Agent-based Internet of Things: State-of-the-art and Research Challenges

Claudio Savaglio[a], Giancarlo Fortino[a], Maria Ganzha[b], Marcin Paprzycki[c], Costin Bădică[d], Mirjana Ivanović[e]

[a]*DIMES, Università della Calabria, Rende, Italy*
[b]*Warsaw University of Technology, Warsaw, Poland*
[c]*Systems Research Institute Polish Academy of Sciences, Warsaw, Poland*
[d]*CIT, University of Craiova, Craiova, Romania*
[e]*DMI, University of Novi Sad, Novi Sad, Serbia*

Abstract

The disruptive potentials of the Internet of Things (IoT) entails multifaceted requirements and development issues (large scale deployments, heterogeneity, cyberphysicality, interoperability, distributed smartness, self-management, etc.). To adequately tackle them, and comprehensively support the development of the IoT ecosystem, the Agent-Based Computing (ABC) represents a proper and solid modeling, programming and simulation paradigm. Indeed, abstractions, design methods, technology and frameworks related to the ABC have been widely exploited, possibly jointly with other well-established/emerging computing paradigms, to actually develop advanced IoT ecosystem. This survey, an extension of our previous work, reports most relevant contemporary contributions in the field, aiming at assessing suitability of the ABC paradigm for the (current and future) IoT development.

Keywords: Software Agents, Internet of Things, IoT

Email addresses: csavaglio@dimes.unical.it (Claudio Savaglio), g.fortino@unical.it (Giancarlo Fortino), m.ganzha@mini.pw.edu.pl (Maria Ganzha), marcin.paprzycki@ibspan.waw.pl (Marcin Paprzycki), cbadica@software.ucv.ro (Costin Bădică), mira@dmi.uns.ac.rs (Mirjana Ivanović)

1. Introduction

In the last two decades, advancements in embedded processing, sensing, actuation, and wireless communication rapidly fueled the spread of novel cyberphysical artifacts and applications, among others, for the ambient assisted living and wellness, entertainment, logistic optimization, energy management, industrial automation. Indeed, simple movement detectors, radio-frequency tags, temperature sensors, but also more sophisticated smart gadgets, smartphones and smart vehicles, allow sensing the physical world, processing data, engaging customized users interactions and triggering actions over the surrounding environment. Also known as “Smart Objects” (SOs) [2] because of the (different degrees of) intelligence they exhibit, and being massively networked both on local and global scales, such heterogeneous devices communicate and cooperate with each other and with conventional systems, thus constituting the Internet of Things (IoT) [1] ecosystem.

The complex development of heterogeneous IoT ecosystems requires comprehensive support from different mainstream paradigms and approaches, especially from the closely related fields of wireless sensor networks, distributed systems, artificial intelligence, ubiquitous and pervasive computing [3]. In particular, the Agent-based Computing (ABC) [4] has been acknowledged as a comprehensive, effective enabler for cooperating, decentralized, dynamic and open IoT ecosystems. In this paper, we discuss how ABC is, currently and effectively, exploited to develop IoT ecosystems. As matter of fact, the ABC provides ideas, metaphors, techniques, methods and tools for systematically conceptualizing, programming and simulating distributed systems composed of heterogeneous interacting entities [15]. Following an extended period of intense research and development, in the last decade the ABC seemed sidelined. Unexpectedly, the ABC recently found new interest and application in the IoT scenario and the marked interest of the research community in the agent-IoT duo [6] is demonstrated by the relevant number of publications materializing along this line (almost 1000 in the last three years, according to Scopus, just considering international peer-reviewed journals and conference papers; a detailed bibliometry is reported in Appendix A) and the presence of the ABC among the IoT enablers outlined within the European H2020 Work Programme 2018-2020[1] as well as within some U.S. National

[1] https://ec.europa.eu/programmes/horizon2020/en/h2020-section/advanced-computing

Science Foundations ongoing Programs[2,3] and awarded projects[4,5]. Such continuous research interest motivated us to extend our previous publication [9], from 2017, by surveying *(i)* the most up-to-date and relevant related work, proposing agents as IoT development enablers; *(ii)* the rise of synergies between the ABC and other mainstream paradigms/technologies, which are independent from IoT but beneficial for its development, e.g., Cloud and Edge Computing, Wireless Sensor Network, Machine Learning, Blockchain and Semantic technologies; and *(iii)* existing IoT applications and derivations, successfully enabled by the ABC, in novel field s like the Internet of Vehicles, Industrial IoT and Social IoT. Due to the extent of existing state-of-the-art material, providing an exhaustive analysis of all agent-based IoT contributions and/or agent-related IoT application scenarios seems unfeasible. Instead, we aim at highlighting main agents' pros and cons for the development of both SOs and IoT systems, surveying most relevant works.

The outline of this paper, graphically reported in Fig.1, is as follows. Section 2, provides selected insights about the principal IoT development challenges and ABC distinctive features. Section 3 follows with a survey of several contributions exploiting the ABC to model, program and simulate SOs and IoT systems. Next, in Section 4, selected synergies between the ABC and other, both well-established and emerging, paradigms are shown, while in Section 5 examples of successful agent-based IoT applications are presented. Finally, the surveyed state-of-the-art is briefly analyzed, the open challenges discussed and some concluding remarks provided.

2. Background

2.1. Internet of Things Development Issues and Requirements

Multifaceted development issues must be faced, and related requirements addressed, in order to actually unfold the disruptive IoT potential [3]. *Heterogeneity* is the clearest issue featuring the kaleidoscopic IoT ecosystem [149], constituted by a plethora of different components (e.g., NFC tags, sensors, microcomputers, wearable gadgets, industrial robots, domotic devices, vehicles) and stakeholders (individuals, companies, public administrations). Therefore, providing *interoperability* to these heterogeneous subjects, connected through various communication

[2] https://www.nsf.gov/pubs/2018/nsf18557/nsf18557.html
[3] https://www.nsf.gov/pubs/2019/nsf19566/nsf19566.html
[4] https://govtribe.com/award/federal-grant-award/project-grant-1703782
[5] https://www.nsf.gov/awardsearch/showAward?AWD-ID=1659774

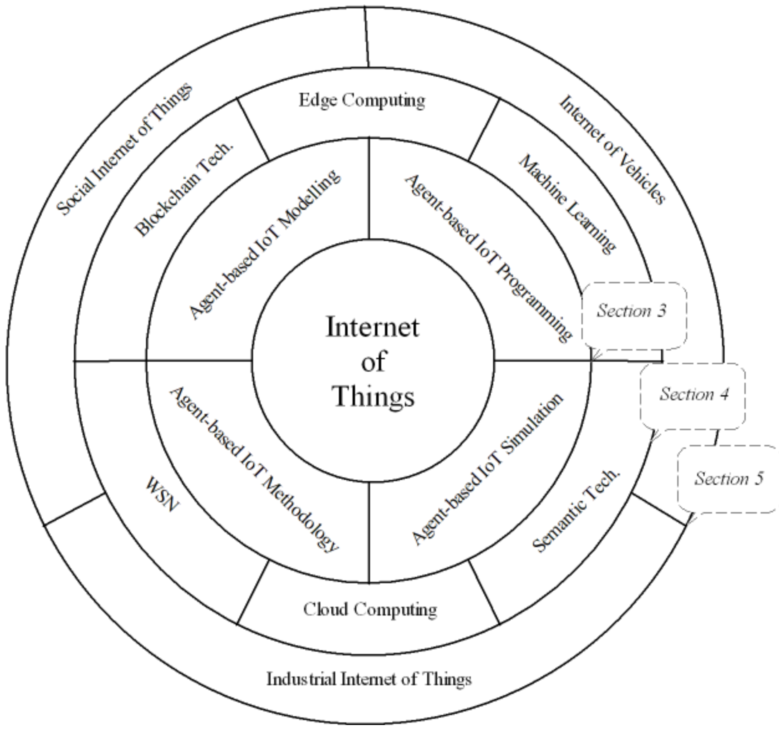

Figure 1: Graphical outline of the survey

technologies (Bluetooth, Zigbee, 3/4/5G, Wi-Fi, etc.), and deployed in networks of different scales (ranging from small-scale personal area networks up to large-scale industrial networks and extreme-scale, very dense metropolitan area networks), is a requirement necessary but challenging to meet. This is so, especially considering that the lack of established standards pushes towards poorly interoperable "intra-nets of things" [150]. Likewise heterogeneity, also *cyberphysicality* is entailed within the IoT concept. Indeed, it is located at the border between virtual and real worlds, where SOs give rise to new cyberphysical functionalities and development issues, unforeseen or superficially treated by conventional computer engineering (cyber-physical security, personal data privacy, etc.). Hence, heterogeneity and cyberphysicality motivate the use of multidisciplinary [16] and comprehensive *methodologies* [17]. These are used also for lowering time-to-market, efforts and probability of failure. The *size* and *rapid evolution* of IoT ecosystem are other two critical development issues given that, according to[6], 20.4 billion of "things" will be networked in 2020. Operations, like things/services naming, discovery and deployment, would be definitively challenging in such a dense and dynamic scenario [1]. Therefore, *self-steering* and *decentralization* become imperative requirements for things and IoT systems, since their human-based, or centralized, management will bring result that are definitively unfeasible [53] (e.g., for the sake of bandwidth saving, an SO should be able, by design, to dynamically adjust its communication patterns; moreover, newly introduced SOs need to be automatically interfaced to and integrated into already deployed applications). Besides self-steering, SOs (which are different from simple resources such as RFID/NFC tags, databases, sensors and actuators) are expected to be *high-performing*, in terms of intelligence, reliability and context-awareness. Similar desiderata should be satisfied by all IoT systems that should expose autonomic, scalability and openness features [148]. Fulfilling these requirements would allow demanding IoT applications (e.g., augmented reality, industrial applications, emergency management, real-time systems) to provide high Quality of Experience (QoE) even in the presence of issues such as *SOs resource scarcity* or *poor infrastructures*. Indeed, without a satisfactory *usability*, the acceptance of novel SOs and IoT applications would be definitively compromised.

Summarizing, a number of development issues and related requirements makes the development of IoT ecosystems extremely challenging. Also other kinds of issues, concerning ethics, business and social sciences, are well known but disregarded because out of the scope of this paper. Nevertheless, the great potentials of IoT still make most of consumers, governments and companies seeing opportunities instead of threats.

2.2. Agent-Based Computing Paradigm

Agent [4], as a sophisticated software abstraction defining an autonomous, social, reactive and proactive computational entity, is the key concept of the Agent-based Computing (ABC) paradigm. Likewise, MASs (Multi Agent Systems) [4] are ensembles of agents, which interact and cooperate in a certain environment (namely, the world of perceived resources), thus constituting distributed and self-steering societies featured by a strong situatedness and well-defined organizational relationships. Without claiming to be exhaustive, the above characterization already suggests that agent-related key abstractions allow for straightforwardly

[6]https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/

modeling complex systems, their components, interactions and organizational relationships, covering variety of domains (logistics, sociology, economy etc.).

Beside s modeling, ABC provides also a well-established *programming* paradigm for implementing agents' advanced features, and effectively addressing key requirements, typical of modern distributed applications. Indeed, agent's, society's and environment's modeling abstractions have outlined a high-level, distributed programming paradigm, based on two milestones [18]: (1) encapsulation of control (each agent has its own reasoning capabilities and thread of control to expose context-aware and autonomous behaviors), and (2) interaction (coordination and cooperation typically based on high-level asynchronous message passing mechanisms). Indeed, shared communication standards and management specifications (like the Foundation for Intelligent Physical Agents FIPA platform [19] and Agent Communication Language ACL [20]) render agents interoperability facilitators and allow incorporating different resources and existing legacy systems within the agent society. By fostering computational efficiency, reliability, responsiveness, interoperability and scalability (particularly compared to centralized approaches), the agent-based programming paradigm allows implementing advanced applications while enhancing system s' performance.

Finally, agent-based systems can be straightforwardly simulated in order to study both emergent, individual patterns or collective phenomena [21]. By focusing on individual agents, their behaviors, and interactions (which reflect the ones of the real world), agent-based *simulation* is a natural approach for understanding and managing the global dynamic of complex systems, like dynamic distribution and supply networks, social sciences. Indeed, even in highly scaling-up and interacting contexts, agent-based simulation facilitates the evaluation of distributed systems which expose discrete, not linear, adaptive behaviors.

Summarizing, the ABC is a rich and complex source of metaphors, techniques and tools: therefore, to provide a systematical approach to the agent-based modeling, programming, and simulation, several agent-oriented development *methodologies* have been designed and successfully applied [22]. However, as underlined in [23], [25], [24] and in Section 5, the ABC is neither a universal nor necessarily effective development solution, and its adoption needs to be carefully assessed. Indeed, both agent-level and society-level pitfalls can occur from different perspectives (e.g., management, conceptual, design) and these can outweigh every agent-related benefits.

3. Agents' Contribution in Development of IoT Systems

Complex, dynamic, situated and autonomous systems, both natural and artificial, can be naturally approached by adopting the agent-oriented perspective [15]. Based on the strong conceptual alignment between agents/MAS and SOs/IoT systems [59], and according to the match between IoT development requirements and agent-related benefits, the ABC has been exploited to systematically drive and speed-up the development of SOs and IoT systems. An overview about the

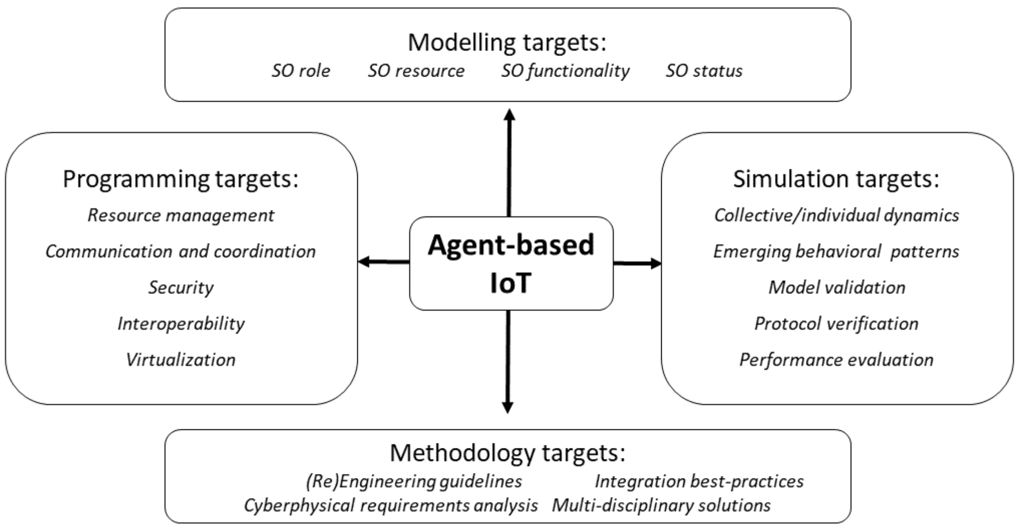

Figure 2: High-level modelling, programming, simulation and methodology targets of the agent-based IoT.

high-level modelling, programming, simulation and methodology targets of the agent-based IoT is shown in Fig.2, while main contributions which exploit the ABC for modeling, programming and simulating purposes have been surveyed and compared in Tab. 1. This table summarizes findings contained in subsections 3.1-3.4, and indicates for each contribution if *(i)* a fine/coarse grained agent-based modeling of IoT entities is performed, *(ii)* mechanisms for (technological/ syntactical/ semantic) interoperability, autonomicity, cognitivity, virtualization or security are implemented, *(iii)* pure or hybrid activity simulation is supported, and *(iv)* an agent-based IoT development methodology is provided. The definitions of these *keywords* is reported in what follows.

Table 1: Relevant agent-based works with their features - *T*echnological, *Sy*ntactical, *Se*mantic interoperability; *A*utonomicity; *C*ognitivity; *V*irtualization; *S*ecurity.

Surveyed work <name, ref. >	**Agent-based IoT model**		**Agent-based IoT implementation**							**Agent-based IoT simulation**		**Agent-based IoT Methodology**
	Fine grained	*Coarse grained*	*T*	*Sy*	*Se*	*C*	*A*	*V*	*S*	*Pure*	*Hybrid*	
Cascadas, [26]	✓		✓	✓			✓					
iCore, [30]	✓		✓	✓	✓	✓			✓			
FIoT, [31]	✓		✓	✓		✓	✓	✓				
CACB, [32]		✓					✓	✓			✓	
[33]		✓					✓	✓			✓	
ACOSO [14], [17], [34], [148], [147]	✓		✓	✓	✓	✓	✓	✓			✓	✓
UBIWARE, [28]; UBIROAD, [38]	✓		✓	✓	✓	✓		✓	✓			
[41]		✓	✓	✓				✓				
[60]	✓		✓	✓	✓		✓	✓				✓
AoT, [39]		✓	✓	✓		✓	✓	✓				
Smart Grids, [56]		✓		✓						✓		
[57]		✓	✓	✓				✓			✓	
TAEC, [55]		✓	✓	✓	✓		✓	✓	✓			
CIoT, [44]	✓		✓	✓	✓	✓		✓	✓			
iSapiens, [35]	✓		✓	✓				✓	✓			
[43]		✓	✓	✓				✓				
Radigost, [54]		✓	✓	✓				✓				
ASSIST, [61]	✓		✓	✓	✓	✓		✓		✓		
BEMOSS, [37]		✓	✓	✓				✓	✓			
INTER-IoT, [66]	✓		✓	✓	✓	✓	✓	✓	✓		✓	✓
VICINITY, [45]	✓		✓	✓	✓	✓		✓	✓			
SOL, [40]	✓		✓	✓	✓		✓					
[29]	✓		✓	✓	✓							
[36]		✓	✓	✓			✓					
[42]		✓	✓	✓		✓	✓					
Smart Santander, [47]		✓	✓	✓	✓			✓	✓			
[46]		✓	✓	✓	✓			✓	✓			
Prometheus, [62]	✓		✓	✓	✓			✓	✓			✓
ASEME, [27]	✓		✓	✓	✓			✓	✓			✓
SAMSON, [67]	✓						✓			✓		
[81]		✓			✓	✓	✓			✓		
[50], [51], [52]		✓	✓	✓			✓	✓	✓	✓		

3.1. ABC as IoT Modeling Paradigm

Key features of both SOs and IoT systems can be captured, at different degrees of granularity and in a technology-agnostic way, through the agent-based modeling. Indeed, the agent model naturally embed s SO autonomicity, proactiveness and situatedness, while other important SO features can be explicitly described through agent-related concepts. For example, in [26], [36] and [67], SO functionality is expressed as goals, SO working plans as behaviors, and SO augmentation devices (i.e., knowledge bases, sensors and actuators) as dynamically bindable agent resources. However, these working plans adopt different mechanisms for specifically characterizing SOs/agents. In [27, 29], for example, SO/agent behaviors, goals and communications paradigms depend on their own role, which is taken from context-dependent repositories (e.g., for the transportation, smart car, smart driver-support, smart road). Likewise, in [30] and [31], each SO/agent complies to a template, which encode s plans and goals according to its functionality. Differently, there are some contributions, which do not provide a-priori defined roles or templates. In [26], each SO/agent has its own model (formalized as an automaton) which, on the basis of perceived stimuli (modeled as messages incoming from the environment or other entities), dictates the actions to be performed. Likewise, in [34, 35], SOs/agents' behaviours encode both its design goals (encapsulated in state-based tasks) and its (re)actions to the risen internal and external events. Finally, real-time sensor data, position and resources availability dynamically determine the state of the SO/agent in [36]. Apart from the specific modeling approach, all the surveyed agent-oriented approaches provide high-level models to effectively abstract principal SOs transparently from a specific technology, thus supporting the preliminary development phase of analysis. However, beside s the satisfactory "per-se" agent-based SOs modeling, further research efforts are necessary to thoughtfully model relationships among agents/SOs interacting within physical environments. For example, [80] describes opportunities and limitations (along with the current and potential socio-cultural impact) of a truly networked and cooperative SO-based IoT based on the "agency" metaphor.

Although concepts such as negotiation, competition, cooperation and delegation, between agents and other entities, have complex definitions, the ABC still represents an effective IoT modeling paradigm, as well as a profitable baseline for subsequent phases of agent-oriented IoT programming and simulations [17].

3.2. ABC as IoT Programming Paradigm

The marked heterogeneity of resources and communication protocols, featured in the IoT ecosystems, motivated an agent-oriented approach to programming uniform interfaces for transparently interacting with different SOs and resources. Software adapters, internally coordinated by a device manager [34], and purposely developed for target technologies, allow accessing and managing augmentation devices of SO/agent in [28], [30]-[34], [38]. Being developed as pluggable and customizable software components, these adapters ensure modularity and extensibility of system programming. A completely different approach is adopted in [29], [37], in which specific agents are deputed to interface resource with the related SO and with the system. Although transparent to technological heterogeneity, this solution does not work for those constrained devices unable to support a complete agent-based architecture.

Beyond facilitating resource management, agent-oriented programming promotes communication and coordination in the IoT ecosystem (i) directly, by implementing the IEEE FIPA 'de facto' standard specifications [19], and (ii) indirectly, by supporting the SOs *Virtualization* [30] for the sake of a major accessibility and integration of the agentified SOs. With respect to the first point, FIPA specifications provide standardized message format and content, an effective message transport service, and both semi/centralized and distributed services of agent discovery. The ACL [20] allows encoding message envelope: the message content, instead, is typically expressed through metadata-oriented languages and ontology to facilitate both context management and data exchange. With respect to the second point, the functionalities of an agentified SO can be transparently accessed over standard, platform- and language-independent Internet protocols (e.g., SOA and REST [43], [54]) and then exploited as a monolithic Web Service [39] or as an ensemble of small, loosely-coupled and distributed microservices [49]. Just microservices are currently on the hype due to the agility they provide for the development of a IoT system and the agent-based programming can provide them further features like autonomy, social ability and elasticity [50], [51], [52].

In brief, agent-based programming fosters:

- *Technical interoperability*, by means of shared resource and communication interfaces [26], [34], [40], [42], [47], [54] (though, in some cases agents belonging to different organizations are not totally interoperable [13]),

- *Syntactical inteoperability*, by means of a shared message format, because ACL is adopted across FIPA standard obeying platforms for message envelope, while XML and JSON are used for message content in [28, 29], [41], [43], [45], [47] (but it is worth noting that ACL is a 'de facto' standard but not the only language [23]); and

- *Semantic interoperability*, by means of shared ontology and knowledge representation [28, 30], [38], [44], [46, 48] (but this feature is quite limited and underdeveloped because of the shortage of grounded domain-specific ontology and semantics [25]).

At a higher level of conceptualization, agents allow to straightforwardly instill smartness and autonomy within a single SO, and realize cognitive and autonomic IoT systems [14], [53]. Indeed, agent-based programming fosters (i) *Autonomicity*, in terms of self-configuring, self-healing, self-protecting and self-optimizing SOs/IoT systems which require minimum human interventions for their management [26], [40], [42], [55]; and (ii) *Cognitivity*, in terms of context-aware and adaptive SOs and IoT systems [30], [44, 45], able to solve complex problems autonomously, if properly trained. Apart from ensuring self-management and distributed intelligence, autonomic and cognitive features are also key to implement advanced *Security*, for example through (un)conventional trust mechanisms (see Social IoT in Section 5) aiming at a secured large-scale IoT ecosystem.

3.3. ABC as IoT Simulation Paradigm

A number of contingent factors [67] (SOs density, network design, irregular traffic, wireless coverage, etc.), the cost of the hardware and its installation, along with the scarcity of professional installers, make the IoT ecosystem deployment complex, costly, error prone and time consuming, especially in large-scale scenarios (e.g., a smart city in which a smart parking system could require very high budgets). Therefore, the simulation activity plays a crucial role [147], enabling comprehension of collective/individual dynamics and the estimation, validation and verification of performance, models and protocol before the deployment phase.

Pure agent-based simulators allow effectively inspecting high-level aspects such as the rise of collective dynamics and behavioral patterns in large scale, distributed, event-based IoT system [82] [84]. JADE has been used in [85] to simulate a Smart City in which heterogeneous "agentified" SOs, able to act autonomously and collaborate, dynamically consume and/or produce energy. In [156], authors integrate agent-based simulation and evolutionary game theory to analyze the cooperative patterns in the IoT scenario, the dynamic process and macro emerging actions. Through a set of simulations carried out on Anylogic, authors evaluate conditions leading certain business models to dominate the IoT market. In [152], [83] and [151], the research focus of the simulation is on IoT services, considered as MASs. Authors of [152] exploit agent-based simulation to understand and visualize the quality of service and evaluate several discovery strategies considering different communication topologies. In [151], instead, authors propose an Agent-Oriented Petri Net to analyze the dynamic behavior of services by performing model checking and, hence, exhaustively and automatically check specifications and properties. Finally, Netlogo, has been exploited in [81] to aggregate IoT services under complex users' requests and to manage real-time traffic information in [83].

Although pure agent-based simulators can successfully validate SOs interactions and operations, they typically outline quasi aseptic simulation environments, distant from the real cyberphysical IoT ecosystem, because low-level communication and mobility issues are often neglected or coarsely handled [56], [61]. Therefore, a research line foresees a hybrid approach, based on joint exploitation of agent-oriented modeling and network-based simulation. In particular, agent's logic is implemented within event-based network simulators like OMNeT++ [14, 57, 59, 147], ASSIST [61] and SenseSim [157], because SOs/agents interactions are asynchronously event-driven and time-dependent. Such hybrid simulation approach allows mitigating limitations of pure agent-based simulation (but maintaining advantages derived from ABC) and effectively simulating IoT systems of different scales (ranging from narrow local networks up to widespread Smart City), with different mobility patterns, interaction protocols, communication parameters, etc. [14], [58, 59]. In this way, both qualitative and qualitative aspects can be assessed, providing a preliminary and reliable overview of operation, of an IoT system to be developed.

Summarizing, few computing paradigms deal with IoT simulation (e.g., Aggregate Computing [153], SOA [154]) and the hybrid agent-based approach represents one of the more valid candidates. However, as opposed to the well-established modeling and programming, agent-based IoT simulation is at an early stage, while IoT-specific simulators are not currently available. Nevertheless, overlooking this aspect could represent a crucial pitfall compromising the agents acceptance in the IoT scenario [155], considering that, without a central control, unexpected and emergent behaviors are likely to appear within MASs [63].

3.4. Agent-Based Methodology for IoT

The peculiar features of the IoT context demand agent-oriented methodologies to be specifically extended [27], [62], or ex-novo designed [17], [60]. Indeed, these methodologies can bring about usage of agent-based models, programming techniques and simulation tools [62]. Moreover, at the same time, they consider aspects that are usually neglected by conventional agent-based methodologies. Therefore, for comprehensively driving the IoT ecosystem development, the aforementioned agent-based methodologies:

- thoroughly face the cyberphysicality of SOs and IoT, providing by design solutions for interoperability, security and scalability [17], [60], [62];
- seamlessly glue hardware and software components by defining, typically at the middleware layer [64, 65], suitable coordination, virtualization and management techniques [17];
- jointly analyze specific SOs and IoT system requirements [27], [60] with the infrastructural features and limitations of the development context;
- accommodate different perspectives and expertise, exploiting both technical (e.g., Unified Modeling Language [27] and Business Process Model Notation [62]) and not-technical (textual descriptions [60]) notations to depict the IoT ecosystem, its relevant use cases, users, services, stakeholders, etc.;
- drive and promote the integration of different computing paradigms for IoT (see Section 4) in different application contexts (see Section 5) through guidelines and best-practices, possibly unbound from specific protocols or technologies [17], [66].

Without extensively facing such aspects, any development methodology, even if effective in conventional agent-based context (for example, Tropos [68]), would fail in supporting the IoT ecosystem development. Finally, it is worth noting that agent-based methodology facilitates and speeds-up not only the development of novel SOs/IoT systems, but also the re-engineering of existing ones. Indeed, as shown in [69], agent-based methodologies can drive SO/IoT system re-engineering, in order to highlight and enhance both functional (e.g., support to interoperability, attention for resource-constrained SOs) and non-functional (e.g., modularity, maintainability, evolvability) features which generate fundamental benefits for the complex, heterogeneous and constantly evolving IoT scenario.

4. Integration of agents with other paradigms and technologies for IoT

As reported in the previous Section, the ABC can enable the SOs' and IoT systems' development; moreover, the ABC is also prone to be beneficially integrated with other computing paradigms and technologies which play an important role within the IoT scenario. Previous, or contemporary to the IoT, these paradigms/technologies have their own independence, but can effectively support the

Table 2: Summary of Section 4 findings

Paradigm\Technology integrated with ABC	**High-Level Integration Goals**	**Examples**
Edge Computing	negoziation, coordination, interoperability	schedule
Cloud Computing	negoziation, coordination, virtualization, data and computation offloading	load balancing, resource discovery
Wireless Sensor Network	resource management, code mobility,	energy and bandwidth saving
Machine Learning	decision support, prediction	time series analysis, real time resource configuration
Blockchain Technology	data security, coordination	
Semantic Technology		

IoT development, also jointly with the ABC. Again, due to the extent of the research filed, an exhaustive survey is not feasible. Therefore, we present some relevant mainstream paradigms and technologies, which perfectly work in conjunction with the ABC, to eventually support the IoT development. For each paradigm/technology, a brief introduction is provided, the contact points with IoT elicited and, finally, the benefits of synergies with the ABC discussed.

4.1. Cloud Computing

Cloud Computing empowers systems' resources and functionality by supporting extreme-scale intensive computation and massive, dynamic, heterogeneous data integration, storage and analytics. In the IoT context, Cloud Computing represents a fundamental enabler for the development of agent-based but resource constrained SOs. Indeed, Cloud Computing allows "agentified" SOs to locally provide complex functionality (enabled by SOs virtual aggregation and SOs services composition) while, behind the scenes, computations and data are offloaded on powerful remote servers. In such a way, SO hardware/software limitations are effectively and trasparently mitigated.

In a Cloud-assisted and Agent-based IoT (CA-IoT) platform [5], [143] wearable SOs, monitoring human activities, are first locally agentified and then virtualized into the Cloud, where incoming data is stored and analyzed, and new, empowered, virtual SOs created, as a meta-aggregation of existing ones. In particular, at the Cloud side, distributed data-flow processes analyze input data and dynamically support development of new application services to be provided by the agents running on the SOs.

Similarly, authors of [91] present an agent-based traffic management system: the Cloud supports processing of complex traffic strategy and massive data transport by providing needed storage and computing resources. In that way, highly demanding and interactive traffic simulations can be performed in a standard development environment (therefore, easily accessible by multiple agents/users like traffic managers, traffic participators, traffic-strategy developers, etc.) and on the basis of both real-time data and historical dataset. From another perspective, the

ABC facilitates constructing software tools and testbeds to automate the management of both cloud's resources and services. With respect to cloud's resources management and, particularly, to provide them with a dynamic load balancing mechanism, authors of [92] propose an Autonomous Agent Based Load Balancing Algorithm (A2LB), for maximizing utilization, responsiveness, scalability throughput and reliability. A set of agents is in charge of monitoring the load of every machine, controlling the transfer, selection and location policies, and proactively allocating the resources aiming to optimally balance the workload while keeping the service level agreement. With respect to cloud's services management, instead, in [90] authors designed and developed Cloudle, an agent-based search engine, supporting cloud service discovery, negotiation, and composition. Cloudle foresees the exploitation of agents for consulting cloud ontologies and automatically determining the similarity/compatibility among cloud services specifications and consumers requirements.

4.2. Edge Computing

Edge computing moves computation and storage resources into the proximity of where data has been generated, thus providing context-awareness, responsiveness, privacy, robustness and efficiency (in terms of both bandwidth and energy consumption) to IoT applications, like video analytics, personal healthcare, autonomous vehicles, etc. However, as a decentralized computing paradigm, Edge Computing demands SOs to be autonomous, interoperable and smart, as far as allowed by their (often constrained) software and hardware resources. And that's where ABC comes in, enabling SOs exhibiting autonomous, context-aware smart behaviors and fostering the collaborative execution of Edge Computing-based IoT applications.

Context-awareness and autonomy are design goals of FLEC (Flexible Edge Computing) architecture [144], which leverages a MAS, to autonomously and dynamically determine (i) allocation of processing load and tasks assignment to the Edge and to the Cloud, according to the properties of application, contexts, and resource situations, and *(ii)* provisioning of services, suitable for each user in real time, by reflecting detailed information like users behaviors, intentions, and preferences collected by IoT devices. Aiming at interoperability, instead, the ROAgent framework, described in [7] and [71], jointly exploits agent-oriented and resource-oriented paradigms, to enable interaction of resource-constrained, heterogeneous, SOs within the IoT ecosystem, through standard Web technologies. Indeed, SOs resources and operations are exposed as programming language- and platform-independent Web services, which can be browsed and searched over the Internet.

Likewise, [77] exploits the ABC for developing IoT Smart Enviro nments which exhibit reactive, proactive, and cognitive behaviors through computational resources located either in the Cloud or at the network Edge.

In [78], the same authors propose the CEIoT platform, a Cognitive-enabled and Edge-based IoT architecture, which leverages on the ABC for the realization of decentralized cognitive algorithms and the versatile development of smart data aggregations.

Finally, CRESCO [86] is an agent-based framework, provided with an application description language for supporting real-time streaming applications. CRESCO allows managing multitude of geographically distributed services based on heterogeneous resources, whose scheduling, provision and performances can be managed through proper high-order modules, while the ABC provides a fine-grained control over the structure, communication, and security protocols of the distributed system.

4.3. Wireless Sensor Networks

Wireless Sensor Network (WSN) is an enabling technology for the IoT, allowing the data collection, gathering, processing and forwarding through (typically resource constrained) wireless sensor nodes scattered across the "monitored area". Rapidly spreading in the first 2000s, WSNs found application in multitude of scenarios (ambient assisted living, structural health, e-health, etc.), which later have become of interest also for the IoT. In that sense, WSNs can be considered one of the forerunners of the IoT systems as well as one of its essential constituent.

Because of the scarce resources of both infrastructures (bandwidth is typically quite limited in wireless networks compared to wired ones) and WSNs' nodes (limited computation, storage and energy), agents' features of smartness, mobility and autonomy can be highly beneficial both to functional (by instilling smartness) and non-functional (energy and bandwidth saving) scopes.

With respect to the first point, lightweight agents can be deployed on sensor nodes, to efficiently handle their low-level functionality (sensing, filtering, pre-processing, storage, communication) and basic services (timer handling, resource access, etc.). This is the case of [89], in which an agent-oriented framework, for real-time human activity monitoring is proposed. In particular, here, lightweight agents operate according to an Event-Condition-Action (ECA) automaton and aim at computing the accelerometric sensory data, aggregating their features, and recognizing user movements and postures. Moreover, the agent-oriented design and programming allows effective and rapid prototyping of the sensor software, which shows also good performance in terms of recognition accuracy and nodes synchronization.

With respect to the second point, agents deployed within WSNs allow saving both network bandwidth and sensors energy. The distributed, intelligent decision making process, offered by lightweight agents is exploited in [146] to drive the opportunistic activation of the sensor nodes, with a subsequent energy saving. In [145], instead, an agent-based architecture implements a distributed cognitive radio resource management framework, which dynamically manages shared radio resources to minimize interference, and hence the energy waste. With the same goal, of minimizing the overall communication costs, sensory data can be processed locally rather than directly moved toward a sink. Following such principle, in [88] and [87] agents locally transform and reduce a large amount of sensory data by eliminating data redundancy among sensors through a context-aware on-node data processing (filtering and clustering, mainly).

4.4. Machine Learning

Machine learning (ML) is recently becoming one of the fastest growing areas of computer science, with extensive possibilities for wide range of applications. Essential concepts, algorithms, and theoretical frameworks in ML predominantly include supervised and unsupervised learning, statistical learning theory, probabilistic graphical models and approximate inference with great possibilities for developing new and more powerful approaches. In fact, ML can be seen as excellent alternative and upgrade for the development of algorithmic solutions, as compared to conventional engineering approaches. Usually, ML refers to automated detection of meaningful patterns in big datasets. Combination of wide variety of ML algorithms and their powerful processing of big datasets collected from IoT, SOs and Smart Environments can offer, and achieve, great benefits in variety of applications in different domains. Additionally, parallelization can significantly speed-up ML algorithms working with big datasets produced in IoT and Smart Environments. Moreover, time-series (TS) analysis, as a specific part of ML techniques, can be extremely useful in smart and IoT environments. TS analysis is used to describe changes of observed phenomena over time, such as the sensory information collected by SOs. Different systems and tools like R and FAP [94] can be used to enhance diverse aspects of smart environments and emergency scenarios [95], for example smart cities.

Agents, and their key abilities of learning, offer a significant shift in employing ML functionality in intelligent decision making processes. During the last two decades, developments in the agent technologies and ML have become complementary and agent-based platforms utilizing ML for intelligent decision support and automation have been constantly developing [93]. A set of agents, in smart environments, employing ML, may increase efficiency of learning and decision-support. Similarly, collective computational intelligence can support several ML algorithms, where a synergistic effect is expected from combining efforts of various kinds of agents. In essence, a set of agents, cooperating in distributed smart environments, can increase performance of such systems. For example CityMatrix [96] is an urban decision support system that facilitates evidence-based and more collaborative urban decision making. Role of applied ML techniques is to support real-time prediction of an agent-based urban traffic model. The system provides efficient optimization feedback, in real-time, to support the decision making process. Another interesting approach that incorporates agents, IoT and ML is presented in [97]. Authors propose ML to assists developing embodied and self-configurable agents for the IoT. In particular, a feedback-evaluative ML enables the reconfiguration of a MAS on the basis of the environmental context.

4.5. Blockchain Technology

Blockchain technology is an emerging and promising approach for securing the decentralization of data and control, as well as for executing Smart Contracts (SCs). Namely, for trusted and automatic activities, which encapsulate arbitrary and stateful functionality. Ranging from healthcare to logistics, from ambient assisted living to energy-trading, large-scale and privacy-sensitive distributed IoT systems, with their (often) resource-constrained SOs, can greatly benefit from Blockchain Technology when performing key operations, like object tracking, identity and policies management, transaction traceability and accountability, coordination, etc. By making such operations more secure, autonomous, flexible and even profitable, fostering also scalability, cross-organizational collaboration, interoperability, Blockchain Technology can fulfill some of the most important requirements and features shared by both multi-agent and IoT systems such as data integrity, privacy, authenticity, big data management and decentralised coordination [8].

The application of agent-based Blockchain Technology in the IoT arena is in its early stage and multifaceted. However, most of the current work deals with security-related topics. The CIoTA framework [139] uses Blockchain Technology to perform distributed and collaborative anomaly detection among resource-constrained IoT devices. Here, software agents runs on every IoT device, and build a local model (an extensible Markov model, precisely) for detecting malicious behaviors in a particular application. Authors of [141], [72] and [140] focus on SCs. In particular, since centralized access management technologies lack ability to efficiently deal with scalable load, [141] proposes fully distributed access control system for IoT, based on Blockchain Technology and ABC. Aiming at arbitrating roles and permissions, every blockchain node is associated with an agent, which is delegated to the deployment and management of a SC, during the lifetime of the access control system. Similarly, in [140] a MAS exploits SCs to manage the entire supply chain process more efficiently, by automatically writing transaction on the blockchain, verifying that both parties abide to the agreed conditions, and, if these are not met, imposing penalties. In a similar direction, in [72], authors propose to "agentify" the SCs, which currently is based on passive objects, in order to enhance their expressiveness with typical agents' features of autonomy, situatedness, sociality, and intelligence. Going beyond accountability and identity management tasks, there exist works that demonstrate the conceptual and technical feasibility of blockchain-based trustworthy coordination in MAS (see, for instance, [73], [142]), i.e., they propose how to ground trustworthy, decentralised coordination in MAS upon blockchain. Nevertheless, these are just preliminary results.

4.6. Semantic Technologies

Relationship between agents and semantic technologies (ST) is a complex one. It can be traced back to 2001, when [136], and [135] have been published. The main idea was (and still is) that ST can make agents in more flexible, with easy update of knowledge about the world, and allow agents to understand "the world" deeper and broader. However, already then things were not perfect. For instance, FIPA Ontology Service Specification[7], which introduced an Ontology Agent (an ontology manager), never became standardized. Furthermore, no popular agent platform implemented such agent (only found reference is [134]). Moreover, developers of one of very popular agent platforms Jade[8] attempted at providing semantic services[9], using ontologies in agent communication[10], or combining Jade ontology services with Protege[11]. However, instead of representing knowledge

[7] http://www.fipa.org/specs/fipa00086
[8] https://jade.tilab.com
[9] https://jade.tilab.com/doc/tutorials/clontosupport.pdf
[10] https://jade.tilab.com/papers/papercaireaart.pdf
[11] https://jade.tilab.com/doc/tutorials/BeanOntologyTutorial.pdf

and forming a flexible knowledge base, ontologies became Java-code fragments.

Overall, most attempts at directly fusing software agents and semantic technologies failed to gain traction. Software agents evolved in three main "non-semantic" directions: (1) general purpose platforms (e.g., Jade), (2) BDI-based platforms (e.g., Jadex or Jason) that, apply Prolog/expert system approach, and (3) simulation platforms (e.g., NetLogo or Repast). At the same time, semantic technologies evolved at their own pace, separately from agent systems.

Nevertheless, one can find a number of projects that attempted at using jointly agent systems and semantics. Let us list a few examples, which illustrate general trends: *(i)* Magenta Technologies: Multi-agent Systems for Ocean logistics – an agent-semantic system solving scheduling problem [130]; (ii) Semantic-based Travel Support System: where semantic representation of "world of travel" is used [129, 128]; (iii) Semantic based Workers support system: based on ontologies of organization, travel, research interests, used to support workers in a virtual organization [127, 126, 125]; and (iv)Agent-semantic system for management of resources in computational grids (or clouds) [123, 121]. Here, an ontology of numerical linear algebra has been developed [120], and possibility of combining agents+semantics+multicriterial analysis has been studied [119].

In all cases, ontologies were stored and processed in a separate repository. In this way, knowledge representation could have been created and updated without (major) changes in the core application/system (see, also, [118]).

The question thus arises, what role can semantic technologies play in the IoT, and how they can be connected with agents. Here, two main scenarios emerge. First, in an IoT ecosystem, a common vocabulary is instantiated. However, we are convinced that (at least for some time to come) only "small", domain specific ontologies will materialize in applications. here, agents will share common modular ontology ([117]) and exchange "semantic messages" (ACL messages, with semantic payload) to communicate (see, [116]). Similarly, ontologies can be used to facilitate access control [115, 114].

Second, when different data models coexist within a single IoT ecosystem (e.g. due to the ecosystem merger), when interoperability has to be achieved, semantic technologies can be used to translate messages exchanged between "things" [113, 112]. This approach can be applied in the ABC. Here, a "translator agent", communicating with individual agents representing each data model, would facilitate the message translation service (see, [111, 110, 109]).

In summary, while the past collaboration of agents and semantics was not particularly successful, it seems that both technologies have reached the state when they are ready to join forces to deliver important results in IoT ecosystems.

5. Successful exploitation of Agents in IoT applications

Based on paradigms and technologies reported in the previous (sub)sections, the IoT has given rise to several applications in many different fields. In contrast with the mostly theoretical and foundational contributions reported in Section 4, hereinafter some successful agent-based IoT applications, and related platforms, are reported.

The use of SOs, and real time analytics, to enhance manufacturing and industrial processes, led to the *Industrial IoT (IIoT)* [12]. PTC ThingWorx[12] is a cloud-based end-to-end technology platform (free use is possible, with limited functionality) that enables innovators to rapidly develop and deploy agent-based solutions for the IIoT. The ABC supports the device modeling (different agents are used for the different types of devices), the business logic definition, the design and implementation of collaboration and security mechanisms required for IoT applications. Recently, PTC ThingWorx included an edge microserver (server software for IoT edge devices) and a software development kit for minimizing devices' power and data demands.

Also in the IIoT arena, Arrayent[13] is a cloud- and agent-based IoT platform specifically designed for manufacturers of mass-market consumer home products. Arrayent empowers brands to get connected and get closer to their customers, by enabling them to connect, monitor and control their IoT products remotely from anywhere. This is possible thanks to lightweight agents, which are deployed on the devices, and have their virtual twins in the Cloud, accessible through RESTful APIs. The ABC helps to manage the devices, their actions and reactions, and their data, as well as in interconnecting different embedded computing platforms (Atmel, Broadcom, Texas Instruments, Raspberry Pi, etc.) and communication protocol (including Wi-Fi, Thread, ZigBee, Z-Wave).

Cumulocity IoT[14] is an IoT platform to connect and to experience heterogeneous "things" instantly. It provides pre-integrated devices, or open device agents, and APIs, to turn insight into action using analytics, application integration and workflow management and to, finally, deliver branded services using secure multi-tenancy and role-based access control. Software agents are used as interoperability facilitators, since they translate device-specific interface protocols into reference protocols (i.e., REST and JSON) and specific domain models into one ref-

[12]http://www.thingworx.com/

[13]https://www.arrayent.com/

[14]https://cumulocity.com/guides/concepts/introduction/

erence domain model, and enable secure remote communication across virtually any network.

Similarly, CloudPlugs [15], a container-based, edge to cloud IoT platform for digital transformation and implementation of IIoT initiatives, exploits a software agents to eliminate slow and costly firmware development, to quickly develop device applications and to deploy them, with one click, to thousands of devices. The CloudPlugs agents provide multi-protocol support, secure data ingestion, telemetry, data processing, protocol mediation and local and remote edge control.

The synergy of the IoT and social networking paradigms enables the development of communities where SOs outline humans-like social interactions and temporary collaborations in contexts where expertise and capabilities are spatially distributed. The *Social Internet of Things (SIoT)* aims at simplifying the navigability of a dynamic network of billions of SOs, as well as enhancing their efficiency and trustworthiness when providing information and services [138], [35].

Speaking Object [98] is a framework pivoting on the argumentation-based coordination: agents can autonomously and dynamically select the better interaction protocol for the current situation, without recurring to prescribed coordination rules. Fostering interoperability in SOs interactions is the role of agents in iSapiens [99], a framework for designing and implementing smart environments with automatic SOs inclusion leveraging on information such as location, ownership, chronology of mutual interactions.

In the Social Factory platform [100] twin agents of humans and SOs are interfaced through broker agents facilitating their contextualized interactions within a cyberphysical environment across an enterprise social network. Here, the agent-based interfaces allows preventing errors and minimizing out-of-the-loop performance of the human-ware by preserving an adequate level of situation awareness and mental workload.

Agents can foster integration of SOs into social networks, for example by updating user's profile, his/her friendships, and by simplifying social network's services discovery and composition [102]. Trustworthiness in SIoT is the focus of [158] and [101]. The former work presents a Trust Framework, which promotes social interactions among SOs by associating each of them to a software agent and enabling the decentralized dissemination of their reputation through the Blockchain Technology. The latter, instead, presents a Trust Service Platform, where agents constitute a semi-centralized trust management and reputation sys-

[15] https://cloudplugs.com/iot-platform-overview/

tem but without exploiting the Blockchain Technology.

The *Internet of Vehicles (IoV)* is a large-scale, distributed system for wireless communication and information exchange between smart vehicles and other SOs (smartphones, smart traffic lights and signals, etc.). As result, complex real-time tasks, such as dynamic and intelligent routing/congestion/traffic management, are performed according to current traffic and environmental conditions [11]. An agent-based approach can be successfully adopted in the IoV scenario because vehicular networks are typically large scale and geographically distributed in dynamic environments, thus requiring autonomous, collaborative and reactive SOs. As reported in [103], several IoV platforms exploit the ABC as modeling and simulation paradigm for complex traffic and transportation systems. For example, [104] proposes JADE-based agents to support traffic operators in determining the best traffic strategies when non-urban roadway meteorological incidents occur.

Moving to urban scenarios, TRACK-R [105] is an agent-based platform which exploits a MAS for providing traffic route recommendation to both agentified SOs and humans. Chen et al. [106] developed Mobile-C, a real-time agent-based traffic detection and management system, in which both stationary and mobile agents collaborate in distributed computing and information fusion. Similarly, aiming at an adaptive and smart traffic and transportation control, a highly distributed agent-based traffic management system is reported in [107], in which a sophisticated control algorithm is decomposed on demand, into simple task-oriented agents to adapt to various control scenarios. An agent-based framework for traffic control without traffic signal systems is presented in [137]. Here, each of connected vehicles is modeled as an agent and they all communicate and collaborate through wireless communication technologies for tackling the scalable and flexible problem of intersections. Finally, [108] presents a testbed to experiment and rapidly prototype multi-agent control systems in road traffic management with different strategies. Interestingly, all the aforementioned works conform to the FIPA standards for the sake of interoperability.

6. Concluding Analysis and Remarks

There are no technology limitations (in terms of computing, storage and communication) hindering the full realization of the IoT ecosystem. However, its multifaceted development issues still need to be comprehensively, and simultaneously, tackled. For example, according to Atzori et al. [1], scalability and self-management used to be separately tackled, in spite of centralized and predefined centralized approaches. This prevented SOs to constitute locally operating, self-organizing and self-adaptive systems.

We believe, and the large number of works analyzed in this survey confirms it, that agents' key features of autonomy, proactivity, intelligence and sociability make the ABC a natural candidate for systematically and effectively developing IoT ecosystems. Indeed, better than other computing paradigms (object-, service-, and component-oriented), the ABC allows modeling both SO/IoT system at different degrees of details and programming (technical, syntactical and semantic) interoperability, autonomicity and distributed intelligence. In addition, the agent-based simulation allows validating multiple design choices before the deployment phase, while agent-based methodologies can systematically and effectively drive both the complete development and the re-engineering process of IoT ecosystem, also in synergy with other paradigms (e.g., cloud and edge computing). However, to ensure that the benefits of an agent-based solution far outweigh its overhead, three pragmatic aspects should be carefully assessed before blindly adopting the ABC paradigm (this holds not only for the IoT, but in any context). Such aspects respectively deal with technology, economic and conceptual issues.

The *first aspect* refers to relative immaturity of agent technology, which born and raised not in industry but mainly in academia: as consequence, few agent-based commercial platforms [23] are currently available, and they made no significantly progresses in the last decades [25], particularly regarding standardization, semantic interoperability and real-timeness. The latter is a crucial aspect for many safety-critical IoT applications in which the consequences of, even rare, time failures are potentially disastrous. As underlined by [75], MASs typically adopt best-effort approaches with internal agent schedulers, negotiation protocols, and communication middleware, which do not include comprehensive and global mechanisms for handling real-timeness. This lack affects reliability and predictability. Moreover, FIPA standards only partially support interoperability in real-time/distributed control and diagnostics [63]. Therefore, extensive interventions in terms of theoretical contributions and practical mechanisms involving the MAS core elements need to be simultaneously and coherently carried out aiming to real-time compliant agent-based SOs, IoT systems and applications.

The *second aspect* refers to greater investments required to implement agent-based IoT solutions, typically costlier and less user-friendly than conventional centralized and service-oriented ones [63] (e.g., data flow programming for Web of Things [70]). Indeed, as underlined in [76], agent-based methodologies, tools and languages are designed primarily to serve expert researchers, otherwise requiring consistent learning efforts. For facilitating both non-technical end-users and non-expert researchers in the development of agent-based SOs and IoT systems, frameworks should be simple to understand and use, for example by providing ready-to-use and easily customizable building blocks and Computer Aided Software Engineering (CASE) tools to semi-automatically drive the application of the methodology.

The *third aspect* refers to, only apparently backdated, misapplications and misconceptions:

- *things can be always profitably agentified*: a large number of dumb devices within the IoT ecosystem (RFID and NFC tags, resource-constrained microcontrollers, etc.) work just with a single thread of control and expose simple conditional behaviours [23, 25]. Without complex tasks to be performed (e.g., automatic resolution of policy conflicts, access synchronization to shared resources, dynamic organization without any a-priori network model [79]), such devices should not be designed as agents, which conversely are intrinsically autonomous multi-thread problem solvers;

- *applications can always profitably exploit agents*: it is false that agents are a universal development solution. For example, agent-based techniques fit just the 30% of control tasks and 60% of diagnostic tasks in the industrial scenario [63, 24].

Figure 3 reports and summarizes the main considerations of this analysis under the form of a SWOT (Strengths, Weaknesses, Opportunities, Threats) matrix.

In conclusion, although adopting the ABC paradigm demands for a careful preliminary evaluation of pros and cons, the several described contributions across this survey proved that, to date, an agent-based development approach represents a suitable and effective choice to face the majority of advanced (current and future) SOs and IoT systems.

Acknowledgments

This work was supported in part by PAS-RAS bilateral project "Semantic foundation of the Internet of Things", as well as a collaboration agreement between University of Novi Sad, University of Craiova, SRIPAS and Warsaw University of Technology.

Bibliography

[1] Atzori, L., Iera, A., Morabito, G., 2010. The internet of things: A survey. Computer networks 54, 2787-2805.

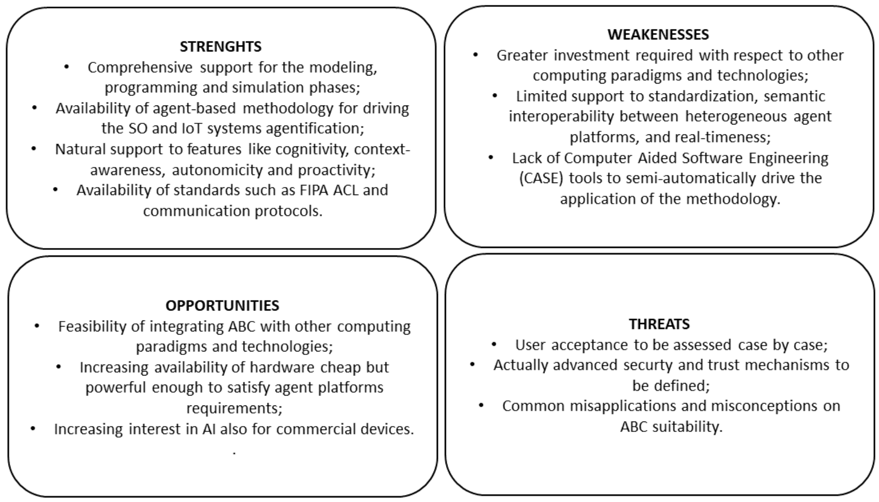

Figure 3: SWOT matrix about ABC for IoT development

[2] Mattern, F., Floerkemeier, C., 2010. From the Internet of Computers to the Internet of Things, in: From Active Data Management to Event-Based Systems and More. Springer, pp. 242-259.

[3] Patel, P., Cassou, D., 2015. Enabling high-level application development for the internet of things. Journal of Systems and Software 103, 62-84.

[4] Luck, M., McBurney, P., Preist, C., 2004. A manifesto for agent technology: Towards next generation computing. Autonomous Agents and Multi-Agent Systems, 203-252.

[5] Fortino, G., Guerrieri, A., Russo, W., Savaglio, C., 2014. Integration of agent-based and Cloud Computing for the smart objects-oriented IoT, in: Computer Supported Cooperative Work in Design (CSCWD), Proc. of the 2014 IEEE 18th Intl. Conf. on. IEEE, pp. 493-498.

[6] P. Pico-Valencia and J. A. Holgado-Terriza, "Agentification of the internet of things: A systematic literature review," *Intl. Journal of Distributed Sensor Networks*, vol. 14, no. 10, p. 1550147718805945, 2018.

[7] T. Leppänen, C. Savaglio, L. Lovén, W. Russo, G. Di Fatta, J. Riekki, and G. Fortino, "Developing agent-based smart objects for iot edge computing:

Mobile crowdsensing use case," in *Intl. Conf. on Internet and Distributed Computing Systems*. Springer, 2018, pp. 235–247.

[8] D. Calvaresi, A. Dubovitskaya, J. P. Calbimonte, K. Taveter, and M. Schumacher, "Multi-agent systems and blockchain: Results from a systematic literature review," in *Intl. Conf. on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2018, pp. 110–126.

[9] C. Savaglio, G. Fortino, M. Ganzha, M. Paprzycki, C. Bădică, and M. Ivanović, "Agent-based computing in the internet of things: a survey," in *Intl. Symposium on Intelligent and Distributed Computing*. Springer, 2017, pp. 307–320.

[10] A. P. Plageras, K. E. Psannis, C. Stergiou, H. Wang, and B. B. Gupta, "Efficient iot-based sensor big data collection–processing and analysis in smart buildings," *Future Generation Computer Systems*, vol. 82, pp. 349–357, 2018.

[11] M. Chen, Y. Tian, G. Fortino, J. Zhang, and I. Humar, "Cognitive internet of vehicles," *Computer Communications*, vol. 120, pp. 58–70, 2018.

[12] S. Wang, J. Wan, D. Zhang, D. Li, and C. Zhang, "Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination," *Computer Networks*, vol. 101, pp. 158–168, 2016.

[13] A. Tate, Y.-H. Chen-Burger, J. Dalton, S. Potter, D. Richardson, J. Stader, G. Wickler, I. Bankier, C. Walton, and P. Williams, "I-room: a virtual space for intelligent interaction," *IEEE Intelligent Systems*, vol. 25, no. 4, pp. 62–71, 2010.

[14] Savaglio, C., Fortino, G., Zhou, M., 2016. Towards interoperable, cognitive and autonomic IoT systems: An agent-based approach, in: Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on. IEEE, pp. 58-63.

[15] Jennings, N.R., 1999. Agent-based computing: Promise and perils.

[16] Fortino, G., Rovella, A., Russo, W., Savaglio, C., 2016. Towards Cyberphysical Digital Libraries: Integrating IoT Smart Objects into Digital Libraries, in: Management of Cyber Physical Objects in the Future Internet of Things. Springer, pp. 135-156.

[17] Fortino, G., Guerrieri, A., Russo, W., Savaglio, C., 2015. Towards a development methodology for smart object-oriented IoT systems: A metamodel approach, in: Systems, Man, and Cybernetics, 2015 IEEE Intl. Conf. on. IEEE, pp. 1297-1302.

[18] Ricci, A., Santi, A., 2011. Agent-oriented computing: Agents as a paradigm for computer programming and software development, in: Proc. of the 3rd Intl Conf. on Future Computational Tech. and Applications. Wilmington: Xpert Publishing Services. Citeseer, pp. 42-51.

[19] Poslad, S., 2007. Specifying protocols for multi-agent systems interaction. ACM Transactions on Autonomous and Adaptive Systems (TAAS) 2, 15.

[20] Fipa, A., 2002. Fipa acl message structure specification. Foundation for Intelligent Physical Agents, http://www. fipa. org/specs/fipa00061/SC00061G. html (30.6. 2004).

[21] Macal, C.M., North, M.J., 2005. Tutorial on agent-based modeling and simulation, in: Simulation Conf., 2005 Proc. of the Winter. IEEE, p. 14-pp.

[22] Bergenti, F., Gleizes, M.-P., Zambonelli, F., 2006. Methodologies and software engineering for agent systems: the agent-oriented software engineering handbook. Springer Science & Business Media.

[23] Wooldridge, M.J., Jennings, N.R., 1999. Software engineering with agents: Pitfalls and pratfalls. IEEE Internet Computing 3, 2027.

[24] Gawinecki, M., Kruszyk, M., Paprzycki, M., Ganzha, M., 2007. Pitfalls of agent system development on the basis of a Travel Support System, in: W. Abramowicz (ed), Proc. of the BIS 2007 Conf., Springer, Berlin, LNCS 4439, 488-499

[25] Nwana, H.S., Ndumu, D.T., 1999. A perspective on software agents research. The Knowledge Engineering Review 14, 125142.

[26] Manzalini, A., Zambonelli, F., 2006. Towards autonomic and situation-aware communication services: the cascadas vision, in: Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006. DIS 2006. IEEE Workshop on. IEEE, pp. 383-388.

[27] Spanoudakis, N., Moraitis, P., 2015. Engineering ambient intelligence systems using agent technology. IEEE Intelligent Systems 30, 60-67.

[28] Katasonov, A., Kaykova, O., Khriyenko, O., Nikitin, S., Terziyan, V.Y., 2008. Smart Semantic Middleware for the Internet of Things. ICINCO-ICSO 8, 169-178.

[29] Ruta, M., Scioscia, F., Loseto, G., Di Sciascio, E., 2014. Semantic-based resource discovery and orchestration in home and building automation: A multi-agent approach. IEEE Trans. on Industrial Informatics 10, 730-741.

[30] Vlacheas, P., Giaffreda, R., Stavroulaki, V., Kelaidonis, D., Foteinos, V., Poulios, G., Demestichas, P., Somov, A., Biswas, A.R., Moessner, K., 2013. Enabling smart cities through a cognitive management framework for the internet of things. IEEE Communications Magazine 51, 102-111.

[31] N. M. do Nascimento and C. J. P. de Lucena, "Fiot: An agent-based framework for self-adaptive and self-organizing applications based on the internet of things," *Information Sciences*, vol. 378, pp. 161–176, 2017.

[32] K. Batool and M. A. Niazi, "Modeling the internet of things: a hybrid modeling approach using complex networks and agent-based models," *Complex Adaptive Systems Modeling*, vol. 5, no. 1, p. 4, 2017.

[33] M. Dzaferagic, M. M. Butt, M. Murphy, N. Kaminski, and N. Marchetti, "Agent-based modelling approach for distributed decision support in an iot network," *arXiv preprint arXiv:1901.04585*, 2019.

[34] Fortino, G., Guerrieri, A., Russo, W., 2012. Agent-oriented smart objects development, in: Computer Supported Cooperative Work in Design (CSCWD), 2012 IEEE 16th Intl. Conf. on. IEEE, pp. 907-912.

[35] F. Cicirelli, A. Guerrieri, G. Spezzano, and A. Vinci, "An edge-based platform for dynamic smart city applications," *Future Generation Computer Systems*, vol. 76, pp. 106–118, 2017.

[36] Kato, T., Chiba, R., Takahashi, H., Kinoshita, T., 2015. Agent-Oriented Cooperation of IoT Devices Towards Advanced Logistics, in: Computer Software and Applications Conf., 2015 IEEE 39th Annual. IEEE, pp. 223-227.

[37] Zhang, X., Adhikari, R., Pipattanasomporn, M., Kuzlu, M., Bradley, S.R., 2016. Deploying IoT devices to make buildings smart: Performance evaluation and deployment experience, in: Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on. IEEE, pp. 530-535.

[38] Terziyan, V., Kaykova, O., Zhovtobryukh, D., 2010. Ubiroad: Semantic middleware for context-aware smart road environments, in: Internet and Web Applications and Services (Iciw), 2010 Fifth Intl. Conf. on. IEEE, pp. 295-302.

[39] Mzahm, A.M., Ahmad, M.S., Tang, A.Y., 2013. Agents of Things (AoT): An intelligent operational concept of the Internet of Things (IoT), in: Intelligent Systems Design and Applications (ISDA), 2013 13th Intl. Conf. on. IEEE, pp. 159-164.

[40] Ayala, I., Amor, M., Fuentes, L., 2015. The Sol agent platform: Enabling group communication and interoperability of self-configuring agents in the Internet of Things. Journal of Ambient Intelligence and Smart Environments 7, 243-269.

[41] Leppänen, T., Riekki, J., Liu, M., Harjula, E., Ojala, T., 2014. Mobile agents-based smart objects for the IoT, in: Internet of Things Based on Smart Objects. Springer, pp. 29-48.

[42] Pujolle, G., 2006. An autonomic-oriented architecture for the internet of things, in: Modern Computing, 2006. IEEE John Vincent Atanasoff 2006 Int.l Symp. on. IEEE, pp. 163-168.

[43] Manate, B., Munteanu, V.I., Fortis, T.-F., 2013. Towards a scalable multi-agent architecture for managing iot data, in: P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth Intl. Conf. on. IEEE, pp. 270-275.

[44] Wu, Q., Ding, G., Xu, Y., Feng, S., Du, Z., Wang, J., Long, K., 2014. Cognitive internet of things: a new paradigm beyond connection. IEEE Internet of Things Journal 1, 129-143.

[45] VICINITY - Open virtual neighbourhood network to connect IoT infrastructures and smart objects, http://vicinity2020.eu/vicinity/

[46] Kiljander, J., Delia, A., Morandi, F., Hyttinen, P., Takalo-Mattila, J., Ylisaukko-Oja, A., Soininen, J.-P., Cinotti, T.S., 2014. Semantic interoperability architecture for pervasive computing and internet of things. IEEE access 2, 856-873.

[47] Cheng, B., Longo, S., Cirillo, F., Bauer, M., Kovacs, E., 2015. Building a big data platform for smart cities: Experience and lessons from santander, in: Big Data (BigData Congress), 2015 IEEE Intl. Congr. on. IEEE, pp. 592-599.

[48] Ganzha, M., Paprzycki, M., Pawlowski, W., Szmeja, P., Wasielewska, K., 2017. Semantic interoperability in the Internet of Things: An overview from the INTER-IoT perspective. Journal of Network and Computer Applications 81, 111-124.

[49] D. Namiot and M. Sneps-Sneppe, "On micro-services architecture," *International Journal of Open Information Technologies*, vol. 2, no. 9, pp. 24–27, 2014.

[50] P. Krivic, P. Skocir, M. Kusek, and G. Jezic, "Microservices as agents in iot systems," in *KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*. Springer, 2017, pp. 22–31.

[51] R. W Collier, E. O'Neill, D. Lillis, and G. O'Hare, "Mams: Multi-agent microservices¢Å," in *Companion Proceedings of The 2019 World Wide Web Conference*. ACM, 2019, pp. 655–662.

[52] K. Kravari and N. Bassiliades, "Storm: A social agent-based trust model for the internet of things adopting microservice architecture," *Simulation Modelling Practice and Theory*, vol. 94, pp. 286–302, 2019.

[53] Savaglio, C., Fortino, G., 2015. Autonomic and Cognitive Architectures for the Internet of Things, in: Intl. Conf. on Internet and Distributed Computing Systems. Springer, pp. 39-47.

[54] Mitrović, D., Ivanović, M., Budimac, Z., Vidaković, M., 2014. Radigost: Interoperable web-based multi-agent platform. Journal of Systems and Software 90, 167178.

[55] Xu, X., Bessis, N., Cao, J., 2013. An Autonomic Agent Trust Model for IoT systems. Procedia Computer Science 21, 107113.

[56] Karnouskos, S., De Holanda, T.N., 2009. Simulation of a smart grid city with software agents, in: Computer Modeling and Simulation, 2009. EMS09. Third UKSim European Symposium on. IEEE, pp. 424-429.

[57] D'Angelo, G., Ferretti, S., Ghini, V., 2017. Multi-level simulation of Internet of Things on smart territories. Simulation Modelling Practice and Theory 73, 3-21.

[58] Fortino, G., Russo, W., Savaglio, C., 2016. Simulation of Agent-oriented Internet of Things Systems, in: Proc. 17th Workshop" From Objects to Agents. pp. 8-13.

[59] Fortino, G., Russo, W., Savaglio, C., 2016. Agent-oriented modeling and simulation of IoT networks, in: Computer Science and Information Systems (FedCSIS), 2016 Federated Conf. on. IEEE, pp. 1449-1452.

[60] Zambonelli, F., 2016. Towards a General Software Engineering Methodology for the Internet of Things. arXiv preprint arXiv:1601.05569.

[61] Kasnesis, P., Toumanidis, L., Kogias, D., Patrikakis, C.Z., Venieris, I.S., 2016. ASSIST: An agent-based SIoT simulator, in: Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on. IEEE, pp. 353358.

[62] Manate, B., Fortis, F., Moore, P., 2014. Applying the Prometheus methodology for an Internet of Things architecture, in: Proc. of the 2014 IEEE/ACM 7th Intl. Conf. on Utility and Cloud Computing. IEEE Computer Society, pp. 435-442.

[63] Marik, V., McFarlane, D., 2005. Industrial adoption of agent-based technologies. IEEE Intelligent Systems 20, 2735.

[64] Razzaque, M.A., Milojevic-Jevric, M., Palade, A., Clarke, S., 2016. Middleware for internet of things: a survey. IEEE Internet of Things Journal 3, 70-95.

[65] Fortino, G., Guerrieri, A., Russo, W., Savaglio, C., 2014. Middlewares for smart objects and smart environments: overview and comparison, in: Internet of Things Based on Smart Objects. Springer, pp. 1-27.

[66] Kubler, O., S.; Framling, K.; Zaslavsky, A.; Doukas, C.; Olivares, E.; Fortino, G.; Palau, C. E.; Soursos, S.; Podnar Åarko, I.; Fang, Y.; Kro,

S.; Heinz, C.; Grimm, C.; Broering, A.; Miti, J.; Olstedt, K.; Vermesan, O., 2016. Digitising the Industry: Internet of Things Connecting the Physical, Digital and Virtual Worlds. River Publishers, chapter 9, pp. 431-448, Vol. 49.

[67] Morris, A., Giorgini, P., and Abdel-Naby, S., 2009. Simulating BDI-Based Wireless Sensor Networks. 2009 IEEE/WIC/ACM Intl. Joint Conf. on Web Intelligence and Intelligent Agent Technology, volume 2, IEEE, pp. 78-81.

[68] Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J., 2004. Tropos: An agent-oriented software development methodology. Autonomous Agents and Multi-Agent Systems 8, 203236.

[69] C. Savaglio, W. Russo, G. Fortino, T. Leppänen, and J. Riekki, "Re-engineering iot systems through acoso-meth: the ietf core based agent framework case study," in *19th Workshop from Objects to Agents (WOA 2018), Italy*, 2018, pp. 28–29.

[70] Guinard, D., Trifa, V., Pham, T., Liechti, O., 2009. Towards physical mashups in the web of things, in: Networked Sensing Systems (INSS), 2009 Sixth Intl. Conf. on. IEEE, pp. 14.

[71] T. Leppänen, "Resource-oriented mobile agent and software framework for the internet of things," *University of Oulu, Faculty of Information Technology and Electrical Engineering*, 2018.

[72] G. Ciatto, A. Maffi, S. Mariani, and A. Omicini, "Towards agent-oriented blockchains: Autonomous smart contracts," in *PAAMS 2019*, 2019.

[73] G. Ciatto, S. Mariani, and A. Omicini, "Blockchain for trustworthy coordination: A first study with Linda and Ethereum," in *2018 IEEE/WIC/ACM Intl. Conf. on Web Intelligence (WI)*, Dec. 2018, pp. 696–703.

[74] P. Pico-Valencia, and J.A. Holgado-Terriza, "Semantic agent contracts for internet of agents," in *2016 IEEE/WIC/ACM Intl. Conf. on Web Intelligence Workshops (WIW)*. IEEE, 2016, pp. 76–79.

[75] D. Calvaresi, M. Marinoni, A. Sturm, M. Schumacher, and G. Buttazzo, "The challenge of real-time multi-agent systems for enabling iot and cps," in *Proc. of the Intl. Conf. on web intelligence*. ACM, 2017, pp. 356–364.

[76] H. Yu, Z. Shen, and C. Leung, "From internet of things to internet of agents," in *2013 IEEE Intl. Conf. on green computing and communications and IEEE Internet of Things and IEEE cyber, physical and social computing*. IEEE, 2013, pp. 1054–1057.

[77] F. Cicirelli, A. Guerrieri, A. Mercuri, G. Spezzano, and A. Vinci, "Itema: A methodological approach for cognitive edge computing iot ecosystems," *Future Generation Computer Systems*, vol. 92, pp. 189 – 197, 2019.

[78] F. Cicirelli, A. Guerrieri, G. Spezzano, and A. Vinci, "A Cognitive Enabled, Edge-Computing Architecture for Future Generation IoT Environments," in *In the Proceeding of the IEEE 5th World Forum on Internet of Things*, Limerick, Ireland, 2019.

[79] C. Savaglio, P. Pace, G. Aloi, A. Liotta, and G. Fortino, "Lightweight reinforcement learning for energy efficient communications in wireless sensor networks," *IEEE Access*, vol. 7, pp. 29 355–29 364, 2019.

[80] N. Cila, I. Smit, E. Giaccardi, and B. Kröse, "Products as agents: metaphors for designing the products of the iot age," in *Proc. of the 2017 CHI Conf. on Human Factors in Computing Systems*. ACM, 2017, pp. 448–459.

[81] S. Berrani, A. Yachir, B. Djemaa, and M. Aissani, "Extended multi-agent system based service composition in the internet of things," in *2018 3rd Intl. Conf. on Pattern Analysis and Intelligent Systems (PAIS)*. IEEE, 2018, pp. 1–8.

[82] M. Pérez Hernández, B. Alturki, and S. Reiff-Marganiec, "Fabiot: A flexible agent-based simulation model for iot environments," 2018.

[83] H. O. Al-Sakran, "Intelligent traffic information system based on integration of internet of things and agent technology," *Intl. Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 6, no. 2, pp. 37–43, 2015.

[84] G. D'Angelo, S. Ferretti, and V. Ghini, "Simulation of the internet of things," in *2016 Intl. Conf. on High Performance Computing & Simulation (HPCS)*. IEEE, 2016, pp. 1–8.

[85] S. Karnouskos and T. N. De Holanda, "Simulation of a smart grid city with software agents," in *2009 Third UKSim European Symposium on Computer Modeling and Simulation*. IEEE, 2009, pp. 424–429.

[86] V. C. Bumgardner, V. W. Marek, and C. D. Hickey, "Cresco: A distributed agent-based edge computing framework," in *2016 12th Intl. Conf. on Network and Service Management (CNSM)*. IEEE, 2016, pp. 400–405.

[87] M. Chen, T. Kwon, Y. Yuan, and V. C. Leung, "Mobile agent based wireless sensor networks," *Journal of computers*, vol. 1, no. 1, pp. 14–21, 2006.

[88] H. Qi, Y. Xu, and X. Wang, "Mobile-agent-based collaborative signal and information processing in sensor networks," *Proc. of the IEEE*, vol. 91, no. 8, pp. 1172–1183, 2003.

[89] F. Aiello, F. L. Bellifemine, G. Fortino, S. Galzarano, and R. Gravina, "An agent-based signal processing in-node environment for real-time human activity monitoring based on wireless body sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 7, pp. 1147–1161, 2011.

[90] K. M. Sim, "Agent-based cloud computing," *IEEE Transactions on services computing*, vol. 5, no. 4, pp. 564–577, 2012.

[91] Z. Li, C. Chen, and K. Wang, "Cloud computing for agent-based urban transportation systems," *IEEE Intelligent Systems*, vol. 26, no. 1, pp. 73–79, 2011.

[92] A. Singh, D. Juneja, and M. Malhotra, "Autonomous agent based load balancing algorithm in cloud computing," *Procedia Computer Science*, vol. 45, pp. 832–841, 2015.

[93] B. Lorica, "How to think about AI and machine learning technologies, and their roles in automation: an overview and framework, including tools that can be used to enable automation," *https://www.oreilly.com/ideas/how-to-think-about-ai-and-machine-learning-technologies-and-their-roles-in-automation,* accessed 20.04.2019, June 11, 2018.

[94] V. Kurbalija, M. Ivanović, Z. Geler, and M. Radovanović, "Two faces of the framework for analysis and prediction, part 2-research," *Information Technology And Control*, vol. 47, no. 3, pp. 489–502, 2018.

[95] M. Lujak and S. Ossowski, "Evacuation route optimization architecture considering human factor," *AI Communications*, vol. 30, no. 1, pp. 53–66, 2017.

[96] Y. Zhang, A. Grignard, K. Lyons, A. Aubuchon, and K. Larson, "Real-time machine learning prediction of an agent-based model for urban decision-making," in *Proc. of the 17th Intl. Conf. on Autonomous Agents and Multi-Agent Systems*. Intl. Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 2171–2173.

[97] N. Nascimento, P. Alencar, C. Lucena, and D. Cowan, "An iot analytics embodied agent model based on context-aware machine learning," in *2018 IEEE Intl. Conf. on Big Data (Big Data)*. IEEE, 2018, pp. 5170–5175.

[98] M. Lippi, M. Mamei, S. Mariani, and F. Zambonelli, "An argumentation-based perspective over the social iot," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2537–2547, 2018.

[99] F. Cicirelli, A. Guerrieri, G. Spezzano, A. Vinci, O. Briante, and G. Ruggeri, "isapiens: a platform for social and pervasive smart environments," in *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*. IEEE, 2016, pp. 365–370.

[100] D. Romero, T. Wuest, J. Stahre, and D. Gorecky, "Social factory architecture: social networking services and production scenarios through the social internet of things, services and people for the social operator 4.0," in *IFIP Intl. Conf. on Advances in Production Management Systems*. Springer, 2017, pp. 265–273.

[101] N. B. Truong, T.-W. Um, and G. M. Lee, "A reputation and knowledge based trust service platform for trustworthy social internet of things," *Innovations in clouds, Internet and networks (ICIN)*, 2016.

[102] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization," *Computer networks*, vol. 56, no. 16, pp. 3594–3608, 2012.

[103] B. Chen and H. H. Cheng, "A review of the applications of agent technology in traffic and transportation systems," *IEEE Transactions on intelligent transportation systems*, vol. 11, no. 2, pp. 485–497, 2010.

[104] V. R. Tomás and L. A. García, "Agent-based management of non urban road meteorological incidents," in *Intl. Central and Eastern European Conf. on Multi-Agent Systems*. Springer, 2005, pp. 213–222.

[105] A. Garcia-Serrano, D. T. Vioque, F. Carbone, and V. Mendez, "Fipa-compliant mas development for road traffic management with a knowledge-based approach: The track-r agents," in *Proc. Challenges Open Agent Syst. Workshop*, 2003.

[106] B. Chen, H. H. Cheng, and J. Palen, "Integrating mobile agent technology with multi-agent systems for distributed traffic detection and management systems," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 1, pp. 1–10, 2009.

[107] F.-Y. Wang, "Toward a revolution in transportation operations: Ai for complex systems," *IEEE Intelligent Systems*, vol. 23, no. 6, pp. 8–13, 2008.

[108] R. T. van Katwijk, P. van Koningsbruggen, B. De Schutter, and J. Hellendoorn, "A test bed for multi-agent control systems in road traffic management," in *Applications of Agent Technology in Traffic and Transportation*. Springer, 2005, pp. 113–131.

[109] M. Ganzha, M. Paprzycki, W. Pawowski, P. Szmeja, and K. Wasielewska, "Streaming semantic translations," in *2017 21st Intl. Conf. on System Theory, Control and Computing (ICSTCC)*, Oct 2017, pp. 1–8.

[110] J. L. R. Moreira, L. Daniele, L. F. Pires, M. van Sinderen, K. Wasielewska, P. Szmeja, W. Pawlowski, M. Ganzha, and M. Paprzycki, "Towards iot platforms' integration semantic translations between W3C SSN and ETSI SAREF," in *Joint Proc. of SEMANTiCS 2017 Workshops co-located with the 13th Intl. Conf. on Semantic Systems (SEMANTiCS 2017), Amsterdam, Netherlands, September 11 and 14, 2017.*, 2017.

[111] R. Gonzalez-Usach, C. E. Palau, M. Julian, A. Belsa, M. A. Llorente, M. Montesinos, M. Ganzha, K. Wasielewska, and P. Sala, "Use cases, applications and implementation aspects for iot interoperability," in *Distributed Intelligence at the Edge and Human Machine-to-Machine Cooperation*, O. Vermesan and J. Bacquet, Eds. River Publishers, 2018, pp. 139–173.

[112] M. Ganzha, M. Paprzycki, W. Pawowski, P. Szmeja, and K. Wasielewska, "Alignment-based semantic translation of geospatial data," in *2017 3rd Intl. Conf. on Advances in Computing,Communication Automation (ICACCA) (Fall)*, 2017, pp. 1–8.

[113] M. Ganzha, M. Paprzycki, W. Pawlowski, P. Szmeja, and K. Wasielewska, "Semantic interoperability in the internet of things: An overview from the inter-iot perspective," *J. Network and Computer Applications*, vol. 81, pp. 111–124, 2017. [Online]. Available: https://doi.org/10.1016/j.jnca.2016.08.007

[114] M. Drozdowicz, M. Ganzha, and M. Paprzycki, "Semantic policy information point - preliminary considerations," in *ICT Innovations 2015 - Emerging Technologies for Better Living, Ohrid, Macedonia, 1-4 October, 2015*, 2015, pp. 11–19.

[115] M. Drozdowicz, M. Alwazir, M. Ganzha, and M. Paprzycki, "Graphical interface for ontology mapping with application to access control," in *Intelligent Information and Database Systems - 9th Asian Conf., ACIIDS 2017, Kanazawa, Japan, April 3-5, 2017, Proc., Part I*, 2017, pp. 46–55.

[116] M. M. Mesjasz, D. Cimadoro, S. Galzarano, M. Ganzha, G. Fortino, and M. Paprzycki, "Integrating jade and maps for the development of agent-based wsn applications," in *Intelligent Distributed Computing VI - Proc. of the 6th Intl. Symposium on Intelligent Distributed Computing - IDC 2012, Calabria, Italy, September 2012*, 2012, pp. 211–220.

[117] M. Ganzha, M. M. Mesjasz, M. Paprzycki, and M. Ouedraogo, "Inserting "brains" into software agents - preliminary considerations," in *Internet and Distributed Computing Systems - 7th Intl. Conf., IDCS 2014, Calabria, Italy, September 22-24, 2014. Proc.*, 2014, pp. 3–14.

[118] G. Frackowiak, M. Ganzha, M. Gawinecki, M. Paprzycki, M. Szymczak, C. Badica, Y. Han, and M. Park, "Adaptability in an agent-based virtual organization," *IJAOSE*, vol. 3, no. 2/3, pp. 188–211, 2009.

[119] K. Wasielewska, M. Ganzha, M. Paprzycki, C. Badica, M. Ivanovic, and I. Lirkov, "Multicriteria analysis of ontologically represented information," in *AIP Conf. Proc.*, vol. 1629, 2014, pp. 281–295.

[120] K. Wasielewska, M. Ganzha, M. Paprzycki, and I. Lirkov, "Developing ontological model of computational linear algebra – preliminary considerations," in *AIP Conf. Proc.*, vol. 1561, 2013, pp. 133–143.

[121] K. Wasielewska, M. Ganzha, M. Paprzycki, C. Badica, M. Ivanovic, I. Lirkov, and S. Fidanova, "Agents in grid extended to clouds," in *AIP Conf. Proc.*, vol. 1773, 2016.

[122] M. Drozdowicz,M. Ganzha,K. Wasielewska, M. Paprzycki,P. Szmeja, "Using ontologies to manage resources in grid computing: Practical aspects," in *Agreement Technologies*, ser. Law, Governance and Technology, S. Ossowski, Ed. Springer, 2013, pp. 149–168.

[123] M. Paprzycki, M. Drozdowicz, M. Ganzha, K. Wasielewska, I. Lirkov, R. Olejnik, and N. Attaoui, "Utilization of modified coregrid ontology in an agent-based grid resource management system," in *Proc. of the ISCA 25th Intl. Conf. on Computers and Their Applications, CATA 2010, March 24-26, 2010, Sheraton Waikiki Hotel, Honolulu, Hawaii, USA*, 2010, pp. 240–245.

[124] D. Lavbic, O. Vasilecas, and R. Rupnik, "Ontology-based multi-agent system to support business users and management," *CoRR*, vol. abs/1807.03646, 2018.

[125] M. Szymczak, G. Frackowiak, M. Gawinecki, M. Ganzha, M. Paprzycki, M. Park, Y. Han, and Y. T. Sohn, "Adaptive information provisioning in an agent-based virtual organization-ontologies in the system," in *Agent and Multi-Agent Systems: Technologies and Applications, Second KES Intl. Symposium, KES-AMSTA 2008, Incheon, Korea, March 26-28, 2008. Proc.*, 2008, pp. 271–280.

[126] G. Frackowiak, M. Ganzha, M. Gawinecki, M. Paprzycki, M. Szymczak, M. Park, and Y. Han, "Considering resource management in agent-based virtual organization," in *Intelligent Agents in the Evolution of Web and Applications*, 2009, pp. 161–190.

[127] S. K. Rhee, J. Lee, M. Park, M. Szymczak, G. Frackowiak, M. Ganzha, and M. Paprzycki, "Measuring semantic closeness of ontologically demarcated resources," *Fundam. Inform.*, vol. 96, no. 4, pp. 395–418, 2009.

[128] M. Kruszyk, M. Ganzha, M. Gawinecki, and M. Paprzycki, "Introducing collaborative filtering into an agent-based travel support system," in *Proc. of the 2007 IEEE/WIC/ACM Intl. Conf. on Web Intelligence and Intl. Conf. on Intelligent Agent Technology - Workshops, 2-5 November 2007, Silicon Valley, CA, USA*, 2007, pp. 439–443.

[129] M. Ganzha, M. Gawinecki, M. Paprzycki, R. Gasiorowski, S. Pisarek, and W. Hyska, "Utilizing semantic web and software agents in a travel support system," in *Semantic Web Technologies and eBusiness: Virtual Organization and Business Process Automation*, A. F. Salam and J. Stevens, Eds. Idea Publishing Group, Hershey, USA, 2006, pp. 325–359.

[130] J. Himoff, "Magenta logistics i-scheduler," in *4th Intl. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2005), July 25-29, 2005, Utrecht, The Netherlands*, 2005, pp. 159–160.

[131] O. Dikenelli, R. C. Erdur, and Ö. Gümüs, "SEAGENT: a platform for developing semantic web based multi agent systems," in *4th Intl. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2005), July 25-29, 2005, Utrecht, The Netherlands*, 2005, pp. 1271–1272.

[132] A. Cieslik, M. Ganzha, and M. Paprzycki, "Utilizing open travel alliance-based ontology of golf in an agent-based travel support system," in *Artificial Intelligence and Soft Computing - ICAISC 2008, 9th Intl. Conf., Zakopane, Poland, June 22-26, 2008, Proc.*, 2008, pp. 1173–1184.

[133] ——, "Developing and utilizing ontology of golf based on the open travel alliance golf messages," in *Web Information Systems and Technologies, 4th Intl. Conf., WEBIST 2008, Funchal, Madeira, Portugal, May 4-7, 2008, Revised Selected Papers*, 2008, pp. 133–156.

[134] P. A. Mitkas, A. L. Symeonidis, D. D. Kehagias, I. N. Athanasiadis, G. Laleci, G. Kurt, Y. Kabak, A. C. Acar, and A. Dogac, "An agent framework for dynamic agent retraining: Agent academy," *CoRR*, vol. cs.MA/0407025, 2004

[135] E. Sirin, B. Parsia, and J. Hendler, "Template-based composition of semantic web services," in *Agents and the Semantic Web, Papers from the 2005 AAAI Fall Symposium*, 2005, pp. 85–92.

[136] D. Fensel, *Ontologies - a silver bullet for knowledge management and electronic commerce*. Springer, 2001.

[137] K.-H. N. Bui and J. J. Jung, "Internet of agents framework for connected vehicles: A case study on distributed traffic control system," *Journal of Parallel and Distributed Computing*, vol. 116, pp. 89–95, 2018.

[138] H. L. Nguyen and J. E. Jung, "Socioscope: A framework for understanding internet of social knowledge," *Future Generation Computer Systems*, vol. 83, pp. 358–365, 2018.

[139] T. Golomb, Y. Mirsky, and Y. Elovici, "Ciota: Collaborative iot anomaly detection via blockchain," *arXiv preprint arXiv:1803.03807*, 2018.

[140] R. Casado-Vara, J. Prieto, F. De la Prieta, and J. M. Corchado, "How blockchain improves the supply chain: Case study alimentary supply chain," *Procedia computer science*, vol. 134, pp. 393–398, 2018.

[141] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, 2018.

[142] E. C. Ferrer, "The blockchain: a new framework for robotic swarm systems," in *Proceedings of the Future Technologies Conference*. Springer, 2018, pp. 1037–1058.

[143] G. Fortino and W. Russo, "Towards a cloud-assisted and agent-oriented architecture for the internet of things." in *WOA@ AI* IA*. Citeseer, 2013, pp. 60–65.

[144] T. Suganuma, T. Oide, S. Kitagami, K. Sugawara, and N. Shiratori, "Multiagent-based flexible edge computing architecture for iot," *IEEE Network*, vol. 32, no. 1, pp. 16–23, 2018.

[145] J. Xie, I. Howitt, and A. Raja, "Cognitive radio resource management using multi-agent systems," in *IEEE CCNC*, 2007.

[146] S. Shen and G. M. O'Hare, "Wireless sensor networks, an energy-aware and utility-based bdi agent approach," *International Journal of Sensor Networks*, vol. 2, no. 3/4, pp. 235–245, 2007.

[147] G. Fortino, R. Gravina, W. Russo, and C. Savaglio, "Modeling and simulating internet-of-things systems: A hybrid agent-oriented approach," *Computing in Science & Engineering*, vol. 19, no. 5, pp. 68–76, 2017.

[148] G. Fortino, W. Russo, C. Savaglio, W. Shen, and M. Zhou, "Agent-oriented cooperative smart objects: From iot system design to implementation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 11, pp. 1939–1956, 2017.

[149] G. Fortino, W. Russo, C. Savaglio, M. Viroli, and M. Zhou, "Modeling opportunistic iot services in open iot ecosystems."

[150] B. Molina, C. E. Palau, G. Fortino, A. Guerrieri, and C. Savaglio, "Empowering smart cities through interoperable sensor network enablers," in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2014, pp. 7–12.

[151] S. Yamaguchi, S. Tsugawa, and K. Nakahori, "An analysis system of iot services based on agent-oriented petri net pn2," in *2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*. IEEE, 2016, pp. 1–2.

[152] P. C. Ccori, L. C. C. De Biase, M. K. Zuffo, and F. S. C. da Silva, "Device discovery strategies for the iot," in *2016 IEEE International Symposium on Consumer Electronics (ISCE)*. IEEE, 2016, pp. 97–98.

[153] R. Casadei, G. Fortino, D. Pianini, W. Russo, C. Savaglio, and M. Viroli, "Modelling and simulation of opportunistic iot services with aggregate computing," *Future Generation Computer Systems*, vol. 91, pp. 252–262, 2019.

[154] S. N. Han, G. M. Lee, N. Crespi, K. Heo, N. Van Luong, M. Brut, and P. Gatellier, "Dpwsim: A simulation toolkit for iot applications using devices profile for web services," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*. IEEE, 2014, pp. 544–547.

[155] T. Jung, N. Jazdi, and M. Weyrich, "A survey on dynamic simulation of automation systems and components in the internet of things," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2017, pp. 1–4.

[156] Q. Zhang and C. Cao, "A multi-agent simulation model combined with evolutionary game for cooperative patterns of iot," *JOURNAL OF INFORMATION &COMPUTATIONAL SCIENCE*, vol. 10, no. 10, pp. 2933–2942, 2013.

[157] M. Dyk, A. Najgebauer, and D. Pierzchala, "Sensesim: An agent-based and discrete event simulator for wireless sensor networks and the internet of things," in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. IEEE, 2015, pp. 345–350.

[158] G. Fortino, F. Messina, D. Rosaci, and G. M. Sarne, "Using blockchain in a reputation-based model for grouping agents in the internet of things," *IEEE Transactions on Engineering Management*, 2019.

7. Appendix

From 2010, year in which the IoT turn on the spotlight, the number of the publications related to the ABC and the IoT has grown steadily. According to Scopus, 1565 works have been published to date, with a marked rise in the last years, as reported in Fig.4 (a). Interestingly, as shown in Fig.4 (b), there are still few books, chapters and editorials on this topic, a good number of journals articles but many more conference papers. This could be due to the recently risen interest on agent-based IoT: indeed, a general practice is to present preliminary results in workshop/conference papers and develop it into a journal with new material later on. As matter of fact, the number of published journal articles in the last two years has doubled. Another noteworthy fact concerns the subject area of these publications: as shown in Fig.5, most of them are (obviously) related to computer science, but also in energy, mathematics, decision science, etc. This confirm the suitability and flexibility of the ABC, perfectly matching with the multifaceted nature of the IoT.

With respect to the publications surveyed in this article (a subset of the available ones, given the wide of the state-of-the-art), we focused on the most relevant ones according to the number of obtained citations, exploitation in/derivation from valuable (inter)national research projects, and personal experience. In particular, as shown in Fig.6, we covered the whole time windows 2010-2019 and preferred journal articles, if possible.

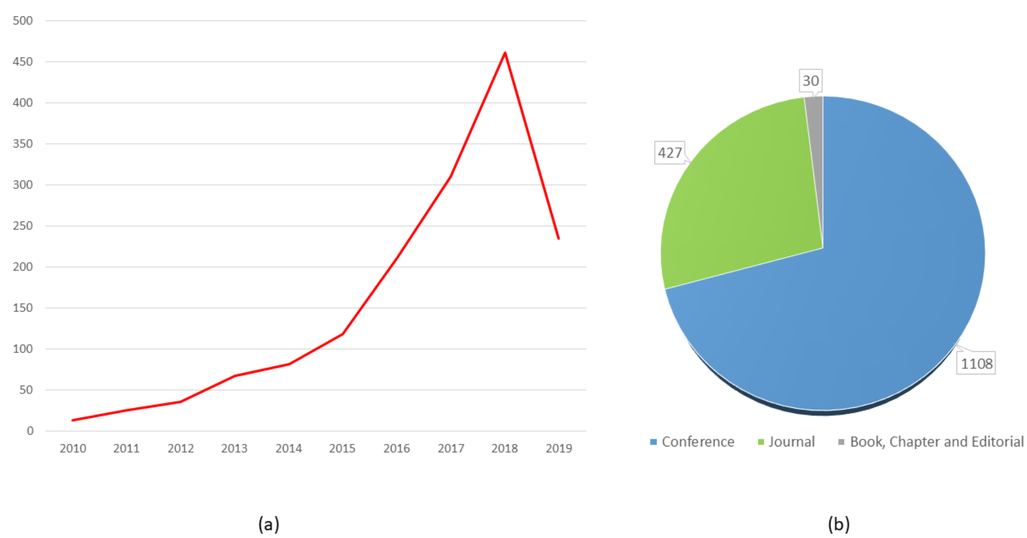

Figure 4: Agent-based Internet of Things publications (a) per year and (b) per type.

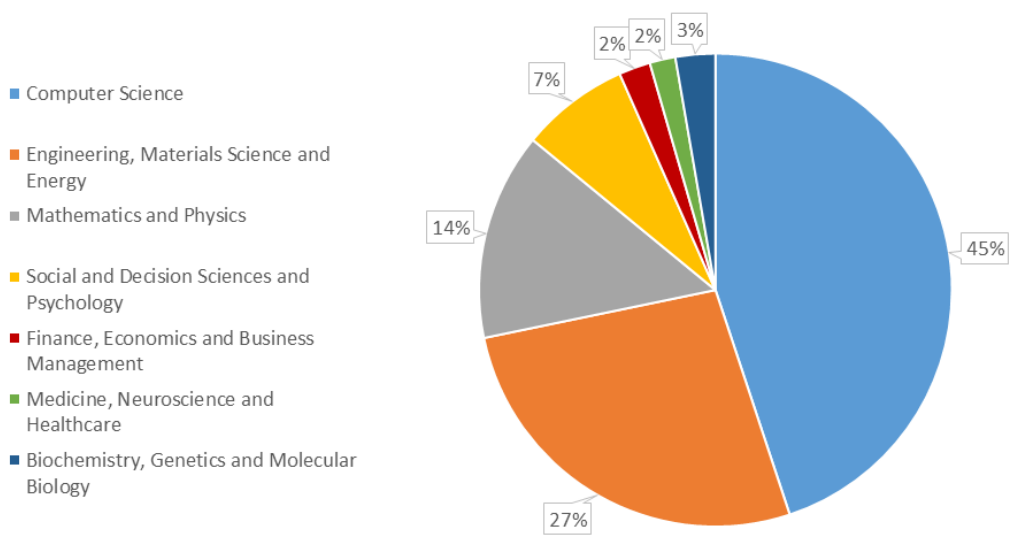

Figure 5: Agent-based Internet of Things publications per subject area.

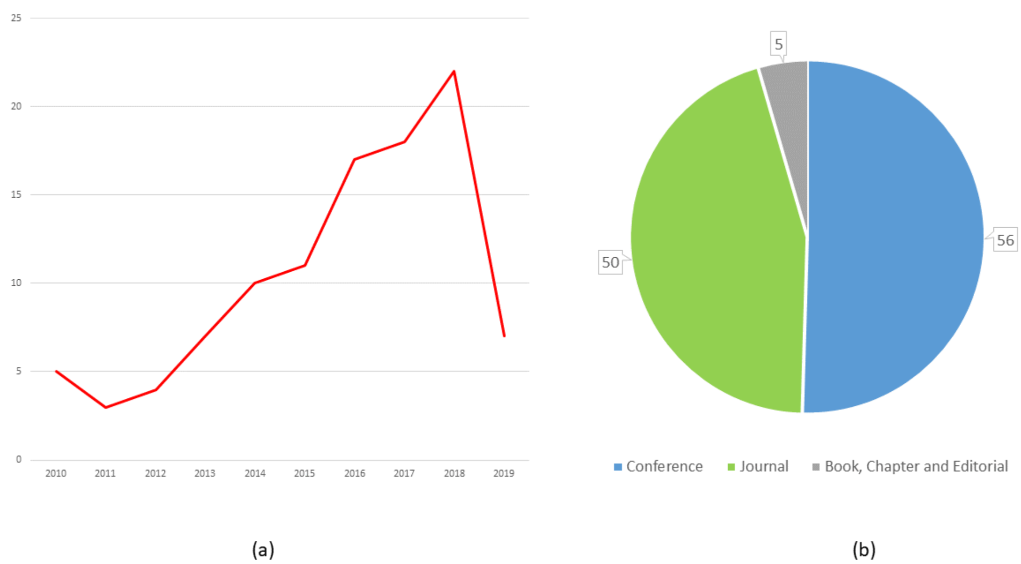

Figure 6: Surveyed Agent-based Internet of Things publications (a) per year and (b) per type.