# Topology-Aware Adaptive Inspection for Fraud in I4.0 Supply Chains

Thomas Welsh, *Member, IEEE*, Faeq Alrimawi, Ali Farahani, Diane Hassett, Andrea Zisman, and Bashar Nuseibeh

***Abstract*—Supply chain fraud involving counterfeit or adulterated products presents threats to human health and safety. Quality inspection is a key fraud mitigation tool where inspection planning involves allocating inspection resources across geographically dispersed assets considering both the cost and value of the inspection. I4.0 environments pose further challenges as their heterogeneous and dynamic cyber-physical environment creates a large inspection resource allocation solution space, causing the corresponding analysis to be computationally complex. In this article, we contribute to supporting optimal inspection decisions of dynamic cyber-physical supply chains through the use of structural representations—*topologies* of the supply chain, physical premises, and their production context. We present an approach for topology modeling of supply chains and illustrate its use within an adaptive inspection approach, showing that structural information can reduce malicious process discovery times by up to 90%.**

***Index Terms*—Adaptive, fraud, I4.0, inspection, supply chains, topology.**

## I. Introduction

RECENTLY, a number of global events including the COVID19 pandemic and trade wars have highlighted how essential supply chains are for the functioning of modern society. Unfortunately, they are also known to suffer widely from fraud, such as counterfeit or adulterated electronics and medicines [1]. One key technique to mitigate fraud is *inspection* [2], which is found in a variety of roles through the supply chain, such as quality verification and auditing. Inspection is resource-constrained due to large inspection surfaces of geo-distributed supply chains, high product volumes, and increasingly complex cyber-physical environments. However, cyber-physical supply chains present new opportunities for less resource-constrained digital inspection. Yet, novel characteristics related to Industry 4.0 (I4.0), such as decentralized and autonomous control, increase the complexity of such inspection [3], [4].

The structural properties of supply chains have previously been theoretically studied for influencing fraud [5]. For example, in [6], globalized pharmaceutical supply chains have such structural properties which drive the production of counterfeit and adulterated medicines. The mix of regulations and cultures causes conflicting quality standards. Yet quality inspection is made cost-ineffective due to these properties. Deciding where and when to allocate finite inspection resources is known as the *inspection resource allocation problem* [2], [7]. This is further complicated by emerging cyber-physical characteristics, such as cross-organizational system integration and dynamic autonomous decision-making at both the factory and supply chain level. Throughout this article, we consider the example of a globalized pharmaceutical supply chain where consumers have noted low-quality medicines. The supply chain must be inspected to determine where the product adulteration is occurring.

Topology models are one way to represent the high complexity of cyber-physical environments. They describe structural relationships of a given space and its components, allowing queries of structural properties, such as containment, connectivity and proximity. A *containment* query would return all assets contained within a certain room, factory or server, while a *proximity* query would return all assets spatially located within a given distance. Previously in manufacturing, topologies have been employed for modeling industrial processes to formally describe a plant structure [8]. They have also been employed for adaptive security and forensics of cyber-physical environments [9], [10], [11], [12].

In this article, we propose the use of a different type of topology: Of supply chains and their constituent smart factories. Employing bigraphs, which have previously been used for smart-building topology modeling [12], we generate graph structures of supply chains. One directed acylic graph describes the ordering of the assets and supply chain processes and another tree graph denotes the asset containment (e.g., a machine within a factory). These two graphs combine together to form the

Manuscript received 17 February 2022; revised 6 June 2022 and 29 July 2022; accepted 19 August 2022. Date of publication 9 September 2022; date of current version 22 March 2023. This work was supported in part by Science Foundation Ireland under Grant 16/RC/3918, Grant 13/RC/2094_P2, and Grant 16/SP/3804, and in part by EPSRC under Grant EP/R013144/1 and Grant EP/S036091/1. Paper no. TII-22-0742. *(Corresponding author: Thomas Welsh.)*

Thomas Welsh, Faeq Alrimawi, Ali Farahani, and Diane Hassett are with the Department of Computer Science and Information Systems, University of Limerick, V94 T9PX Limerick, Ireland (e-mail: thomas.welsh@ul.ie; faeq.alrimawi@ul.ie; ali.farahani@ul.ie; diane.hassett@ul.ie).

Andrea Zisman is with the School of Computing and Communications, The Open University, Milton Keynes MK7 6AA, U.K. (e-mail: andrea.zisman@open.ac.uk).

Bashar Nuseibeh is with the Department of Computer Science and Information Systems, University of Limerick, V94 T9PX Limerick, Ireland, and also with the School of Computing and Communications, The Open University, Milton Keynes MK7 6AA, U.K. (e-mail: bashar.nuseibeh@ul.ie).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TII.2022.3205369.

Digital Object Identifier 10.1109/TII.2022.3205369

topology which permits the aforementioned structural queries. We use them as a key component of *adaptive inspection*—previously proposed to optimize inspection of dynamic manufacturing environments to mitigate supply chain fraud [6]. Adaptation is needed to accommodate environment and contextual changes. However, approaches for inspection planning in I4.0 are generally absent from the literature [13]. While traditional techniques are mostly statistical [7], and fail to consider the cyber-physical interplay, modeling these relationships is essential due to their inter-dependency, and for which topologies are well suited [11]. The impact of topology models upon inspection resource allocation in dynamic supply chains and factories has not previously been studied. This work is therefore directed by the following research questions.

**RQ1:** What is the computational performance impact upon adaptive inspection of supply chains when informed by topology models?

**RQ2:** What features of topologies are most suited to informing specific supply chain inspection cases?

In this article, we answer these questions empirically. We employ graph-theoretic discrete simulations of supply chains to measure the time taken for adaptive inspection to discover a malicious process, when informed by varying structural queries. We vary supply chain parameters such as asset containment density, dynamism (structural changes), and inspection costs. While earlier work [6] defined adaptive inspection and proposed its use of topology models, the novel contributions of this work are in evidencing their use in modeling the structure of dynamic cyber-physical supply chains, and illustrating the performance improvement within the adaptive inspection framework. Topology models have been used previously for modeling industrial plant processes and for security and forensics of smart buildings. Yet, we have identified a gap in literature in that they have not been investigated for modeling cyber-physical supply chains, particularly as a tool to optimize inspection resource allocation decisions for purposes of mitigating fraud relating to product integrity. This work extends previous work through implementing the defined adaptive inspection and topology modeling technique for supply chains. We developed three graph-theoretic structural queries upon the topology model, illustrating their performance quantitatively for improving the inspection process. This contributes to the literature of software-based fraud mitigation in supply chains. The article shows empirically that through modeling supply chains with topological techniques, structural information can be used to inform inspection techniques in efforts to reduce fraud.

The rest of this article is structured as follows. Section II provides the background in inspection of supply chains and the use of topological models of cyber-physical environments. Section III presents the adaptive inspection framework, Section IV discusses the experimental setup and results. Section V discusses the work and, finally, Section VI concludes this article.

## II. Background and Related Work

### A. I4.0 Supply Chain

A supply chain comprises a network of entities that collaborate to achieve the manufacturing and sale of a product: Mining raw materials and their refinement, manufacturing and integration, and the distribution and sale of the final product to the end consumer. An I4.0 supply chain still seeks to accomplish the same goals as a traditional one. It operates upon data-rich, integrated, autonomous, and decentralized environments built upon the principles of multidimensional integration [14]: Horizontal intercorporation cooperation across departments; vertical within the factory; and end-to-end in the form of product data across the value chain. Cross-corporation boundary data flow will permit agile and dynamic manufacturing, culminating in collaborative manufacturing which quickly responds to changing markets and individual product customization.

The migration from a manufacturing environment with low digital technology penetration to one which is strongly automated is a primary indicator of I4.0 maturity. The highest level of maturity is achieved once this digital penetration is integrated across the entire value chain [14]. Plaga et al. [15] and Brettel et al. [16] model this evolution from the perspective of the IEC 62264 automation pyramid, yet with the inclusion of cross-organization decentralized decision-making. Moreover, *value* drives the dynamic nature of the supply chain as a result of greater horizontal and end-to-end value chain integration [17]. In contrast to the supply chain which involves the physical movement of goods from one point of the chain to another, the value chain is responsible for the creation of value at each step. Therefore, as an addition to the decentralized supply chain model, I4.0 contains decentralized value chains or *value-networks*. Traditional supply chain environments, which were linear processes composed of distinct entities, are now moving to a decentralized, nonlinear process where entities are integrated through digital means. This creates a fundamentally different landscape, requiring new processes for analyzing fraud which considers these value-driven structures.

### B. Fraud in the Supply Chain

Fraud is an activity in which value is transferred from one party to another through deceptive means [18]. The target of this transfer is an *asset* which holds a perceived social or financial value to both parties. Fraud is instigated by one or more collaborating *deceivers* against one or more *victims*, be they natural persons or organizational entities, e.g., companies, organizations, or governments. *Drivers* of fraud are commonly tangible economic reasons such as financial manipulation or to bypass regulation. They may also be intangible due to culture, high complexity, or irrational behavior [19]. Fraud should be considered during supply chain risk management, which allocates hard (physical) or soft (managerial) controls according to the perceived risk [20]. Supply chains are inherently *value-driven*, where the precursors and final products are assets which are all potential targets of fraud. They also contain value in supporting assets, e.g., machinery, vehicles, people, IT, data, geographical space, contractual agreements, social, corporate, and public relationships. Therefore, this *asset-rich* environment creates a *value-rich* attack surface suitable for varying forms of fraudulent deception. In general, fraud in the supply chain could result in integrity violations of any one of these assets [18], [21]. Therefore, fraud is *enabled* in the supply chain where

controls have been inadequately applied or risks not sufficiently considered or prioritized [22]. The assets listed previously can be targets for controls depending upon the type of fraud which needs to be reduced.

Asset provenance is a common control employed both internally and externally to an organization. Tracking and tracking of assets through the chain is seen often in the literature [23], with plentiful technical solutions being developed [24] such as IoT and radio frequency identification (RFID)-based tracking approaches [25]. However, some simple attacks in the physical domain (e.g., mislabelling) [26] can subvert the provenance of these assets while the digital records which represent them are also open to a variety of traditional information security attack vectors, further complicated as the systems are managed independently across geographical and organizational boundaries by different actors. As one mitigation tool, distributed ledgers are seeing considerable interest as they provide a cryptographically assured immutable database of transactions which can be used for monitoring asset provenance through transparency and irrefutability of records [27]. They can be combined with IoT techniques to ensure difficulty in maliciously reducing asset integrity [28]. Unfortunately, distributed ledger-based solutions suffer from a fundamental disconnect between the cyber and physical dimensions. The digital representation and transactions associated with an asset within the ledger are immutable, the physical representation or tag is not protected by them. For example, tags can be swapped between assets. Therefore, even with the introduction of these digital assurance techniques, the physical dimension must still be considered.

Fraud *prevention* is arguably more important than *detection*, particularly for product violations which could have grave public health risks such as food and medicines [29]. Spink et al. [30] argue that fraud should move from risk mitigation to strategic prevention. Despite this, a wide range of detection techniques can be seen in the literature and are necessary as absolute security is generally unachievable and the dynamic nature of modern smart supply chains coupled with the continuous development of digital technologies and, thus evolving attack vectors, requires detection techniques to adapt to new attacks.

Data mining for fraud detection in a variety of contexts has been studied extensively [31], particularly for matters of financial fraud [32]. Semi-supervised machine learning was used to classify transactions as fraudulent or not for smart supply chains [33] and unsupervised machine learning was used to detect anomalous itineraries to predict which shipping containers were likely to be risky enough to inspect [34] while another data-driven approach to detecting smuggling and miscoding in international shipping is used in [35]. Analyzing a firm's financial reporting has been shown to predict fraudulent supply chain practices when considering managerial performance and personal information [36]. Yet, social media is an alternative source of data, such as for comparing information on social media with the traditional supply chain data [37]. The authors illustrate that using the "wisdom of crowds" inherent to social media enables better identification of corporate fraud. Downsides to these approaches focus around the system's ability to adapt to change due to model convergence, the availability of suitable data given cross-organizational systems, and lack of insight into the physical domain as with digital track and tracing techniques. We therefore posit that due to the high level of interplay between the cyber and physical dimensions within modern supply chains, techniques to mitigate fraud should consider both dimensions in tandem. We suggest that cyber-physical inspection techniques are one such vector for achieving this.

### C. Inspection of Supply Chains for Fraud

Quality inspection (QI) is often used to verify product integrity as it moves through the supply chain and can be useful in detecting and preventing instances of fraud. However, *testing policies* (technique selection) are often known to the supplier and attempts to subvert them are common [7]. QI is constrained by physical supply chain characteristics like high volumes of assets, large size, and geographical distribution. Targeted inspection is necessary as total observation is cost inefficient while also eroding privacy and trust. Therefore, given the size and dynamic nature of supply chains, *sampling policies* must selectively choose when and where and how to inspect, a challenge knows as the *inspection resource allocation problem* [2], [7].

The basis of inspection requires a *target*—an asset which is *valued* by the stakeholders whose characteristics are verified against instance-specific requirements, and a corresponding *technique*—which can interface with the target and provide data to validate the requirements, which has an associated cost and accuracy. Inspection also has one or more *constraints*—mostly fixed and variable costs related to the inspection process and its impact upon the nominal supply chain functions. These constraints are complicated in the presence of emerging cyber-physical supply chain characteristics driving assets and their costs to change, necessitating an adaptive response.

Approaches to manage QI resources often have statistical basis [7], as defects in products and the processes that manufacture and inspect them are considered inevitable due to the stochastic properties of the natural world [38]. Inspection can occur considering a probability distribution, where the cost of inspection can be balanced against the probability of a defective product occurring and/or errors in testing methods [39]. Avoiding QI to save costs is known to have a detrimental effect in the longer term [40].

Statistical methods are thwarted by intentional human action, for example, defrauding a testing technique, and in [7], the authors use belief desire intention (BDI) modeling as a decision support system (DSS) to consider when an actor may choose to defraud inspection. They illustrate that DSS can reduce incidences of fraud through learning intentions from QI and instigating contractual changes in response [2]. A similar QI-contract relationship is found in [41]. Although Lin et al. [1] evidence that while inspection policies can influence a decision to commit fraud in cold-vaccine supply chains, sometimes excessive inspection does not. However, in [42], the authors illustrate that QI alone cannot prevent defrauding, as deferring payment or other incentives is necessary. These works illustrate that statistical techniques alone lack insight when considering complex socio-technical environments.

Furthermore, dynamic supply chains require corresponding dynamic inspection regimes, and adaptive inspection [6] was proposed to handle this change. While a data-rich I4.0 environment contains cyber-physical interfaces into all assets to accommodate integration, integrated approaches to inspection planning in I4.0 are generally absent from the literature [13]. However, analysis of the environment and context is key to adaptation, requiring modeling techniques suitable for both the cyber-physical environment and the change required for optimization decisions. We suggest *topology models* as a suitable technique for this purpose.

### D. Topology Models

In this article, topologies model *structural relationships* of an environment between its *components*, including any entities (assets or actors) contextually associated within it, e.g., machines, people, or products within a factory. They permit structural relationships to be queried in a computationally efficient manner to inform various analyses, the formal representation of which are *structural queries*. Such queries will take a component, a desired relationship property, and the topology itself as an input. Executing a query will then output a corresponding number of components according to their relationships.

A variety of structural relationships may be queried depending upon the domain and use-case. For example, *containment* describes where a component is placed within an enclosure or which components are collocated with it. *Proximity* indicates more precise spatial relationships such as a component being in a specific physical distance to another (e.g., an actor in close proximity to a machine). Components may be *connected* or *adjacent* to another, such as a room to a corridor or a PC to a network switch, and components may be *reachable* to one another through a series of connections.

Topology models have previously been proposed to model combined cyber and physical dimensions of buildings for adaptive security, privacy, and forensics [9]. An asset *contained* in one room will affect the security controls employed to assure associated security goals of that asset [11]. They have also been used to support adaptive access control in smart buildings in response to contextual changes [10]. For privacy, they can model the location of an entity and any information exposure from being in proximity to a potential threat. Further success was shown in supporting analysis of complex cyber-physical incidents for forensics [12].

The technique used for modeling topologies will be specific to the analytical requirements of each use-case. For example, traditional implementations include Building Information Management [10], computer-aided design drawings, and manufacturing process diagrams [8], which are used by domain professionals depending on their skills. They have also been heavily represented using various discrete graph techniques [9], [10] due to the availability of structural analysis algorithms and extensibility [6]. Variations such as bigraghs are employed for modeling multidimensional cyber-physical environments and their extension: Bigraphical reactive systems [11], [12] are used as a formalism for change. We note the lack of topology models in the literature for cyber-physical supply chains, particularly those suitable for dynamic environments and highlight the need to evaluate their future use.

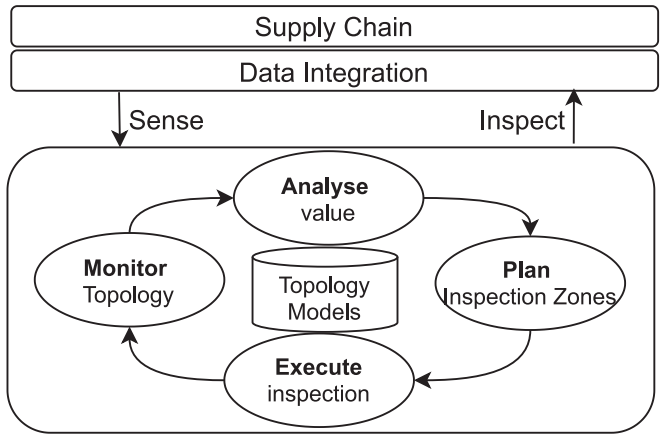


Fig. 1. Topology-aware adaptive inspection of supply chains adapted from [6].

## III. Adaptive Inspection of Supply Chains

Adaptive inspection provides the ability to optimize inspection planning within the presence of contextual and environmental change. However, topology models are needed to support inspection optimization of high-complexity cyber-physical environments. This section now brings these two concepts together. In our previous work [6], we laid out a research agenda for *adaptive inspection of supply chains*. The conceptual architecture (Fig. 1) is based upon the Monitor Analyze Plan Execute-Knowledge (MAPE-K) feedback loop reference model [43], in which the integrated supply chain data is sensed and then inspected according to the decisions of the adaptive inspection loop. In this section, we expand on that model by illustrating how *structural queries* can inform stages of the MAPE loop. Centrality is used during *analysis* to focus inspection on assets with strongly connected properties, containment queries are used to reduce costs during *planning*, and adjacency is used during *execution* to find malicious assets through indirect inspection.

In this article, we work with the following assumptions: Principally that supply chain and manufacturing systems are integrated horizontally and vertically with the factory and end-to-end across the supply chain to support I4.0 processes and thus data is available to build and maintain topology and asset models; that supply chains are a linear process which are typically free of loops so that they can be modeled consistently; that supply chain actors have assigned value to their assets which are accessible to allow comparison; that a pool of inspection techniques exists for each asset type and that the costs of using them are known so that they can be compared and selected; that factories and other premises are not open, they are contained (i.e., have distinct boundaries such as walls or digital compartmentalization) which cause the cost of different inspection techniques to vary in order to cross these boundaries; that similar supply chain processes tend to be grouped together in the same contained environment to

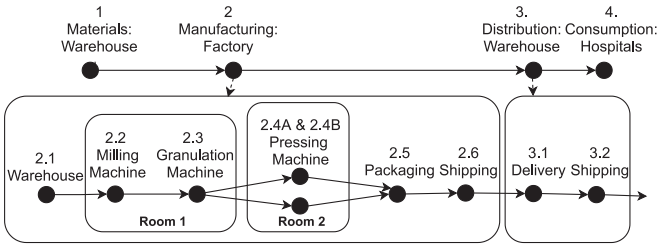


Fig. 2. Example of supply chain topology illustrating containment and connectivity of supply chain processes.

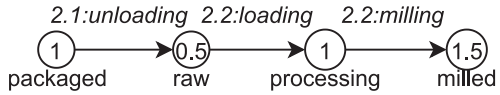


Fig. 3. Example linear transition system asset representation.

allow consistent supply chain model generation; and that supply chain process configuration changes are distinct enough to be observed and verified through software means.

Supply chains describe a typically linear process in which assets move through a sequence of processes causing a corresponding sequence of state changes. We consider them as *spatio-temporal* with the supply chain structure corresponding to the dimension of space and the changing state of the assets across processes representing time. In the running example, a pharmaceutical supply chain must be inspected to determine where the low-quality medicines are being produced. This requires verifying the integrity of manufacturing processes and their outputs. For example, a granulation machine could be inspected to determine if the balance of ingredients was correct, or a packaging machine could be inspected to verify the authenticity of the packaging. This results in a large search space of candidate inspection locations which may be subject to change.

Fig. 2 illustrates a semi-formal and simplified pharmaceutical supply chain topology taken from the example, represented as a directed acyclic graph (DAG) with added containment relationships. The vertices in the DAG correspond to models of cyber-physical processes within the manufacturing process, which receive *assets* as input and will then undergo a state transformation according to one or more states of the cyber-physical process. For example, consider node *2.1 Warehouse* and *2.2 Milling Machine*. The asset in Fig. 3 will change state from *packaged* to *raw* according to the process state *2.1 Unloading* from *2.1 Warehouse* and then the *2.2 Loading* process state from *2.2 Milling Machine* will cause the asset to transition to state *processing*.

The variables used in the following definition are listed in Table I. Formally, we define the supply chain topology as a tuple $SC = (P, E, H, \mu, F, \delta, K, \kappa)$, where $P$ is a set of combined cyber-physical supply chain processes (e.g., $\{MillingMachine, ShippingDepot\}$); $E$ is a list of ordered relations between the processes $E \subseteq P \times P$; $H$ is a set of process attributes; $\mu$ is a mapping between processes and their attributes $\mu : P \rightarrow H$; $F$ is a set of unique flags used to maintain

TABLE I
VARIABLES USED FOR THE MODEL DEFINITION

| Var. | Description |
|---|---|
| $SC$ | Supply chain topology as a tuple $(P, E, H, \mu, F, \delta, K, \kappa)$ |
| $P$ | A set of cyber-physical supply chain processes $p \in P$ |
| $E$ | A list of ordered process relations $E \subseteq P \times P$ |
| $H$ | A set of process attributes |
| $\mu$ | A mapping of process attributes $\mu : P \rightarrow H$ |
| $F$ | A set of unique flags of inspection outcomes |
| $\delta$ | A mapping between processes and flags $\delta : P \rightarrow F$ |
| $K$ | A set of container environments $k \in K$ |
| $\kappa$ | A mapping between containers and process $p$ |
| $AST$ | An asset model as a tuple $(A, V, \sigma, C, \tau)$ |
| $A$ | A set of asset states |
| $V$ | A set of value changes $V \subset \{R+\}$ |
| $\sigma$ | A mapping of asset state value changes $\sigma : A \rightarrow V$ |
| $C$ | A set of cyber-physical supply chain process states |
| $\tau$ | A mapping of supply chain process states $\tau \subseteq A \times C \times A$ |
| $VM$ | Value model as a tuple $(X, \lambda)$ |
| $X$ | A set of centrality values |
| $\lambda$ | A mapping of asset state values $\lambda : X \times A$ |
| $C_a$ | Asset inspection costs |
| $Z$ | Maximum inspection cost |
| $E_p^-$ | Adjacent-in processes for any process |
| $E_p^+$ | Adjacent-out processes for any process |

a track of inspection outcomes (e.g., a negative, neutral, or positive inspection $\{-1, 0, 1\}$); and $\delta : P \rightarrow F$ is a mapping between processes and flags. $h \in H$ is used to identify and verify the processes (such as a unique hash of the process function), to ensure the relations $\delta$ are current. $p \in P$ may be another topology representing its subprocesses or $\emptyset$ depending on the level of detail required to the model. This permits multilevel topologies. Finally, $K$ is a set of container environments (e.g., $\{MachineRoom1, MachineRoom2\}$), and $\kappa$ is a mapping between containers $k \in K$ and processes $p$. From this model, two data structures analogous to bigraphs can be drawn where the ordered relations $E$ are the connectivity graph and $\kappa$ the location graph.

As an asset moves through the supply chain processes, it will undergo a sequence of state changes which may adjust its value, although the particular order and number of states are unknown due to the dynamic nature of the environment. Formally, a tuple $AST = (A, V, \sigma, C, \tau)$, where $A$ is a set of asset states, $V$ is a set of value changes such that $V \subset \mathbb{R}+$, $\sigma$ is a mapping between asset states and changes in value $\sigma : A \rightarrow V$, $C$ is a set of cyber-physical supply chain process states, e.g., $\{unloading, loading, milling\}$, and $\tau$ is a mapping in which the supply chain process transmutes an asset from one state to another $\tau \subseteq A \times C \times A$.

### A. Monitoring Supply Chain Topology Models

In this stage, the generated topology models are monitored through comparing the current model to the previous one to identify changes. At each iteration, a cryptographically secure hash is made of the topology attributes and stored within a database. This hash is then compared at the next iteration whereby a clash will indicate changes have occurred. This could include the structure of the supply chain changing due to factories reconfiguring, new suppliers entering the chain, transport

routes closing, smart products changes due to customer needs, or machine configuration changes as a result of updates or new products (e.g., medicines) having different ingredients. In the topology given previously, the unique attributes $H$ are used for this comparison.

### B. Topology Value Analysis

Supply chain topology models provide the *inspection surface*, a set of spatio-temporal coordinates suitable for inspection. An analysis of this inspection surface determines the *value* for inspection, which can be later balanced against the *cost*. Value analysis must be computable at varying scale to permit timely operation while considering the environment and its context. Value analysis may take many forms. In this instance, we first analyze the environment's structure empirically through measuring each node's degree centrality, which indicates the importance of a node according to its connections [44]. This is useful as a higher number of connections correlate to a higher level of observability of the network through observing the input and output of different processes. The resulting values of each node can then be compared quantitatively according to the degree centrality. For example, in the running example, *2.5 Packaging* would have a higher value than *2.1 Warehouse* as it provides indirect inspection of three processes over one process. This approach is less intrusive than inspecting directly since it reduces the disruption and cost, and increases the value of the inspection. We consider centrality as a structural query as it is a form of adjacency analysis. Our approach selects all processes within the topology that are suitably flagged according to previous inspections. $Q = \{p : p \in P \wedge \delta(p) > 0\}$, where $\delta(p)$ is given in (1).

$$\delta(p) = \begin{cases} -1 & \text{if p inspection was negative} \\ 0 & \text{if p inspection was positive} \\ 1 & \text{if p has not been inspected} \\ 2 & \text{if p should be prioritized for inspection} \end{cases} . \quad (1)$$

Following the selection process, centrality is calculated in the normal way for each $q \in Q$, $C_D(q)$. The value model is a tuple $VM = (X, \lambda)$, where $X$ is a set of centrality values multiplied with reachability and corresponding contextual value $x \in X = q \cdot r \cdot v$ with $v \in V$ as previously defined contextual value in the asset model and $r = R(p)$. Finally, $\lambda : X \times A$ maps the asset state to its combined value. The contextual value acts as a multiplier, whose sensitivity will be adjusted according to the requirements. A product with high financial value would be reflected in the context and, thus, scale the value accordingly. Whether the optimization would seek a high or low value is scenario-dependent. Cases of theft could consider high value and low adulteration.

### C. Inspection Zones Planning

Once the value analysis has been computed, inspection can be planned by defining inspection zones [6] around one or more assets according to available inspection resources. Inspection zone planning (IZP) involves selecting a subgraph of the topology according to the *value* of inspection against the cost. IZP is a combinatorial optimization problem and, therefore, a variety of search-based solutions may be applicable. IZP could be considered as an instance of the knapsack problem [45], for maximizing the value of inspection associated with the asset state's value. This is similar to the value model ($VM$) within constraints of inspection cost, which correlates to the knapsacks total weight constraint. Consider the asset states $a \in A$, values $x_i$ with costs $c_i$, and maximum inspection cost $Z$. Equations (2) and (3) describe the IZP.

$$\text{Max} \sum_{i=1}^{|A|} x_i a_i \quad (2)$$

$$\text{Subject to} \sum_{i=1}^{|A|} c_i a_i \leq Z \text{ and } a \in \{0, 1\}. \quad (3)$$

In order to find a solution to the IZP, the cost of inspecting each location and the maximum cost allowed must be calculated from costs directly associated with the inspection process and contextually associated with the environment. The complexity and scale of these costs are out of the scope of this article. $C_a$, the cost of inspecting asset state $a$, is simply the sum of the elements of all direct and contextual costs. Costs may be adjusted according to asset's containment. For example, by reducing the cost of all collocated components being inspected by a certain factor. If containment has an effect, we determine containment relationships, using the definition of the topology model. Let $Con(p)$ be the set of cocontained for any process $p \in P$, e.g., $Con(2.4\,APressing) = \{2.4\,BPressing\}$.

### D. Executing Inspection

Inspection can be executed based on the defined processes and assets, e.g., verifying the pharmaceutical ingredient ratios match the configuration of *2.3 Granulation Machine* or the firmware version of a machine. The result of the execution will inform the next iteration of the MAPE-K loop. It can exclude places previously inspected and flag processes adjacent to those which are malicious to ensure completeness. The inspection function $\iota(a)$ returns the result of the integrity evaluation of a process and asset (positive or negative), which is added to the topology model.

Execution is a candidate for optimization according to adjacency, within a linear process such as the factory or supply chain. If an inspection notices that the input to the process had not met its quality requirements, it provides indirect inspection, allowing the suspicious adjacent process to be flagged for prioritized inspection. Adjacent processes are those with both edges in and out to the process $out(p)$ nodes of a $p$, where $E_p^-$ formally defines the subset of ordered relations in $E$ of the form $\{p,' p\}$ and $E_p^+$ the subset of ordered relations in $E$ of the form $\{p, p'\}$. For example, $E_{2.4A}^- = \{2.4\,AGranulation\}$ and $E_{2.4}^+ A = \{2.5Packaging\}$.

## IV. Methodology, Results, and Analysis

To address the research questions posed in the introduction and illustrate the efficacy of topology-aware adaptive inspection,

TABLE II
SIMULATION VARIABLES

| Variable | Description |
|---|---|
| $\|P\|$ | Quantity of supply chain processes set to 100, noting that size is related to linear performance increases |
| Dynamism | The rate of change of processes, e.g., 0.25 would denote 25% of randomly selected nodes would change configuration at each iteration |
| Cost model | The numerical base cost to inspect a supply chain process. The maximum cost per iteration is 1; e.g., 0.5 would allow 2 inspections each iteration |
| Adjacency | Enables adjacency analysis in edges $E_p^-$ to provide indirect inspection during the execution stage |
| Containment | Reduces the inspection cost of collocated supply chain processes by this factor; e.g., Cont10% reduces the cost of all to 10% of the base cost during the plan stage |
| Centrality | Structural value analysis enabled or disabled |
| Timeout | Set to $10 \cdot \|P\|$. While high, it includes outlier test cases resulting from dynamism |
| Result | Quantity of iterations till malicious node discovery |

we present an evaluation of simulated supply chain topologies. Discrete simulations were chosen for their applicability to graph techniques as this work studies the structure of the supply chain as opposed to factors related to the flow of products more relevant to continuous simulations.

In the following subsections, we first present our methodology, and then the results and analysis of tests upon static supply chains which provide a performance baseline and comparison for the results and analysis of dynamic supply chains tests, used to evaluate the structural techniques.

## A. Methodology

The methodology is described below and Table II gives the variables used for this purpose. The dataset employed is a result of generated supply chains. The simulated approach is as follows. First, a graph of size $P$ representing the supply chain connectivity is generated. In this instance, the graph is a growing random network [46] due to its linear direction and structure being similar to the properties of a typical supply chain. Next, the containment mappings ($\kappa$) are generated: $P \cdot 0.3 \cdot jr$ (where, $j$ is jitter) factory containers $k$ are added, e.g., $\{(SC, factory1), (SC, factory2)\ldots\}$, then $2j$ containers (e.g., rooms or servers) are added to each factory, e.g., $\{(factory1, room1)\ldots\}$. Then, $\forall p \in P$ generates a random variable $r$, and $\{k \in K | \text{where k is a room}\}$ $o(r)$ determines where $p$ is contained. This was chosen to distribute processes unevenly with a bias toward grouping assets together, similarly to a production line.

$$o(r) = \begin{cases} \text{Next container} & r \leq 0.3 \\ \text{Next factory, 1st container} & 0.3 \leq r \leq 0.5 \\ \text{Current container} & r \geq 0.5 \end{cases}.$$

One process is then marked as malicious according to a random distribution. The goal of the adaptive process is to reduce search times for the malicious process under varying level of dynamism. Next, all permutations of the variables in Table II are executed 100 times with the number of iterations taken to discover the malicious processes stored as a result. Multiple values for iterations between 10 and 1000 were first tested and

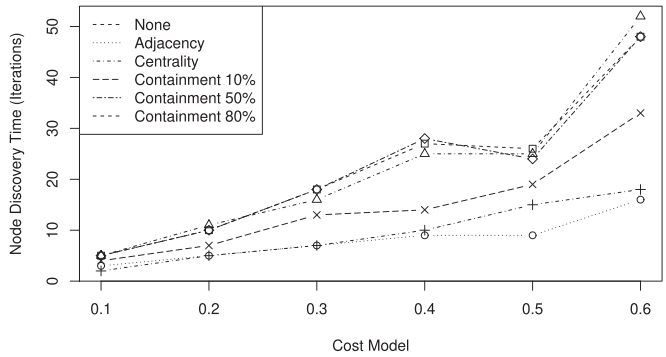


Fig. 4. Illustration of the mean malicious node discovery time for all structural queries and cost models 0.1–0.6 for static supply chains.

100 was chosen for a consistent convergence of the results across different test types. However, for different supply chain sizes and characteristics (e.g., asset containment density), this value may need to be tuned. The results below are summarized through the mean of all tests. Due to the high timeout, the mean shows trends for comparison of different parameters as opposed to exact performance. This was chosen due to the strong pseudorandom elements of the simulation causing outliers. These tests then permit an analysis of the different employed structural queries to determine and optimize the adaptive inspection process.

## B. Static Supply Chain Results and Analysis

We first analyze the performance of adaptive inspection upon static networks, where the dynamism $= 0$, to provide a base line for later comparison and also to validate the simulation.

*Structural Query Performance* is shown in Fig. 4 illustrating the static supply chain tests for cost models 0.1–0.6, where all distinct structural queries show a positive incline. When the cost increases, fewer assets can be inspected per iteration. Cost models 0.4 and 0.5 have similar results as the base inspection allows for two asset inspections per iteration. The exception is for containment queries as these reduce the base cost. As the cost models increase, the performance tends to decrease by ten iterations. However, for cost model 0.6, the decrease in the performance doubles to ~20 iterations due to only allowing one asset per iteration by default, confirmed with the performance improvement Cont10% and Cont50%. The figure also illustrates the performance difference between structural queries, with adjacency being the best to perform. Cost models 0.1 and 0.5 are next with another intuitive increase in performance as the cost reduction diminishes. While centrality is performed poorly with similar results, or worse to the baseline, when containment cost reduction reaches 0.8, it provides no improvement.

*Combined Structural Query Performance* is illustrated in Fig. 5. We select just one containment model 0.1, as the decrease in performance between higher values is given. We apply these techniques to the poorest performing cost models, 0.5 and 0.6, with a random baseline for comparison, as the poorest performing are those best targeted for improvement. Comparing combinations shows clear performance improvement. Centrality, which had minimal effect alone, improves performance when

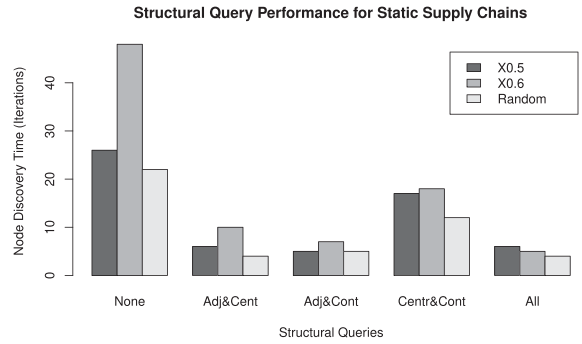


Fig. 5. Illustration of the mean malicious node discovery time for combinations of structural queries for static supply chains.

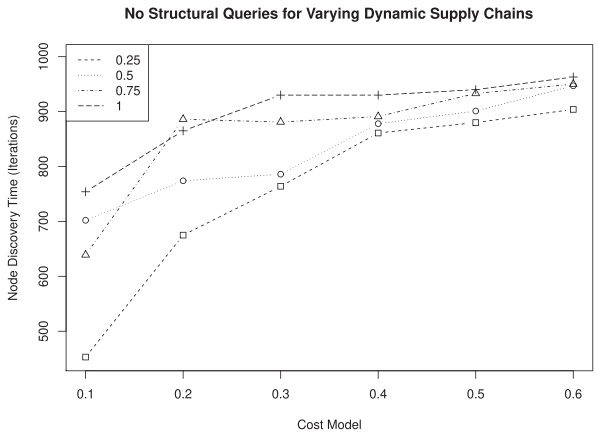


Fig. 6. Illustration of the effect of different cost models upon varying levels of supply chain dynamism for no structural queries.

combined with adjacency, but not with containment. Likewise, adjacency and containment combined are better performing than individually. Interestingly, all combined are better performing for the higher cost model of 0.6 but similar in performance to adjacency and centrality for the lower and random cost models. Due to increased complexity, this illustrates the use of selecting specific combinations of techniques according to the underlying supply chain parameters. Overall, these results show variety in combining structural queries for performance increase.

### C. Dynamic Supply Chain Results and Analysis

Next we present the results of tests with varying levels of dynamism in the supply chain to see the effect of the structural queries on the time to find a malicious node.

*Supply Chain Dynamism and the Cost Model Relationship* are shown in Fig. 6, illustrating the effect of dynamism on the mean search time with no enabled structural queries. It shows the negative effect where search times have moved from ∼50 iterations previously to ∼900 to ∼1000 for the highest cost model. The best performing (0.1 at 25% dynamism) is still an increase from ∼40 to ∼400 iterations.

*Structural Query Performance* for a selected case is shown in Fig. 7. We select the test cases with 25% dynamism (i.e., 25% of the nodes within the supply chain will change at each iteration, where changes refer to configuration of the supply chain asset such as machine software or its structure in the

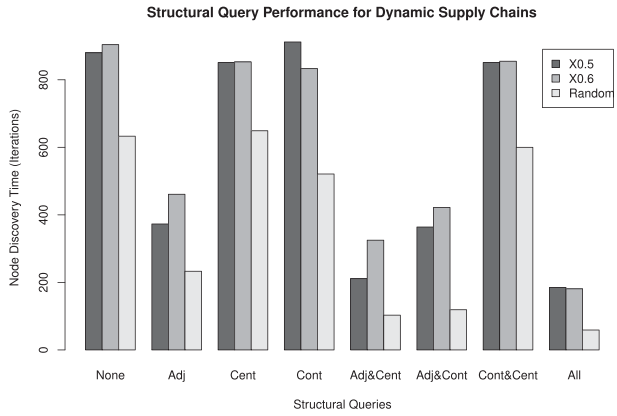


Fig. 7. Illustration of the mean malicious node search time for combined structural queries where supply chain dynamism is 25%.

chain). Higher levels of change within factories seem unrealistic and the linear reduction in performance when increasing the dynamism rate is implied. As done previously, we also select the higher cost models as these are a priority target for optimization. As with the static tests, these results illustrate clear increases in performance when using and combining different queries. Adjacency is again the strongest performing distinct query, reducing to 50% iterations of no queries. When combined with centrality, performance is improved further by ∼23% and 35% for cost models 0.5 and 0.6, respectively. Centrality highlights the areas of the structure with influential adjacency links so the success of this combination is dynamic, but not static supply chains. In contrast to the static tests, adjacency combined with containment is not as successful, and containment combined with centrality is close to no structural queries, while all queries combined is still the best performing. As it completes in ∼20% of the number of iterations it took to complete none (an 80% increase in performance) in both cost models 0.5 and 0.6. For the random cost model, this is ∼10% of the time of none (a 90% increase in performance). These results illustrate that the supply chain parameters such as the dynamism and cost model can inform the set of structural queries chosen.

## V. Discussion

Considering RQ1, the results in Section IV illustrate that all structural queries could reduce the node discovery time by up to 90% for the random test case, and to 80% for the other cases, and thus the use of topology models reduces the computational impact of adaptive inspection. Reflecting on RQ2, centrality, for example, was more effective when combined with adjacency than alone for dynamic networks, and had little effect for static networks. Containment reduction analysis has an effect relative to its context-dependent cost reduction, although low reductions of less than 50% provided little benefit.

### A. Threats to Validity

Internal threats to validity are caused by the pseudo-random number generation of the costs, malicious node selection, and graphs can all produce outliers. This can cause wide variation which can bias the node discovery time. An example can be

seen in Fig. 5 cost models 0.1 and 0.2 for dynamism 0.25, where outliers skewed the trend. We mitigated this by generating 100 cases per test case and examining the mean, yet noting that these results illustrate trends over exact values.

External threats to validity which reduce generalizability of our example are noted in this work due to the simulated nature. Generated supply chain models are suited for comparison of structural query effects on inspection, yet are not verified as representing real-world supply chains. Therefore, in future work, we aim to use plant-scale digital twin software to generate more realistic topology models. Another external threat is the selection of the malicious nodes. We previously highlighted [6] that perceived *value* of assets plays a key role in motivating and thus analyzing inspection for fraud. Value models are specific to the stakeholders and are out of the scope of this article. The supply chain process to be marked was chosen using a pseudo-random distribution and, therefore, the human-motivation for fraud was none. In future work, the impact of more precise value analysis will likely adjust the impact of the evaluated structural queries. Further external threats relate to the computational complexity of the structural queries and the discrete nature of the simulation, which does not consider time and resource constraints for processing. In terms of the structural queries, degree centrality typically has a time complexity of $O(|P|)$. Executing the degree centrality algorithm using the experimental setup as in Section IV-A where $P = 100$, 1000, 10 000, and 100 000 takes 0.000053, 0.00029, 0.0037, and 0.033 s, respectively. This result is approximately in line with the linear performance increase expected given shared CPU usage. The implementation using NetworkX is not optimized. Therefore, as a conservative estimate, scaling to large real-world supply chains up to 10 000 000 processes would still be calculated in less than 10 s. The most significant computationally intensive task of the adaptive inspection approach is the inspection zones planning. It is an instance of the Knapsack problem, with the decision portion known to be NP-complete. Yet, the optimization solution varies according to the algorithm chosen. The greedy example with sort chosen in this instance is known to be $O(P \times logP)$. Adjacency is less related to the structure, instead assuming a technique at the process to verify the asset specification. This will be use-case-specific and may not always be available, although smart-product-based digital twins show promise. Containment analysis is also use-case-specific, with cost and asset density parameters affecting performance. The complexity would be $O(|Con(p)|)$ for all asset states $a$ chosen according to (2) during IZP. In our dynamic supply chain simulation, we assume static and high cost reduction of 90%. Such scenarios might apply if an agent (e.g., a drone) traveling to an environment, or developing a particular software analysis technique, would reduce the inspection cost. However, in real-world usage, these cost reductions may be dynamic, complex, or even small, where $\leq 50\%$ was shown to have little effect in Fig. 3.

In a real-world supply chain, considering the complexity impact upon the processing time of the approach is crucial due to the dynamic supply chain changes. At every time step in our discrete simulation, the graph, centrality, containment, and greedy algorithm are all computed. Within the real world, this timestep interval would need to be large enough such that it accommodates all of the previously discussed in addition to the real-world events such as time for the practical inspection, the assets to reconfigure, the relevant data to be collected, checked for privacy and accuracy, and transferred to the relevant inspection agent for processing. An attacker's awareness of this periodicity may allow them to subvert the adaptive inspection process if their attack considers the time between changes. Therefore, it is crucial that this process time is reduced and contextually suitable. Given the dynamic and nondeterministic nature of the world, this suggests another consideration for the adaptive feedback loop of constraints set by activities in continuous time and their impact on desired goals (e.g., security) of the system.

### B. Implications for Practice

Considering the results in Section IV and in light of the threats to validity discussed in the previous section, we now discuss the implications of these findings for practice.

*1) Inspection Optimization With Structural Information:* Inspection planning is constrained by cost. Therefore, the most critical finding of our results is that inspection resource allocation decisions can be informed, and thus optimized, through integrating information about the structure of the plant and other premises, and the high-level supply chain itself. This is particularly relevant when production processes are subject to autonomous change, as autonomous inspection planning is necessary to prevent delays caused by human operators' inputs.

*2) Practical Computation and Integration:* The analysis in Section V-A highlighted that the availability and maturity of graph algorithms ensures that the computational requirements for realistic size supply chains are feasible for trivial hardware environments. This ensures that, assuming relevant information about the factory or supply chain layout is available, topology-aware inspection planning techniques can be integrated into established work flows using established hardware. This will also enable wider availability of the techniques due to a low-cost barrier for usage.

*3) Cyber-Physical Inspection Techniques:* An assumption of this work is that suitable inspection techniques exist for the assets, have a cost, and are usable. Although this work is not concerned with these type of techniques, our findings illustrating cost reduction highlight that inspection costs can be reduced considerably if techniques can be reused across dissimilar assets. This indicates that developing techniques which focus on the cyber-physical interfaces, as assumed in this work, can provide more opportunities for reuse and thus dramatic cost reduction.

### C. Implications for Design

The value of this research extends beyond employing topology models for optimizing adaptive inspection. It shows the potential of topology models as a basis for informing design decisions for dynamic restructuring of cyber-physical manufacturing environments.

*1) Smart Factories:* Smart factories can adjust their structure to optimize inspection not only for quality, but for human-motivated activities such as fraudulently adulterated products. Restructuring according to the value of co-located assets within a container can optimize inspection costs, reduce computational

analysis through graph segmentation, or enhance or reduce adjacent processes for both security and privacy. This approach would employ an adaptive loop which will affect the structure of the supply chain instead of inspecting it.

*2) Smart Products:* More relevant to the adaptive inspection approach is the ability for emerging smart products to select their production route according to different criteria such as cost, quality, trust, configuration changes, and emerging events occurring such as natural or security related. An asset might analyze the topology and select processes with higher colocated assets, or a greater number of adjacent processes to increase the chance of observation and, thus, reduce incidences of fraud. An adaptive software loop would sense the supply chain, but act only internally through affecting the design of the route taken by the smart asset.

*3) Privacy and Inspection Zones:* A priority goal for adaptive inspection is to maintain privacy through targeted observation over blanket surveillance to reduce adversarial relationships between stakeholders. However, adaptive inspection would violate this principle if the supply chain processes were inspected in a way which was deemed excessively intrusive to stakeholders. Therefore, a suitable preinspection check of the supply chain structure might identify inspection timeout values to avoid this violation, which should consider the values of the individual and collective stakeholders.

## VI. Conclusion

In this article, we illustrated that structural information from graph-theoretic topology models of supply chains and factories can be used to optimize inspection in efforts to mitigate fraud. This technique can effectively reduce search times for malicious processes by up to 90% when combining multiple structural queries. In addition, we illustrated that change in the structure of dynamic smart factories and supply chains can be managed using adaptive-software techniques. However, structural query performance relating to the cost and type of inspection techniques are context-specific and therefore the cost and availability of these techniques must be considered before use. For example, cost reductions related to asset containment of $< 50\%$ make minimal impact and therefore can be excluded. Furthermore, adjacency queries are dependent upon an indirect inspection technique being available.

We suggest that due to the discrete nature of our simulation, it is necessary to evaluate these structural queries and the practical implications of different inspection techniques under continuous time. Therefore, in future work, we aim to integrate these techniques into digital twin factory simulation software in order to accommodate and evaluate their practicality in continuous time.

## References


[1] Q. Lin, Q. Zhao, and B. Lev, "Cold chain transportation decision in the vaccine supply chain," *Eur. J. Oper. Res.*, vol. 283, no. 1, pp. 182–195, 2020.

[2] J. Yan, X. Li, Y. Shi, S. Sun, and H. Wang, "The effect of intention analysis-based fraud detection systems in repeated supply chain quality inspection: A context of learning and contract," *Inf. Manage.*, vol. 57, no. 3, 2020, Art. no. 103177.

[3] E. Hofmann and M. Rüsch, "Industry 4.0 and the current status as well as future prospects on logistics," *Comput. Ind.*, vol. 89, pp. 23–34, Aug. 2017.

[4] M. Hermann, T. Pentek, and B. Otto, "Design principles for Industrie 4.0 scenarios," in *Proc. 49th Hawaii Int. Conf. Syst. Sci.*, 2016, pp. 3928–3937.

[5] S. DuHadway, C. Mena, and L. M. Ellram, "Let the buyer beware: How network structure can enable (and prevent) supply chain fraud," *Int. J. Operations Prod. Manage.*, vol. 42, pp. 125–150, 2021.

[6] R. Edacted, "Towards adaptive inspection for fraud in i4.0 supply chains," in *Proc. 26th IEEE Int. Conf. Emerg. Technol. Factory Automat.*, 2021, pp. 1–8.

[7] J. Yan, X. Li, S. X. Sun, Y. Shi, and H. Wang, "A BDI modeling approach for decision support in supply chain quality inspection," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 3, pp. 884–898, Mar. 2020.

[8] H. Koziolek, J. Rückert, and A. Berlet, "Industrial plant topology models to facilitate automation engineering," in *Proc. Int. Conf. Syst. Model. Manage.*, 2020, pp. 91–108.

[9] L. Pasquale, C. Ghezzi, C. Menghi, C. Tsigkanos, and B. Nuseibeh, "Topology aware adaptive security," in *Proc. 9th Int. Symp. Softw. Eng. Adaptive Self-Manag. Syst.*, 2014, pp. 43–48.

[10] L. Pasquale et al., "Topology-aware access control of smart spaces," *Computer*, vol. 50, pp. 54–63, 2017.

[11] C. Tsigkanos, L. Pasquale, C. Ghezzi, and B. Nuseibeh, "On the interplay between cyber and physical spaces for adaptive security," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 3, pp. 466–480, May/Jun. 2018.

[12] F. Alrimawi, L. Pasquale, and B. Nuseibeh, "On the automated management of security incidents in smart spaces," *IEEE Access*, vol. 7, pp. 111513–111527, 2019.

[13] M. Rezaei-Malek, M. Mohammadi, J.-Y. Dantan, A. Siadat, and R. Tavakkoli-Moghaddam, "A review on optimisation of part quality inspection planning in a multi-stage manufacturing system," *Int. J. Prod. Res.*, vol. 57, no. 15/16, pp. 4880–4897, 2019.

[14] E. Gökalp, U. Şener, and P. E. Eren, "Development of an assessment model for industry 4.0: Industry 4.0-mm," in *Proc. Int. Conf. Softw. Process Improvement Capability Determination*, 2017, pp. 128–142.

[15] S. Plaga, N. Wiedermann, S. D. Anton, S. Tatschner, H. Schotten, and T. Newe, "Securing future decentralised industrial IoT infrastructures: Challenges and free open source solutions," *Future Gener. Comput. Syst.*, vol. 93, pp. 596–608, 2019.

[16] M. Brettel, N. Friederichsen, M. Keller, and M. Rosenberg, "How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective," *Int. J. Inf. Commun. Eng.*, vol. 8, no. 1, pp. 37–44, 2014.

[17] D. Buhr, *Social Innovation Policy for Industry 4.0*, Friedrich-Ebert-Stiftung, Division Social Econ. Policies Berlin, 2015.

[18] L. Manning, "Food fraud: Policy and food chain," *Curr. Opin. Food Sci.*, vol. 10, pp. 16–21, 2016.

[19] U. Arnold, J. Neubauer, and T. Schoenherr, "Explicating factors for companies' inclination towards corruption in operations and supply chain management: An exploratory study in Germany," *Int. J. Prod. Econ.*, vol. 138, no. 1, pp. 136–147, 2012.

[20] S. M. Van Ruth, P. A. Luning, I. C. Silvis, Y. Yang, and W. Huisman, "Differences in fraud vulnerability in various food supply chains and their tiers," *Food Control*, vol. 84, pp. 375–381, 2018.

[21] G. Van Drunen, M. O'connell, M. F. Hansen, S. Tavares, and K. S. Waldrop, "Supply chain fraud: An holistic approach to prevention, detection and response," 2017. [Online]. Available: https://assets.kpmg/content/dam/kpmg/be/pdf/Markets/supply-chain-fraud.pdf

[22] E. Mu and J. Carroll, "Development of a fraud risk decision model for prioritizing fraud risk cases in manufacturing firms," *Int. J. Prod. Econ.*, vol. 173, pp. 30–42, 2016.

[23] F. Dabbene, P. Gay, and C. Tortia, "Traceability issues in food supply chain management: A review," *Biosyst. Eng.*, vol. 120, pp. 65–80, 2014.

[24] L. Li, "Technology designed to combat fakes in the global supply chain," *Bus. Horiz.*, vol. 56, no. 2, pp. 167–177, 2013.

[25] S. Choi, B. Yang, H. Cheung, and Y. Yang, "RFID tag data processing in manufacturing for track-and-trace anti-counterfeiting," *Comput. Ind.*, vol. 68, pp. 148–161, 2015.

[26] H. R. Shehata, D. Bourque, D. Steinke, S. Chen, and R. Hanner, "Survey of mislabelling across finfish supply chain reveals mislabelling both outside and within Canada," *Food Res. Int.*, vol. 121, pp. 723–729, 2019.

[27] P. Dutta, T.-M. Choi, S. Somani, and R. Butala, "Blockchain technology in supply chain operations: Applications, challenges and research opportunities," *Transp. Res. E: Logistics Transp. Rev.*, vol. 142, 2020, Art. no. 102067. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1366554520307183

[28] B. Alangot et al., "Trace and track: Enhanced pharma supply chain infrastructure to prevent fraud," in *Proc. Int. Conf. Ubiquitous Commun. Netw. Comput.*, 2017, pp. 189–195.
[29] R. Davidson et al., "From food defence to food supply chain integrity," *Brit. Food J.*, vol. 119, pp. 52–66, Jan. 2017.
[30] J. Spink, W. Chen, G. Zhang, and C. Speier-Pero, "Introducing the food fraud prevention cycle (FFPC): A dynamic information management and strategic roadmap," *Food Control*, vol. 105, pp. 233–241, 2019.
[31] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *J. Netw. Comput. Appl.*, vol. 68, pp. 90–113, 2016.
[32] J. West and M. Bhattacharya, "Intelligent financial fraud detection: A comprehensive review," *Comput. Secur.*, vol. 57, pp. 47–66, 2016.
[33] F.-V. Constante-Nicolalde, P. Guerra-Terán, and J.-L. Pérez-Medina, "Fraud prediction in smart supply chains using machine learning techniques," in *Proc. Int. Conf. Appl. Technol.*, 2019, pp. 145–159.
[34] E. Camossi, T. Dimitrova, and A. Tsois, "Detecting anomalous maritime container itineraries for anti-fraud and supply chain security," in *Proc. Eur. Intell. Secur. Inform. Conf.*, 2012, pp. 76–83.
[35] R. Triepels, H. Daniels, and A. Feelders, "Data-driven fraud detection in international shipping," *Expert Syst. Appl.*, vol. 99, pp. 193–202, 2018.
[36] G. Hoberg and C. Lewis, "Do fraudulent firms produce abnormal disclosure?," *J. Corporate Finance*, vol. 43, pp. 58–85, 2017.
[37] F. Xiong, L. Chapple, and H. Yin, "The use of social media to detect corporate fraud: A case study approach," *Bus. Horiz.*, vol. 61, no. 4, pp. 623–633, 2018.
[38] D. N. Chorafas, "Fundamentals of statistical quality inspection," in *Quality Control Applications*. Berlin, Germany: Springer, 2013, pp. 235–255.
[39] M. Khan, M. Y. Jaber, and A.-R. Ahmad, "An integrated supply chain model with errors in quality inspection and learning in production," *Omega*, vol. 42, no. 1, pp. 16–24, 2014.
[40] H. Hu, Q. Wu, Z. Zhang, and S. Han, "Effect of the manufacturer quality inspection policy on the supply chain decision-making and profits," *Adv. Prod. Eng. Manage.*, vol. 14, no. 4, pp. 472–482, 2019.
[41] Y. Zhang and Y. Zhao, "Analysis of the third party inspection strategy under asymmetric quality cost information," in *Proc. Int. Conf. Syst. Inform.*, 2012, pp. 1281–1286.
[42] V. Babich and C. S. Tang, "Managing opportunistic supplier product adulteration: Deferred payments, inspection, and combined mechanisms," *Manuf. Service Operations Manage.*, vol. 14, no. 2, pp. 301–314, 2012.
[43] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer*, vol. 36, no. 1, pp. 41–50, 2003.
[44] M. J. Alenazi and J. P. Sterbenz, "Comprehensive comparison and accuracy of graph metrics in predicting network resilience," in *Proc. 11th Int. Conf. Des. Reliable Commun. Netw.*, 2015, pp. 157–164.
[45] H. M. Salkin and C. A. De Kluyver, "The knapsack problem: A survey," *Nav. Res. Logistics Quart.*, vol. 22, no. 1, pp. 127–144, 1975.
[46] P. L. Krapivsky and S. Redner, "Organization of growing random networks," *Phys. Rev. E*, vol. 63, May 2001, Art. no. 066123. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.63.066123

**Thomas Welsh** (Member, IEEE) received the Ph.D. degree in computer science from Staffordshire University, Stoke-on-Trent, U.K., in 2020.

He is currently a Postdoctoral Researcher with CONFIRM—The SFI Research Centre for Smart Manufacturing, Dublin, Ireland and Lero—the SFI Research Centre for Software, Dublin, Ireland. His research interests include analyzing and influencing structural characteristics of complex cyber-physical systems to achieve security and resilience properties.

**Faeq Alrimawi** received the Ph.D. degree in computer science from the University of Limerick, Limerick, Ireland, in 2020.

He is currently a Postdoctoral Researcher with Lero—The SFI Research Centre for Software, Dublin, Ireland. His research interests include software engineering and digital forensics applied to the domains of cyber-physical systems.

**Ali Farahani** received the B.Sc. degree in computer engineering and the M.Sc. and Ph.D. degrees in information technology engineering from Shahid Beheshti University, Tehran, Iran, in 2009, 2011, and 2017, respectively.

He was a Software Engineering and Architectural Researcher with Lero—The Irish Software Research Centre, University of Limerick, in 2019. Since 2022, he has been a Researcher with Lero, University College Dublin, Dublin, Ireland, and a Senior Software Developer with Genesys, Daly, CA, USA. His research interests include autonomic computing, self-adaptation, and self-organization of software systems.

**Diane Hassett** received the M.Eng. degree in chemical engineering from Queen's University, Belfast, Ireland, in 1998, and the M.Sc. degree in design innovation from Maynooth University, Maynooth, Ireland, in 2018. She is currently working toward the Ph.D. degree in computing with Lero, University of Limerick, Limerick, Ireland.

She is currently an Engineering Manager with Aerogen, Galway, Ireland. Her research interests include human values in the design of systems of systems.

**Andrea Zisman** received the B.Sc. and M.Sc. degrees in computer science from Brazil, and the Ph.D. degree in computer science from the Imperial College of Science Technology and Medicine, London, U.K., in 1998.

She is a Professor of computing, with expertise in the areas of software and service engineering, with The Open University, Milton Keynes, U.K. She was Principal and Co-Investigator in various EPSRC, European, and industry-funded research projects, including the multidisciplinary. Her research interests include adaptation of software systems; traceability of software artifacts; secure software engineering; service engineering, reputation and trust of services; cloud computing; and IoT-based systems and its applications to food security.

Prof. Andrea holds a Parnas Fellowship from Irish Software Research Centre (Lero). She is a member of the EPSRC's Information and Communication Technologies (ICT) Strategic Advisory Team (SAT). She has been Program Co-Chair of major conferences in SE (ASE'05, ETAPS'12, ESEC/FSE'17, ICSE'18 NIER, RE'2020, and ICSE'2022 Area Chair). She is Vice-Chair of the IFIP Working Group 2.9, and Associate Editor of the *Science of Computer Programming Journal* and the *Transactions on Service Computing*.

**Bashar Nuseibeh** received the B.Sc. degree in computer systems engineering from the University of Sussex, Brighton, U.K., and the M.Sc. and Ph.D. degrees in software engineering from Imperial College London, London, U.K.

He is currently a Chief Scientist with Lero—The Irish Software Research Centre, a Professor of Software Engineering, University of Limerick, Limerick, Ireland, and a Professor of computing with The Open University, Milton Keynes, U.K. He is also a Principal Investigator with CONFIRM—The Irish Research Centre on Smart Manufacturing, Dublin, Ireland. His research interests include the intersection of requirements engineering, adaptive systems, and security and privacy.

Dr. Nuseibeh was Editor-in-Chief of IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, *Automated Software Engineering Journal*, and *ACM Transactions on Autonomous and Adaptive Systems*. He is currently an Associate Editor of *IEEE Security & Privacy Magazine*. He has received many awards, including a Philip Leverhulme Prize, a Royal Society-Wolfson Merit Award, and an IET Innovation Award for Cyber Security. He is a Fellow of the Institution of Engineering & Technology, the British Computer Society, and the Irish Computer Society, and is a Member of Academia Europaea and the Royal Irish Academy.