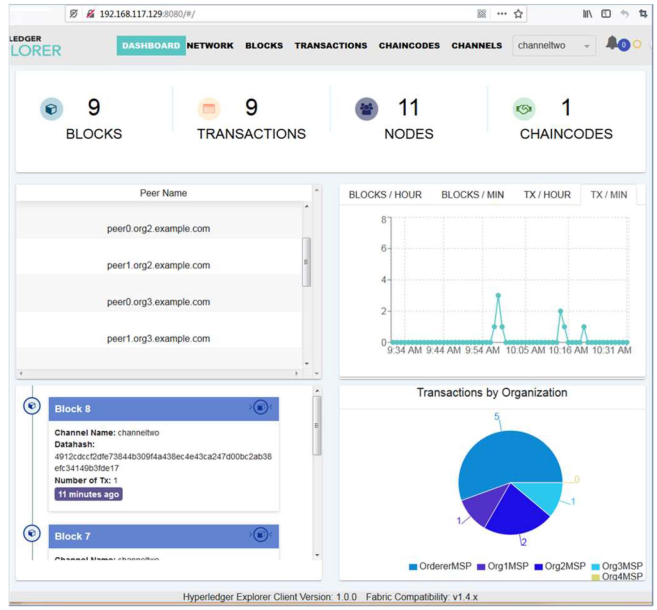


Fig. 8. Architecture of the MBCoT prototype system.



(a)

Block Details

Channel name:	channeltwo
Block Number	8
Created at	2020-06-03T02:20:43.000Z
Number of Transactions	1
Block Hash	db7212537d710c18350a951813536e41b657222e21316573d439cd2b6dc4574f
Data Hash	4912cdccf2dfe73844b309f4a438ec4e43ca247d00bc2ab38efc34149b3fde17
Prehash	a1b40ea0588b551515d9183f0f7e1307f4a30f4e0780dba148b174d7c296392a

(b)

Fig. 9. (a) MBCoT prototype system and (b) block example visualized by Hyperledger Explorer.

jointly used to analyze the disturbances and globally optimize the lower level performance of MC. On the other hand, they also support upper level service-oriented manufacturing among enterprises and distributed customers [Fig. 7(e)]. The construction details of DT and process twin could be found in our previous works [10], [11]. The implementation of DKB and KM are introduced in [15] and [46].

B. Prototype System

Based on the building blocks of the testbed, this section constructs an MBCoT prototype system using Fabric. Fig. 8 shows the overall architecture of the prototype system. Here, Fabric connects SN, PTN, and MDN with customers to form a social network for upper level smart manufacturing services. Fabric also connects functional nodes with enterprise information systems and DAPP to form an optimal-state control network for lower level autonomous operations of DK-DTMC.

Specifically, the MBCoT prototype system is comprised of seven pieces, including networks, channels, blocks, transactions, chaincodes, membership service provider (MSP), and ordering service. Fig. 9(a) shows an example of the Web front end of the optimal-state control network on channel C_2, which contains nine blocks, nine transactions, 11 nodes and one chaincode (namely, $S2$ on C_2) so far. Here, 11 nodes correspond to the peer nodes within the optimal-state control network defined in Fig. 8. As shown in Fig. 9(b), each block collects at least one transaction (20 transactions at most) that records process planning or manufacturing events and connects with other blocks based on their hash values to form $L2$. Here, the block size is set to 100 MB and CouchDB is used for word state storage and retrieval. Chaincode $S2$ packages and manages smart contracts that are written in java language. $S2$ runs within Docker for isolation and is responsible for handling transactions in the optimal-state control network. At the back end, MSP manages CAs of different dimensions of DK-DTMC and wallets of DAPP, which is responsible for associating participants in the network with cryptographic identities. Ordering service establishes consensus on the order of transactions by collecting transactions, packaging them into a block and broadcasting the block to peer nodes.

C. Application Scenarios

Fig. 10 illustrates the application scenarios of the prototype system, including smart manufacturing services and manufacturing process visualization in the social network, intelligent process planning, and optimal-state control in the optimal-state control network. The prototype system autonomically handles transactions for each application scenario with five standard procedures. First, distributed peer nodes deal with transaction proposals from DAPP by executing smart contracts in chaincode to generate responses with endorsements. Second, DAPP receives enough responses with endorsements from the required peer nodes and confirms their effectiveness. Third, ordering service nodes collect effective transactions with endorsements, package them into new blocks, and broadcast them to the network. Fourth, ordering service nodes also deliver new blocks to peer nodes. Finally, each peer node validates the transactions in new blocks, resulting in adding these

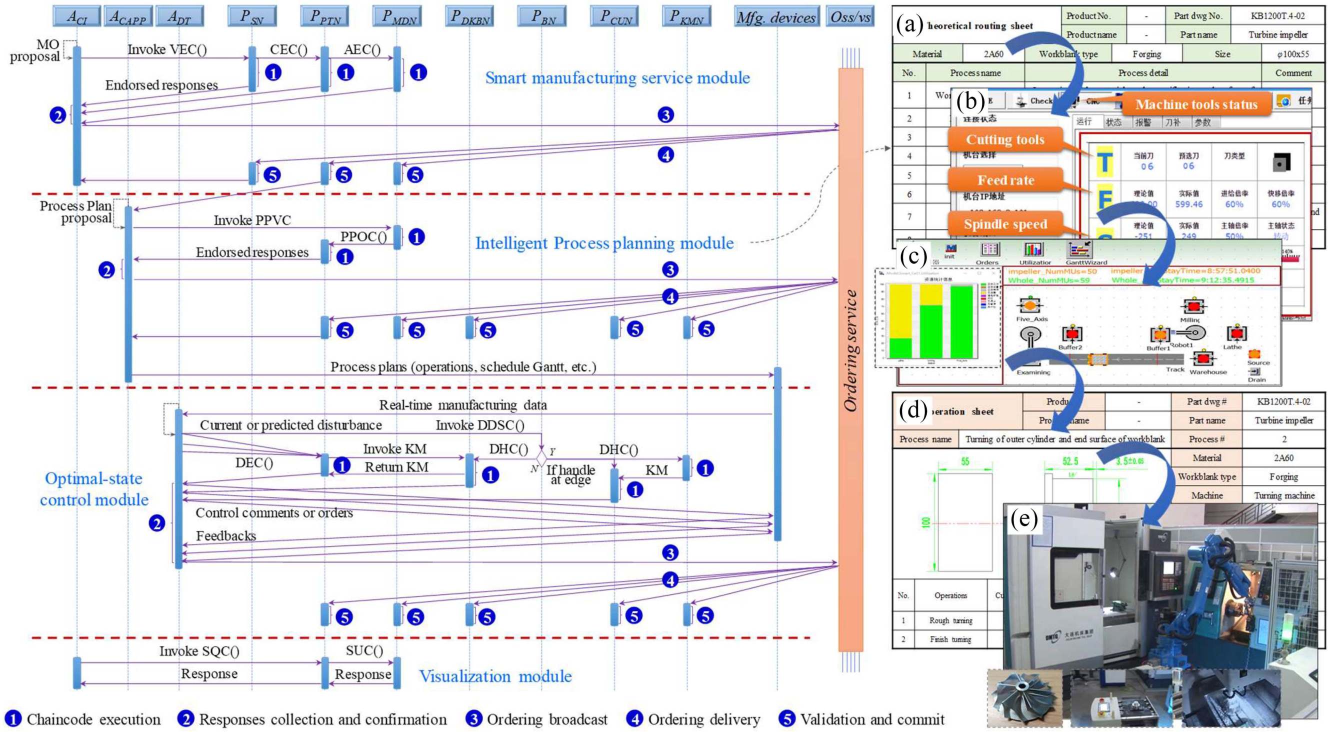


Fig. 10. Prototype system transaction flow (left) and application example (right). (a) Theoretical process plan supplied by A_{CAPP} as a transaction proposal. (b) Process plan validation by executing PPVC to check the capacity and availability of mfg. devices. (c) Process plan optimization by executing PPOC to invoke PT and its embedded algorithms. (d) Optimized plan and practical operations as endorsed response to guide. (e) Manufacturing process at PS.

blocks to its copy of the ledger and committing to the network to inform DAPP.

Specifically, the smart manufacturing service module handles business transactions between customers and manufacturing enterprises via *S1*. The intelligent process planning module takes the requirements of a validated business transaction as input and generates process plans based on PPVC and PPOC in *S2* to guide the manufacturing process. The optimal-state control module deals with disturbances perceived or predicted by DT in a distributed environment. Here, each disturbance of the device is solved at edge or cloud immediately based on DDSC and DHC in *S2* to locally optimize the performance of that device. At the same time, PTN collects and simulates many of these disturbances to understand and evaluate their influence on the performance of DK-DTMC, so as to globally optimize its performance based on DEC in *S2*.

The right part of Fig. 10 illustrates an example of the intelligent process planning module. A_{CAPP} is informed by P_{PTN} with a processed business transaction whose requirements are taken as the input for the deep learning model [14] of A_{CAPP}. A_{CAPP} generates a theoretical process plan [Fig. 10(a)] as the transaction proposal and transmits it to P_{MDN} and P_{PTN}. P_{MDN} executes PPVC to validate the capacity and availability of machine tools, cutting tools, etc., by invoking historical or real-time manufacturing data [Fig. 10(b)]. P_{MDN} transforms the theorical process plan into practical operations as transaction response, to which PPOC in P_{PTN} is carried out to optimize the parameters of each operation and sequence of all operations with manufacturing time and cost as objectives using PT and its embedded algorithms [Fig. 10(c)]. A_{CAPP} receives the optimized plan [Fig. 10(d)], confirms its effectiveness and distributes it to PS to guide manufacturing process [Fig. 10(e)].

D. Discussion

As the fusion of IIoT and blockchain, MBCoT enjoys significant advantages in communicating security, operating stability, and implementing costs when compared with traditional IIoT. This attributes to the application of cryptography and fault-tolerant mechanism in as well as the decentralized deployment architecture of the permissioned blockchain. In addition, MBCoT could not only establish ubiquitous connections between physical devices but also connect software-defined functional nodes and business nodes to form a manufacturing network supporting autonomous operations of DK-DTMC. From a technological view, the permissioned blockchain could empower DK-DTMC with identifiable participants, a secure and trackable communication mechanism and autonomous distributed decision-making environment, while achieving competitive throughput and latency. To illustrate the effectiveness of the permissioned blockchain, namely, Fabric used in MBCoT, we compare the performance of Fabric with a permissionless blockchain—Ethereum in terms of throughput and latency. To this end, two test nets, namely, Ethereum test net—Ropsten with PoW as its consensus algorithm and Fabric test net—Fabcoin with CFT as its consensus prototype are employed. The reason for employing these test nets is that they come the closest to the performance of the actual Ethereum and Fabric blockchains. The experimental environment is set as in [47] for Ropsten and as in [17] for Fabcoin. Fig. 11 illustrates their performance on

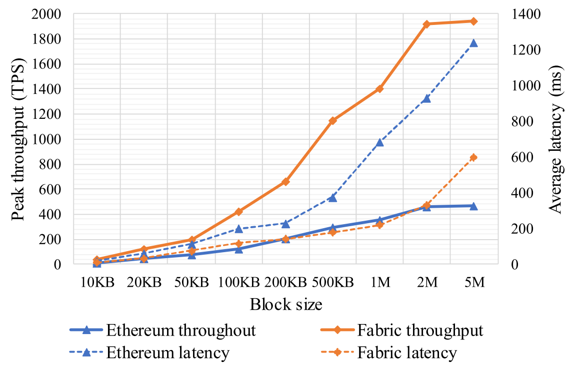


Fig. 11. Throughput and latency performance comparison between Ethereum and Fabric.

peak throughput measured at peer nodes and average peer-to-peer latency with changing the block size. The results show that the Fabric blockchain achieves higher throughput and lower latency than Ropsten for all cases. Fabric obtains the best throughput of about 2000 transactions per second (TPSs) with acceptable latency of roughly 300 ms when the block size is 2 MB. This case demonstrates that the permissioned blockchain such as Fabric is more applicable for MBCoT.

Although MBCoT obtains many significances, such as security, traceability, and stability for configuring DK-DTMC, there are still several potential limitations and challenges for industrial implementation. Transactions in the manufacturing process are complex as much information, such as disturbance reasons and solutions, is included in each transaction, leading to the usage of a larger block to contain transactions. However, increasing block size results in worse latency performance, which may be unacceptable for real-time optimization and control tasks. In addition, since disturbances in the manufacturing process are stochastic and diversiform, how to establish an extendible and realizable smart contract to handle the disturbances is still challenging. Fortunately, according to the current experiences, incorporating machine learning algorithms, such as deep Q-Learning [34] and federated learning [44], into consensus process or smart contracts, may solve the above issues. Therefore, future directions related to MBCoT will focus on the optimization of the consensus algorithm to achieve better latency performance when increasing the block size, and also the establishment of a smart contract model to better handle manufacturing disturbances, by combining with the state-of-the-art machine learning approaches.

V. Conclusion

MBCoT integrates IIoT with the permissioned blockchain for configuring DK-DTMC toward Industry 4.0. The result was a secure, tracible, stable, and decentralized IMS network connecting three-layer and 5-D of DK-DTMC to support upper level smart manufacturing service and lower level optimal-state control of the manufacturing process. MBCoT was explained based on its system architecture, hardware infrastructure, software-defined components, including MBCoT network, ledger model, and smart contracts, and consensus-oriented transaction logic. Based on the results presented in this article, several contributions were of significance. First, a novel MBCoT architecture for the configuration of DK-DTMC was proposed by integrating IIoT with the permissioned blockchain. MBCoT enjoyed significant advantages in communicating security and traceability, and operating stability and implementing cost while achieving competitive throughput and latency performance. Second, the design of hardware infrastructures, software-defined components, and transaction logic provided an insight into the industrial implementation of MBCoT. Third, the prototype system and its application examples strengthen the feasibility of MBCoT for DK-DTMC configuration. The evaluation experiment demonstrated that the permissioned blockchain could achieve higher throughput and lower latency than the permissionless blockchain, which was more resource efficient and applicable for MBCoT.

References

[1] X. Chen, S. Zhang, and J. M. P. Geraedts, "Guest editorial focused section on sensing and perception systems for intelligent manufacturing (SPIM)," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 983–985, Jun. 2018.

[2] G. Xue and Y. Xia, "A flexible logistics system for intelligent manufacturing workshop," in *Proc. IEEE ICMIMT*, Cape Town, South Africa, 2019, pp. 133–137.

[3] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent manufacturing in the context of industry 4.0: A review," *Engineering*, vol. 3, no. 5, pp. 616–630, Oct. 2017.

[4] K. Ding, F. T. S. Chan, X. Zhang, G. Zhou, and F. Zhang, "Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors," *Int. J. Prod. Res.*, vol. 57, no. 20, pp. 6315–6334, Oct. 2019.

[5] J. Leng *et al.*, "Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model," *Robot. Comput. Int. Manuf.*, vol. 63, Jun. 2020, Art. no. 101895.

[6] S. Jeong, W. Na, J. Kim, and S. Cho, "Internet of Things for smart manufacturing system: Trust issues in resource allocation," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4418–4427, Dec. 2018.

[7] A. V. Barenji, Z. Li, W. M. Wang, G. Q. Huang, and D. A. Guerra-Zubiaga, "Blockchain-based ubiquitous manufacturing: A secure and reliable cyber-physical system," *Int. J. Prod. Res.*, vol. 58, no. 7, pp. 2200–2221, Apr. 2020.

[8] C. Lee, N. Sung, L. Nkenyereye, and J. Song, "Blockchain enabled Internet-of-Things service platform for industrial domain," in *Proc. IEEE ICII*, Seattle, WA, USA, 2018, pp. 177–178.

[9] S. Garg, K. Kaur, G. Kaddoum, and K. R. Choo, "Towards secure and provable authentication for Internet of Things: Realizing industry 4.0," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4598–4606, May 2020.

[10] G. Zhou, C. Zhang, Z. Li, K. Ding, and C. Wang, "Knowledge-driven digital twin manufacturing cell towards intelligent manufacturing," *Int. J. Prod. Res.*, vol. 58, no. 4, pp. 1034–1051, Feb. 2020.

[11] C. Zhang, G. Zhou, J. He, Z. Li, and W. Cheng, "A data-and knowledge-driven framework for digital twin manufacturing cell," *Procedia CIRP*, vol. 83, pp. 345–350, May 2019.

[12] K. Zhang *et al.*, "Digital twin-based opti-state control method for a synchronized production operation system," *Robot. Comput. Int. Manuf.*, vol. 63, Jun. 2020, Art. no. 101892.

[13] A. Angrish, B. Starly, Y.-S. Lee, and P. H. Cohen, "A flexible data schema and system architecture for the virtualization of manufacturing machines (VMM)," *J. Manuf. Syst.*, vol. 45, pp. 236–247, Oct. 2017.

[14] C. Zhang, G. Zhou, J. Hu, and J. Li, "Deep learning-enabled intelligent process planning for digital twin manufacturing cell," *Knowl. Based. Syst.*, vol. 191, Mar. 2020, Art. no. 105247.

[15] C. Zhang, G. Zhou, Q. Lu, and F. Chang, "Graph-based knowledge reuse for supporting knowledge-driven decision-making in new product development," *Int. J. Prod. Res.*, vol. 55, no. 23, pp. 7187–7203, Dec. 2017.

[16] K. Ding, P. Jiang, and S. Su, "RFID-enabled social manufacturing system for inter-enterprise monitoring and dispatching of integrated production and transportation tasks," *Robot. Comput. Int. Manuf.*, vol. 49, pp. 120–133, Feb. 2018.

[17] E. Androulaki *et al.*, "Hyperledger fabric: A distributed operating system for permissioned blockchains," Jan. 2018. [Online]. Available: arxiv.10.1145/3190508

[18] T. Mitani and A. Otsuka, "Traceability in permissioned blockchain," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Atlanta, GA, USA, 2019, pp. 286–293.

[19] J. Leng *et al.*, "Makerchain: A blockchain with chemical signature for self-organizing process in social manufacturing," *J. Clean. Prod.*, vol. 234, pp. 767–778, Oct. 2019.

[20] M. Zhaofeng, W. Xiaochang, D. K. Jain, H. Khan, G. Hongmin, and W. Zhen, "A blockchain-based trusted data management scheme in edge computing," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 2013–2021, Mar. 2020.

[21] S. Geissler, T. Prantl, S. Lange, F. Wamser, and T. Hoßfeld, "Discrete-time analysis of the blockchain distributed ledger technology," in *Proc. ITC*, Budapest, Hungary, 2019, pp. 130–137.

[22] M. Shurman, A. A. Obeidat, and S. A. Al-Shurman, "Blockchain and smart contract for IoT," in *Proc. ICICS*, Irbid, Jordan, 2020, pp. 361–366.

[23] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1432–1465, 2nd Quart., 2020.

[24] M. Singh, A. Singh, and S. Kim, "Blockchain: A game changer for securing IoT data," in *Proc. IEEE WF-IoT*, Singapore, 2018, pp. 51–55.

[25] A. Marsalek and T. Zefferer, "A correctable public blockchain," in *Proc. IEEE TrustCom/BigDataSE*, Rotorua, New Zealand, 2019, pp. 554–561.

[26] K. Gai, Y. Wu, L. Zhu, L. Xu, and Y. Zhang, "Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7992–8004, Mar. 2019.

[27] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[28] I. G. A. K. Gemeliarana and R. F. Sari, "Evaluation of proof of work (POW) blockchains security network on selfish mining," in *Proc. ISRITI*, Yogyakarta, Indonesia, 2018, pp. 126–130.

[29] Q. Li, H. Li, Z. Wen, and P. Yuan, "Research on the P2P sybil attack and the detection mechanism," in *Proc. IEEE ICSESS*, Beijing, China, 2018, pp. 668–671.

[30] M. Urbina, T. Acosta, J. Lázaro, A. Astarloa, and U. Bidarte, "Smart sensor: SoC architecture for the industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6567–6577, Aug. 2019.

[31] B. Mayer, D. Tantscher, and C. Bischof, "From digital shop floor to real-time reporting: An IIoT based educational use case," *Procedia Manuf.*, vol. 45, pp. 473–478, Jan. 2020.

[32] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4724–4734, Nov. 2018.

[33] S. Munirathinam, "Chapter Six–Industry 4.0: Industrial Internet of Things (IIOT)," in *Advance Computing*, P. Raj and P. Evangeline, Eds. Amsterdam, The Netherlands: Elsevier, 2020, pp. 129–164.

[34] C. Qiu, F. R. Yu, H. Yao, C. Jiang, F. Xu, and C. Zhao, "Blockchain-based software-defined industrial Internet of Things: A dueling deep *Q*-learning approach," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4627–4639, Jun. 2019.

[35] J. Leng *et al.*, "ManuChain: Combining permissioned blockchain with a holistic optimization model as bi-level intelligence for smart manufacturing," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 1, pp. 182–192, Jan. 2020.

[36] J. Sengupta, S. Ruj, and S. D. Bit, "A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT," *J. Netw. Comput. Appl.*, vol. 149, Jan. 2020, Art. no. 102481.

[37] O. Novo, "Blockchain meets IoT: An architecture for scalable access management in IoT," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1184–1195, Dec. 2018.

[38] L. Bai, M. Hu, M. Liu, and J. Wang, "BPIIoT: A light-weighted blockchain-based platform for industrial IoT," *IEEE Access*, vol. 7, pp. 58381–58393, 2019.

[39] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial Internet of Things," *IEEE Trans. Ind. Inf.*, vol. 15, no. 6, pp. 3571–3581, Jun. 2019.

[40] G. Rathee, A. Sharma, R. Kumar, and R. Iqbal, "A secure communicating things network framework for industrial IoT using blockchain technology," *Ad Hoc Netw.*, vol. 94, Nov. 2019, Art. no. 101933.

[41] S. He, W. Ren, T. Zhu, and K. R. Choo, "BoSMoS: A blockchain-based status monitoring system for defending against unauthorized software updating in industrial Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 948–959, Feb. 2020.

[42] J. Huang, L. Kong, G. Chen, M. Wu, X. Liu, and P. Zeng, "Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3680–3689, Jun. 2019.

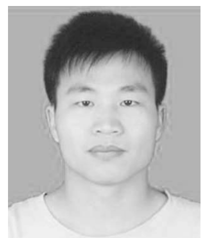
[43] J. Wan, J. Li, M. Imran, D. Li, and A. Fazale, "A blockchain-based solution for enhancing security and privacy in smart factory," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3652–3660, Jun. 2019.

[44] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.

[45] R. Martinez, A. Alejo, P. Portugal, A. Cuno, F. Zapata, and R. Saavedra, "An artifact for X.509 digital certificates delivery," in *Proc. SCCC*, Concepcion, Chile, 2019, pp. 1–8.

[46] C. Zhang, G. Zhou, Q. Bai, Q. Lu, and F. Chang, "HEKM: A high-end equipment knowledge management system for supporting knowledge-driven decision-making in new product development," in *Proc. ASME IDETC/CIE*, 2018, Art. no. V01BT02A014.

[47] A. Angrish, B. Craver, M. Hasan, and B. Starly, "A case study for blockchain in manufacturing: 'FabRec': A prototype for peer-to-peer network of manufacturing nodes," *Procedia Manuf.*, vol. 26, pp. 1180–1192, Jan. 2018.



Chao Zhang received the B.E. degree in mechanical engineering and automation from Sichuan University, Chengdu, China, in 2015. He is currently pursuing the Ph.D. degree in mechanical engineering with the School of mechanical engineering, Xi'an Jiaotong University, Xi'an, China.

He has authored a monograph, 20 journal and conference papers, and ten inventions. His research interests include intelligent manufacturing systems, knowledge-driven decision making, and deep learning.

Guanghui Zhou (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in mechanical engineering from Xi'an Jiaotong University, Xi'an, China, in 1996, 1999, and 2003, respectively.

From 2010 to 2011, he was a Visiting Scholar with the Department of Industrial and Manufacturing Systems Engineering, Florida State University, Tallahassee, FL, USA. He is currently a Professor with the School of Mechanical Engineering, Xi'an Jiaotong University. He has authored two monographs, more than 120 journal and conference papers, and 20 inventions. His research interests include low-carbon manufacturing, intelligent manufacturing, and knowledge-based manufacturing systems.

Prof. Zhou research results won the first prize of the Natural Science of the Ministry of Education of China.

Han Li received the B.S. degree in mechanical engineering and automation from Sichuan University, Chengdu, China, in 2018. He is currently pursuing the M.S. degree in mechanical engineering with Xi'an Jiaotong University, Xi'an, China.

His research interests include intelligent manufacturing and digital twin technology.

Yan Cao received the B.S. degree in mechanical design and manufacturing and the M.S. degree in mechanical manufacturing from Shandong University of Technology, Shandong, China, in 1992 and 1995, respectively, and the Ph.D. degree in mechanical engineering from Xi'an Jiaotong University, Xi'an, China, in 1999.

He is currently a Professor with the School of Mechatronic Engineering, Xi'an Technological University, Xi'an. He has authored more than 150 books, 300 articles, and 80 patents. His research interests include digital-integrated manufacturing, intelligent design and manufacturing, virtual prototype development and performance analysis, and processing technology analysis, and cutting mechanism research of high-performance materials and special structures.