An Analytics Toolbox for Cyber-Physical Systems Data Analysis: Requirements and Challenges

M. Zanin*, E. Menasalvas*, †, A. Rodriguez González*, † and P. Smrz‡
* Centro de Tecnología Biomédica, Universidad Politécnica de Madrid, Madrid, Spain
† Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid, Madrid, Spain
‡ Brno University of Technology, Brno, Czech Republic
[massimiliano.zanin, ernestina.menasalvas, alejandro.rog]@upm.es, smrz@fit.vutbr.cz

Abstract - The fast improvement in telecommunication technologies that has characterised the last decade is enabling a revolution centred on Cyber-Physical Systems (CPSs). Elements inside cities, from vehicles to cars, can now be connected and share data, describing both our environment and our behaviours. These data can also be used in an active way, by becoming the tenet of innovative services and products, i.e. of Cyber-Physical Products (CPPs). Still, having data is not tantamount to having knowledge, and an important overlooked topic is how should them be analysed. In this contribution we tackle the issue of the development of an analytics toolbox for processing CPS data. Specifically, we review and quantify the main requirements that should be fulfilled, both functional (e.g. flexibility or dependability) and technical (e.g. scalability, response time, etc.). We further propose an initial set of analysis that should in it be included. We finally review some challenges and open issues, including how security and privacy could be tackled by emerging new technologies.

Keywords – Cyber-Physical Systems; Cyber-Physical Products; data analytics; information security; information privacy

I. INTRODUCTION

The widespread availability of Cyber-Physical Systems (CPSs) [1, 2], combined with the deployment of telecommunication technologies allowing the transmission of large volumes of data with negligible costs, are triggering a revolution in our societies. Specifically, new business models are being developed, in which novel services (also called Cyber-Physical Products, CPPs) are created by leveraging on the data provided by CPSs. As a familiar example, one may think on the use of geolocated information from mobile phones to detect traffic jams, and the resulting service suggesting optimised routes based on real-time data [3]. Note that the innovation behind CPPs resides in the data they use, as they would not be feasible without the collaboration and sharing made possible by many distributed CPSs.

In spite of many advancements and research works, one important aspect has mostly been neglected: data availability does not equate knowledge availability. In other words, having access to data does not directly imply gaining new knowledge, as data have to be processed. If data analytics is in general a mature field, the same may not be true for data analytics in a CPP environment, due to the specific requirements of the latter – e.g. large data volumes, restricted bandwidths, or privacy of the information. In other words, it may be naïve to think that one-size-fits-all, and that existing analytics solutions may seamlessly be reused in this context.

In this work we focus on the data processing requirements and challenges that will soon have to be tackled, if CPPs have to become a reality. We start by defining a prototypical CPP scenario, developed within the European project Cross-CPP [4], involving a set of cars and building sharing data about the environment they perceive (see Section II). We then tackle two fundamental questions. Firstly, what are the requirements associated to a data analytics toolbox for CPP data? We discuss them in Section III, and observe how existing analytics solutions do not by and large comply with them (Section IV). Secondly, what do these requirements imply, in terms of research challenges and future research works (Section V)? Finally, Section VI draws some conclusions.

II. A CPP SCENARIO

The scenario here envisioned stems from the existence of two complementary paths for value generation.

CPSs producing large volumes of data create new value in themselves, for instance by supporting the improvement of existing services. At the same time, such novel data may be the fuel of new business opportunities, not only for the CPS manufacturer, but especially for cross-sectorial industries and applications. There are therefore two perspectives that have to be taken into account:

- The Data Providers, who manufacture or own the CPSs. These should receive an incentive to share their data, i.e. they should receive a benefit beyond what already guaranteed by the in-house consumption of the data. In addition, they are the party most concerned about privacy and security, as CPS data can
be used to infer activity patterns of the owners and business strategies of the manufacturers.

- The Data Customers, that is, the providers of the new services constructed around the CPS data. They benefit from an easy and homogeneous access to the data, i.e. not depending on multiple protocols and restrictions.

We suppose that data are provided by cars and buildings in cities. This is not arbitrary, but responds to several rationales. Firstly, modern cars are already generating thousands of signals, used to monitor and optimise the functioning of the vehicle; and tests have already been performed on transmitting these data to both local networks [5] and central repositories [6, 7]. While less evident, the same applies to intelligent buildings, in which parameters like temperature and occupancy are constantly monitored [8]. Secondly, cars and buildings can be used to describe the status of a city with high resolution, as they cover most of our daily activities – note that cars can here be intended *lato sensu*, and include e.g. public transportation vehicles. Thirdly, it is easy to envision new services based on these data. To illustrate, weather-related data (including temperature, humidity, or presence of rain) can be used to create detailed weather maps, and hence high resolution weather forecast models; vehicle data can be used to monitor mobility, to detect traffic jams and suggest alternative re-routings in real-time; and, when combined with energy production and consumption trends, to optimise electric vehicle recharging. These three examples share two important features: the user of the CPS (i.e. the car driver or the building inhabitant) gets something in return from sharing the data, e.g. avoiding unexpected traffic jams; and these services would not be feasible without a large scale sharing of data.

In synthesis, the CPP scenario here considered has several characteristics that we expect to be common to most applications, namely:

- Large volumes of data, generated in a continuous (streaming) fashion.
- Redundancy of data, with the same (or similar) values generated by multiple CPSs.
- Potentially low input data reliability, but a required high output reliability.
- Confidentiality of data.

III. REQUIREMENTS FOR A DATA ANALYSIS TOOLBOX

The four characteristics listed above have important implications in terms of the requirements associated to a data analysis toolbox, which are discussed in what follows. For the sake of clarity, these have been organised in two groups: standard and CPP-specific requirements – see Fig. 1.

The review of existing literature, and especially of references [9, 10], has led us to the identification of several fundamental requirements for data handling and processing. We term these requirements as “standard”, in that they are not specific to CPPs, but are instead common to many data-handling systems. They are organised in three global types, depending on the operative condition of the system.

First of all, under normal conditions, the system will be expected to provide:

- Predictability: ability of foreseeing the system’s behaviour or functionality in the future, under standard operational condition. Such prediction could have a qualititative or, ideally, a quantitative nature.
- Accuracy: how close is the system’s observed behaviour with the one forecasted or expected from it.

From the point of view of the functioning of the system under disrupted conditions, the system will require:

- Dependability: the user can expect to be able to access the main system’s functionalities, even under disrupted operations, without a significant degradation.
- Maintainability: easiness of repairing the system in case of failure, and in general to maintain its optimal operational conditions. This includes both the resources needed, and the resulting downtime.
- Availability: capability of being ready for access even when disruptions occur.
- Adaptability: capability of the system to adapt its internal state and operation, in response to a change in the environment or in the users’ requirements, without losing its main characteristics (especially performance-wise).
- Possibility of distributing the computation, such that for instance a sensor may decide to transmit information only when the perceived environment changes, or it presents an unexpected behaviour. Besides reducing the workload of the central system, such distributed approach would also help tackling the problem of data confidentiality, as some personal information may be masked at the source.

Finally, when considering the dynamical nature of the system, and thus the possibility of expanding its scope in real-time, the following characteristics will be required:

- Reconfigurability: capability of the system to be easily changed to perform new functions or operations, following an external request. Note that this is conceptually different from adaptability, as in the latter case it is the system itself that decides to change its functioning, as opposed to in response to an external directive.
- Heterogeneity: the system should be able to internally incorporate heterogeneous components, as for instance different computational modules, in order to be able to cope with heterogeneous external inputs (as, for instance, the existence of heterogeneous data sources).
- Scalability: adaptability of the system to the specific problem of an increase (or decrease) in the workload. The system should thus be able to respond to this type of changes, and be able to dynamically use more or less resources as needed. To illustrate, at the beginning of operations, few users may join a new service; but this number may increase fast in case of commercial success. Note that, while scalability is usually understood in the sense of an increment, the opposite is also relevant, as it allows a saving of resources – for instance when a service is not expecting many users at night.

The second group includes requirements that are specific to the characteristics of CPS and CPP data, and that mostly derive from the description of the scenario presented in Section II:

- Capability of handling large quantities of dynamical data, sent by the different sensors composing the system in near real-time. In order to avoid excessive communication overheads, especially when leveraging on mobile communication networks, it may be necessary to send these data in packets. The system should then be able to analyse the incoming information both in streaming and in batches, and eventually to dynamically change between both operational modes.
- Capability of performing several types of analysis, including (but not limited to): supervised classification of different scenarios [11]; unsupervised identification of common patterns in the data [12]; time series forecast [13]; detection of anomalies and transitions in the dynamics of one or more systems [14].

- Possibility of distributing the computation, such that for instance a sensor may decide to transmit information only when the perceived environment changes, or it presents an unexpected behaviour. Besides reducing the workload of the central system, such distributed approach would also help tackling the problem of data confidentiality, as some personal information may be masked at the source.

IV. LIMITATION OF EXISTING SOLUTIONS

With the widespread adoption of data-based solutions in many real-world scenarios, it is not surprising to find a large number of analytic solutions, spanning from cloud pipelines to commercial and freeware software, and both stemming from research activities or having a commercial nature. It is thus only natural to explore the following question: should an analytics toolbox for CPP be made from scratch, or would it be more efficient to use one of these solutions? For the sake of completeness, the most important of them are here reviewed, with a special attention to their shortcomings with respect to the requirements of a CPP application. For the sake of simplicity, those solutions are grouped according to the place where the computation is performed – i.e. local vs. on the cloud.

On one hand, one can find both commercial and freeware software tools for data analysis that are designed to work on a local environment. In this category one can include, for instance, KNIME [15], SPSS Modeller [16], RapidMiner [17] or Alteryx. These software usually have a broad focus, allowing to process any (or most) kind of data. While prima facie an advantage, such generality can actually become a drawback. Specifically, most of the functions provided by such tools may not be relevant in a CPP context, and, conversely, some essential functions or algorithms may not be included. A clear example of the latter case are algorithms to detect changes in time series [18], which are not available in the mentioned packages. These are essential to detect when changes in the dynamics of the system occur, which in turn can be used to raise alerts or start computation tasks [19]. Additionally, while it is in principle possible to deploy such software tools in a cloud computing environment, the process is not straightforward, thus hindering the scalability of the system.

On the other hand, it is worth discussing a new approach being introduced by Google, Microsoft or Amazon, among others. This is based on full cloud environments, and in the creation of web-based pipelines in which data are fed, processed, and returned to the user in a completely automatic way. This approach presents two advantages: a complete scalability, and a simplified user experience. At the same time, it usually provides a limited spectrum of possible analysis - for instance, Google ML Engine completely relies on Tensor Flow algorithms [20]. In addition, data have to be transferred to the cloud platform, implying a loss of privacy and control over the data. In some cases the location of the data and of the processing may even not be transparent to the user, something in conflict with recent regulations on personal data protection [21, 22].
Finally, a third category of data analytics solutions is worth mentioning, which positions itself in between the two families previously described. While they are designed for a cloud deployment, they can easily be complemented by a local infrastructure; and they shift the focus towards an intuitive representation of the results and simplified user experience. Among others, these include Sisense, Looker, Zoho Analytics, or Tableau. They usually allow to summarise data on high-level dashboards, with specific applications including business analytics [23] or website usage tracking. They nevertheless do not provide the analytical flexibility required by a CPP application.

The previously mentioned toolboxes have not been developed with CPS applications in mind; their limits with respect to the requirements listed in Section III are therefore not surprising. Still, it is interesting to note that many CPS and CPP applications have been proposed in the literature (the reader may refer to [24, 25] for more exhaustive reviews), which may be used as a starting point for the present development. Nevertheless, in most cases these just included a proof of concept of CPP and CPS scenarios; they thus did not dig into the specificities of data analysis algorithms. In spite of this, a brief overview of three relevant projects is reported below, selected for presenting some commonalities with the CPP scenario here considered.

- Carweb [26]. This project proposed the integration of information coming from cars and mobile phones, focusing on the position and movements of users. The information was then integrated at a higher level, and presented in an interactive way to the user, through a webpage, for the identification of traffic jams in a city.

- SIPRESK [27]. Similarly to the previous project, here authors propose the integration of data coming from distributed sensors to dynamically detect congestion points in a city. With respect to the previous case, here more information sources are considered, including social networks and road sensors.

- CBMANET [28]. Integration and analysis of sensor data gathered from a battlefield. As opposed to the previous cases, here information is generated by fixed stations, and not by the vehicles themselves.

In synthesis, even though many data analytics solutions have been proposed during the last decade, as a result of the increasing relevance of data in business and scientific fields, these do not easily fit the requirements of CPP applications. At the same time, it is not trivial to learn from previous CPS research projects, as most of them did not go beyond the proof-of-concept phase.

V. CHALLENGES AND OPEN ISSUES

While many requirements listed in Section III are not new, and therefore techniques and technologies have been developed to deal with them, some are pushing the boundaries of standard solutions. In this last part we discuss some challenges and open issues, which may become the foundation of novel lines of research.

A. Information volume, communication bandwidth and data pre-processing

CPP systems are going to generate extremely large volumes of data, both due to the capacity of individual sensors to record measurements at very high frequencies, and to the high number of elements participating in the scenario. While current technology allows to process such massive volumes of data in economical ways, for instance through cloud computing, CPPs present an additional challenge: the bottleneck of transmitting the information from the individual sensors to the data processing system. In other words, the communication bandwidth may be limited, especially if costs have to be kept under control, and such limit will be tighter than those of any subsequent computational step. While communication limitations were part of previous studies of similar scope [5, 6, 7], none of them has considered future scenarios with data volumes of the order of GB per hour.

There are prima facie solutions to deal with this, and specifically to reduce the quantity of information to be transmitted by every sensor. Data can be pre-processed locally by each CPS, and transformed through sampling, windowing or dimensionality reduction. Several frameworks, integrating some of these techniques, are available in the literature [29]. Yet, the application of these techniques is far from trivial, especially because they usually require some a priori or external knowledge about the system. To illustrate, let us consider the case of a data sampling, in which only one every n measurements is transmitted. Choosing the value of n requires balancing the cost of transmitting potentially redundant information, with the necessity of analysing data with a high-enough time resolution. On top of that, the optimal n may not be constant, but may instead vary according to both the target applications and the dynamics of the system; and multiple elements may share the same infrastructure, such that each element’s n must be chosen to optimise a global (as opposed to a local) cost function.

The ideal solution will then require a pre-processing of the data agnostic to the final application, but still optimal in some sense; for instance, a reduction of the data volume that does not reduce the actual quantity of encoded information. While some mathematical instruments are available for this, e.g. entropies [30, 31], their applicability in real-world scenarios has not hitherto been tested.

B. Security: avoiding malicious uses

As any other large-scale infrastructure, CPP systems are going to be vulnerable to external attacks. Besides external attackers (or hackers) that may try to illicitly access the system, there is also the threat of internal elements or users trying to disrupt its functioning. We will here consider the simplest situation of a sensor being programmed to transmit false information - note that this may also be due to a hardware malfunction, and not necessary to a malicious intention.
The solution to this scenario may come from the data integration and fusion step. Once multiple sensors have sent their corresponding data, these should be integrated, in order to recreate a coherent picture of the system. The system may then realise that a sensor is unreliable, or that has come under external attack [32, 33], and thus exclude it from the collective [34, 35, 36]. The simplest solution involves selecting between conflicting options through, for instance, a majority vote. It has nevertheless to be noted that this does not protect against coordinated attacks, in which multiple sensors are programmed to yield erroneous yet coherent information. Additionally, while not essential in the scenario discussed in Section II, security may become critical in the future, especially if CPS data will be used to support safety-critical applications.

C. Privacy of information

A major challenge related to CPP systems is the confidentiality of the information by them handled. As previously mentioned, information can be private due to what can be inferred about the behaviour of users, but also about the internal functioning of CPSs.

The standard approach to confidentiality involves, on one hand, ensuring that no private information leaves the system, being thus related to the previous security challenge. On the other hand, it also requires trust, as for instance the service providers have to use data only for the analysis tasks they are authorised to do. This second point may be problematic in a CPP environment, as users have no way of checking the service providers’ activity – in other words, they have to blindly trust them.

An alternative solution may be offered by the rising field of secure multiparty computation (SMC) [37, 38]. SMC is based on a set of cryptographic protocols that allows a group of parties (or participants) to perform a computation over a data set without actually accessing the data themselves. In other words, the computation is performed over encrypted data, such that the only information gained by parties is the final result.

The use of SMC would yield an important advantage: as CPSs would only share encrypted information, service providers cannot gain any knowledge, and therefore do not have to be trusted by the users. In other words, and to be more precise, the problem of trust is shifted from the service providers to the SMC protocols themselves. On the other hand, SMC also presents some important challenges. First of all, no practical general protocol (what is usually called fully homomorphic encryption) has hitherto been proposed [39]; instead, specific protocols exist only for specific applications. Secondly, the computational cost of SMC is usually significant [40], especially because part of the computation has to be performed locally at each CPS. Thirdly, different SMC protocols guarantee different levels of security, e.g. against just one, a few or a majority of malicious participants. In synthesis, while SMC can be a promising future solution, it still does not provide a universal and well-defined answer to CPP privacy concerns.

VI. CONCLUSIONS

The data availability offered by the widespread adoption of CPSs will soon trigger a revolution, in which new services based on them will become part of our daily lives. This will nevertheless not come for free: data have to be analysed, and existing analytics solutions are not designed to deal with the idiosyncrasies of CPPs. As discussed in this contribution, aspects like data preprocessing and compression, security, and privacy, are not easily dealt with by current technologies, and will require future research work.

ACKNOWLEDGMENT

This paper presents work developed in the scope of the project Cross-CPP. This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 780167. The content of this paper does not reflect the official opinion of the European Union. Responsibility for the information and views expressed in this paper lies entirely with the authors.

REFERENCES

[1] W. Wolf, “Cyber-physical systems,” Computer, vol. 3, pp. 88-89, 2009.
[2] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber-physical systems: the next computing revolution,” In Design Automation Conference, pp. 731-736, IEEE, 2010.
[3] T. Jeske, “Floating car data from smartphones: What google and waze know about you and how hackers can control traffic,” Proc. of the BlackHat Europe, pp. 1-12, 2013.
[4] A. Correia, C. Wolff, M. Zanin, E. MenaSalvas, E. A. Herrmann, V. Corral, M. Kachelmann, R. DeLong, and P. Smrz, “Cross-cpp - an ecosystem for provisioning, consolidating, and analysing big data from cyber-physical products,” in Proceedings of the 1st Workshop on Cyber-Physical Social Systems co-located with the 9th International Conference on the Internet of Things (IoT 2019), 2019.
[5] H. Huang, M. Hassan, G. Geers, and L. Libman, “CarAgent: Collecting and disseminating floating car data in vehicular networks,” Vehicular Communications, vol. 13, pp. 114-127, 2018.
[6] G. Leduc, “Road traffic data: Collection methods and applications,” Working Papers on Energy, Transport and Climate Change, vol. 1, pp. 1-55, 2008.
[7] K. Rehrl, R. Brunauer, and S. Gröchenig, “Collecting floating car data with smartphones: results from a field trial in Austria,” Journal of Location Based Services, vol. 10, pp. 16-30, 2016.
[8] Z. Han, R. X. Gao, and Z. Fan, “Occupancy and indoor environment quality sensing for smart buildings,” in 2012 IEEE international instrumentation and measurement technology conference proceedings, pp. 882-887, IEEE, 2012.
[9] E. Simon, K. S. Kim, E. Subrahmanian, R. Lee, F. De Vaulx, Y. Murakami, K. Zettu, and R. D. Sriram, “A vision of cyber-physical cloud computing for smart networked systems,” Gaithersburg, Maryland: US Department of Commerce, National Institute of Standards and Technology, 2013.
[10] V. Gunes, S. Peter, T. Givargis, and F. Vahid, “A survey on concepts, applications, and challenges in cyber-physical systems,” KSII Transactions on Internet & Information Systems, vol. 8, 2014.
[11] T. Hastie, R. Tibshirani, and J. Friedman, The elements of statistical learning: data mining, inference, and prediction, Springer Science & Business Media, 2009.
[12] H. B. Barlow, “Unsupervised learning,” Neural computation, vol. 1, pp. 295-311, 1989.
[13] J. D. Hamilton, Time series analysis, New Jersey: Princeton, 1994.

[14] S. Agrawal, and J. Agrawal, “Survey on anomaly detection using data mining techniques,” Procedia Computer Science, vol. 60, pp. 708-713, 2015.

[15] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Köetter, T. Meinl, P. Ohl, K. Thiel, and B. Wisswedel, “KNIME—the Konstanz information miner: version 2.0 and beyond,” ACM SIGKDD explorations Newsletter, vol. 11, pp. 26-31, 2009.

[16] T. Wendler, and S. Größtrup, Data mining with SPSS modeler: theory, exercises and solutions, Springer, 2016.

[17] M. Hofmann, and R. Klinkenberg, RapidMiner: Data mining use cases and business analytics applications, CRC Press, 2016.

[18] S. Aminnakhahahi, and D. J. Cook, “A survey of methods for time series change point detection,” Knowledge and information systems, vol. 51, pp. 339-367, 2017.

[19] E. Keogh, S. Chu, D. Hart, and M. Pazzani, “Segmenting time series: A survey and novel approach,” in Data mining in time series databases, pp. 1-21, 2004.

[20] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, “Tensorflow: A system for large-scale machine learning,” in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI) 16, pp. 265-283, 2016.

[21] M. Goddard, “The EU General Data Protection Regulation (GDPR): European regulation that has a global impact,” International Journal of Market Research, vol. 59, pp. 703-705, 2017.

[22] S. Bu-Pasha, “Cross-border issues under EU data protection law with regards to personal data protection,” Information & Communications Technology Law, vol. 26, pp. 213-228, 2017.

[23] H. Chen, R. H. Chiang, and V. C. Storey, “Business intelligence and analytics: From big data to big impact,” MIS quarterly, pp. 1165-1188, 2012.

[24] F. J. Wu, Y. F. Kao, and Y. C. Tseng, “From wireless sensor networks towards cyber physical systems,” Pervasive and Mobile computing, vol. 7, pp. 397-413, 2011.

[25] V. Gunes, S. Peter, T. Givargis, and F. Vahid, “A survey on concepts, applications, and challenges in cyber-physical systems,” KSII Transactions on Internet & Information Systems, vol. 8, 2014.

[26] C. H. Lo, W. C. Peng, C. W. Chen, T. Y. Lin, and C. S. Lin, “Carweb: A traffic data collection platform,” in The Ninth International Conference on Mobile Data Management (mdm 2008), pp. 221-222, IEEE, 2008

[27] H. Khazaee, S. Zareian, R. Veleda, and M. Litou, “Sipresik: A big data analytic platform for smart transportation,” In Smart City 360°, pp. 419-430, Springer, 2016.

[28] L. A. Tang, J. Han, and G. Jiang, “Mining sensor data in cyber-physical systems,” Tsinghua Science and Technology, vol. 19, pp. 225-234, 2014.

[29] G. D. F. Morales, and A. Bifet, “SAMOA: scalable advanced massive online analysis,” Journal of Machine Learning Research, vol. 16, pp. 149-153, 2015.

[30] M. Gell-Mann, and S. Lloyd, “Information measures, effective complexity, and total information,” Complexity, vol. 2, pp. 44-52, 1996.

[31] M. Zarin, L. Zunino, O. A. Rosso, and D. Papo, “Permutation entropy and its main biomedical and econometrics applications: a review,” Entropy, vol. 14, pp. 1553-1577, 2012.

[32] J. Newsome, E. Shi, D. Song, and A. Perrig, “The sybil attack in sensor networks: analysis & defenses,” in Third international symposium on information processing in sensor networks, 2004. IPSN 2004, pp. 259-268, IEEE, 2004.

[33] J. Park, R. Ivanov, J. Weimer, M. Pajic, and I. Lee, “Sensor attack detection in the presence of transient faults,” in Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems, pp. 1-10, 2015.

[34] J. Chen, S. Kher, and A. Somani, “Distributed fault detection of wireless sensor networks,” in Proceedings of the 2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks, pp. 65-72, 2006.

[35] E. M. Clarke, and P. Zuliani, “Statistical model checking for cyber-physical systems,” in International Symposium on Automated Technology for Verification and Analysis, pp. 1-12, 2011.

[36] L. A. Tang, X. Yu, S. Kim, Q. Gu, J. Han, A. Leung, and T. La Porta, “Trustworthiness analysis of sensor data in cyber-physical systems,” Journal of Computer and System Sciences, vol. 79, pp. 383-401, 2013.

[37] Y. Lindell, “Secure multiparty computation for privacy preserving data mining,” in Encyclopedia of Data Warehousing and Mining, pp. 1005-1009, IGI Global, 2005.

[38] R. Cramer, J. B. Damgård, and J. B. Nielsen, Secure multiparty computation, Cambridge University Press, 2015.

[39] M. Naehrig, K. Lauter, and V. Vaikuntanathan, “Can homomorphic encryption be practical?,” in Proceedings of the 3rd ACM workshop on Cloud computing security workshop, pp. 113-124, 2011.

[40] M. Zarin, T. T. Delibasi, J. C. Triana, V. Mirchandani, E. A. Pereira, A. Enrich, D. Perez, C. Paşaoglu, M. Fidanoglu, E. Koyuncu, and G. Guner, “Towards a secure trading of aviation CO2 allowance,” Journal of Air Transport Management, vol. 56, pp. 3-11, 2016.