

Augmented Virtual Reality: Combining Crowd Sensing and Social Data Mining with Large-Scale Simulation Using Mobile Agents for Future Smart Cities [†]

Stefan Bosse ^{1,2,*} and Uwe Engel ³

¹ Department of Mathematics & Computer Science, University of Bremen, Robert Hooke Str. 5, 28359 Bremen, Germany

² Faculty of Computer Science, University of Koblenz-Landau, 56070 Koblenz, Germany

³ Department of Social Sciences, University of Bremen, 28359 Bremen, Germany; uengel@uni-bremen.de

* Correspondence: sbosse@uni-bremen.de

[†] Presented at the 5th International Electronic Conference on Sensors and Applications, 15–30 November 2018; Available online: <https://ecsa-5.sciforum.net>.

Published: 14 November 2018

Abstract: Augmented reality is well known for extending the real world by adding computer-generated perceptual information and overlaid sensory information. In contrast, simulation worlds are commonly closed and rely on artificial sensory information generated by the simulator program or using data collected off-line. In this work, a new simulation paradigm is introduced, providing augmented virtuality by integrating crowd sensing and social data mining in simulation worlds by using mobile agents. The simulation world interacts with real world environments, humans, machines, and other virtual worlds in real-time. Mobile agents are closely related to bots that can interact with humans via chat blogs. Among the mining of physical sensors (temperature, motion, position, light, ...), mobile agents can perform Crowd Sensing by participating in question–answer dialogs via a chat blog provided by a WEB App that can be used by the masses. Additionally, mobile agents can act as virtual sensors (offering data exchanged with other agents). Virtual sensors are sensor aggregators performing sensor fusion in a spatially region.

Keywords: multi-agent systems; augmented reality; augmented virtuality; simulation; crowd sensing; sensorfusion; socio-technical systems

1. Introduction

Today administration and management of public services and infrastructure relies more and more on user data collected by many domestic and private devices, including smart phones and Internet services. User data and user decision making has a large impact on public decision-making processes, for example, plan-based traffic flow control. Furthermore, intelligent behavior, i.e., cognitive, knowledge-based, adaptive, and self-organizing behavior based on learning, emerges rapidly in today's machines and environments. Social science itself exerts influence on public opinion formation. We observe a rise of Artificial Intelligence (AI) in the shape of computational methods of analysis of data and meta data extracted, e.g., from the web. This statement alludes to the rise of “computational social science” (CSS), the accompanying shift from survey methods to computational methods, and possible implications of this development; usage of different data, different methods, exposition to different threats to data quality (concerning sampling, selection effects, measurement

effects, data analysis), and different rules of inference are likely to result in comparably different conclusions, public reports, and hence in different input to public opinion formation.

Intelligent behavior, i.e., cognitive, knowledge-based, adaptive, and self-organizing behaviour based on learning, emerges rapidly in today's machines. Against this background we are interested in an experimental field study to identify ways bots affect the decision-making process and opinion dynamics in smart environments, including the social media. Bots are assumed to represent a substantial factor of influencing opinion formation in the web.

This influence is the more relevant for society. One major question that results is the ability of humans to identify such intelligent machines. However, being capable of distinguishing humans from machines applies to bots as well. Bots should learn to recognize other bots to avoid undesired cyclic bot–bot interactions and conversation with unpredictable results. Hence, the well-known Turing test originally defined to test machines by humans must be extended to test intelligent machines by machines. Among hardware robots capable of manipulating physical worlds and interacting with humans, there are software robots that manipulate data only by being even capable to interact with humans. One prominent example for a software robot is chat or social bot. Hardware and software robots can be considered close together in a generalized way by an association to the agent model. Additionally, real and artificial humans can be represented by the agent model. Multi-agent systems enable modelling of agent interaction and emergence behaviour. Agent-based modelling and simulation is a suitable methodology to study the interaction and behavior of large groups of relevant human actors, hardware, and software robots, using methods with inherent elements of AI (e.g., learning algorithms) or which use data which may be influenced partly by bots.

Mobile devices like smart phones are valuable sources for social data [1], either by explicit participation of users providing first class data (e.g., performing surveys or polls) or implicitly by collecting secondary class data, i.e., traces of device sensor data delivering, e.g., actual position, ambient conditions, network connectivity, digital media interaction, and so on. Crowd Sensing and Social Data Mining as a data source contribute more and more to investigations of digital traces in large-scale machine–human environments characterized by complex interactions and causalities between perception and action (decision-making).

It is difficult to study such large-scale data collection, data mining, and their effect on public services like smart traffic control in field studies. Agent-based modelling of socio-technical systems is well established [2], however, it is commonly applied in an artificial world, i.e., a simulation is performed in virtual reality worlds only to derive and proof models under hard limitations. In this work, a new concept and framework for augmented virtual reality simulation is introduced, suitable for, but not limited to, the investigation of large-scale socio-technical systems. Mobile agents are already used successfully in-field Crowd Sensing [3]. In this work, mobile agents are used to combine in-field ubiquitous Crowd Sensing, e.g., performed by mobile devices, with simulation. The agents can operate both in real and virtual worlds including games and fuse both worlds with seamless migration. Agents are loosely coupled to their environment and platform and interact with each other via Tuple Spaces (generative communication) and via uni-cast or broadcast signals. Mobility is provided by agent process snapshot migration between platforms.

The agents are programmed in JavaScript and executed by the JavaScript Agent Machine (JAM) that can be deployed on a wide range of host platforms (mobile devices, servers, IoT devices, WEB browsers [4]). Simulation is performed by the Simulation Environment for JAM (SEJAM) [5,6], extending JAM with a simulation layer providing virtualization, visualization of 2D- and 3D worlds, and an advanced simulation control with a wide range of integrated analysis and inspection tools to investigate the run-time behavior of distributed systems. Each JAM node is capable of processing thousands of agents concurrently. JAM and SEJAM nodes can be connected in clusters and in the Internet (of Things) enabling large-scale “real-world”-in-the-loop simulations with Millions of agents.

Such large-scale simulations produce Big Data and can offer an enhanced statistical strength beyond Big Data analysis by understanding the correlation between data and cause. Due to the loosely coupling of agents, the emergence of strong heterogeneous Multi-agent Systems and their

cooperation can be investigated with one unified framework without disruption. Agents can represent different behavior, goals, and individuals, like chat bots, machines, robots, or artificial humans, and their interaction with virtual and real individuals. We expect to simulate large-scale agent societies with agent population beyond one million individual agents to get statistical strength. Agents are mobile with respect to networks of processing platforms in the real world and with respect to the location and interaction domain in virtual worlds. The agent behavior, planning, and decision making is based on an individual and global (domain specific) knowledge base.

There is a significant difference between closed simulations and simulations coupled with real world and real-time environments. Closed simulations are performed on a short time-scale with pre-selected use cases and input data. The simulation can be processed step-wise without a relation to a physical clock. In contrast, closed-loop simulation requires continuous simulation on a large time-scale creating Big Data volumes. A relation to a physical clock is also required.

The proposed agent-based crowd sensing and simulation framework is not limited to simulation aspects only. As Bellavista et al. [7] stated, Mobile Crowd Sensing can be considered as an enabling technology for people as a service mobile computing platform. For example, it is well suited for analysis and classification of the Vehicular Traffic Distribution in an Urban Area [8].

2. Intelligent Machines and Socio-Technical Systems

In the following, robots are considered to be intelligent machines (AI bots) posing some kind of autonomous behaviour, learning, and interaction with environments and humans. There is a micro-context of everyday life and the macro-context of public decision and opinion formation related to human-robot interaction. Social behavior of humans and interaction with intelligent machines will depend on the way intelligent machines will be controlled and the way they interact with humans and other intelligent machines. For example, chat bots influence human opinions, but humans and other chat bots can influence the behavior and knowledge base of chat bots, too. There are different involved ways of mutual affection that are difficult to investigate in classical field studies. The proposed simulation method, integrating real world environments by using mobile agents, can boost investigations of such complex interacting environments. For example, decisions of traffic participants (e.g., car drivers) cause wide range effects on global behavior (emergence), and centralized or decentralized traffic control will influence individual behaviour.

Since both sides (humans and bots) may learn from each other in this interaction, the studies shall explore possibilities of targeted responses of the human side. That way we expect to reveal options of getting influence on possible ways AI will shape future society. In addition, we expect to find out factors in robot—human interaction that let AI gain acceptance in society. This list of factors shall, amongst others, include the visibility of AI during a communication (e.g., can a human discern whether a comment comes from a bot or a human and what difference makes this to his/her response?) and different programmable behavioral and ethical principles in robot—human interaction (by systematically varying the reward function). Furthermore, since we expect in the medium and long-term, respectively, a trend towards the use of robots that not only assist but replace humans in the daily communication, we project the development and exploration of a first scenario in a homely environment, and the evaluation of its applicability in practice.

At society's macro level, robot—human interaction takes place too, though less visible and less direct. We plan an investigation of the case of public decision and opinion formation. We project an exploration of how this process is shaped in the interaction of groups of relevant human actors who interact with science through its reports of findings of projects which either use methods with inherent elements of AI (e.g., learning algorithms), or which use data which may be influenced partly by bots. In this regard, the study is targeted on possible emergent effects of AI at the system level of society.

3. Framework

The basic framework concept is shown in Figure 1. The JavaScript Agent Machine (JAM) is the central component of the distributed agent processing network, consisting of one or more simulators

connected to one or more external JAM instances via Internet connectivity. The Simulation Environment for JAM (SEJAM) is based on a generic JAM instance, providing virtualization with additional simulation control and visualization layers on the top. A JAM world consists of a set of virtual JAM nodes. An agent within a world is assigned to a specific virtual JAM (vJAM) instance. Each vJAM has its own state (agent processing, tuple data bases, ...), but agents are executed by one shared agent task scheduler loop.

Each of the JAM nodes is identified by a randomly created node name using phonemes generators (so humans can remember the names). Based on the used random generator algorithms, the names are more or less unique. Since JAM nodes are commonly connected peer-to-peer in mesh-like networks, only node names of neighboring nodes have to be unique.

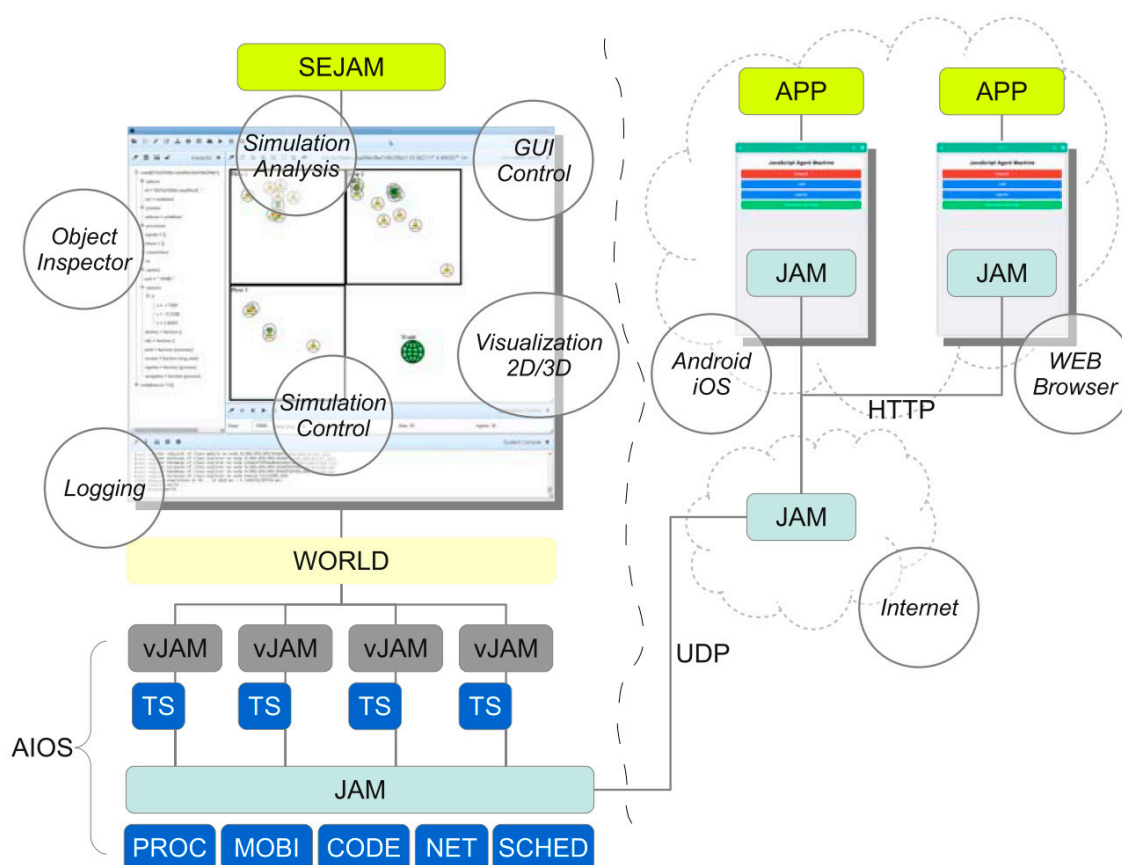


Figure 1. Principle concept of closed-loop simulation for augmented virtuality: **(Left)** Simulation Framework based on the JAM (JavaScript Agent Machine) platform **(Right)** Mobile and non-mobile devices executing the JAM platforms connected to the virtual simulation world (via Internet).

Neither agents operating in artificial worlds nor agents and humans operating in the real world are aware of this mixed reality world. A human interacting with (bot) agents (via a chat, see Section 6) cannot recognize the origin of the agent. JAM agents are programmed in AgentJS, basically JavaScript. There are agent classes from which agents can be instantiated at run-time (created by agents or by the platform). An agent class template is a constructor function defining the agent behavior and data (body variables). The agent behavior is modelled with an Activity-Transition Graph (ATG), shown in Figure 2. An agent is executed in a sandbox environment isolating the agent from other agents and the platform internals. The agent can access the platform (and can interact with other agents) via the Agent Input Output System (AIOS), providing a unique API. Mobile agents carry a snapshot of their state (data and control) and their behavior (code).

```

function agentclassname (param1,param2,..) {
  this.data = expr;
  this.act = { Activities
    a1: function () { statements },
    ..
    an: function () { statements },
  }
  this.trans = { Transitions
    ai:aj,
    aj:function () { return activity }, ..
  }
  this.on = { Signal and Exception handler
    signame: function (arg) { statements }, ..
  }
  this.next = activity
}

```

Figure 2. An agent class template defining the agent behaviour and data.

4. Mobile Agents and Communication

Mobile agents are used in this work for distributed data processing in the real and virtual simulation world seamlessly. It is assumed that the real-world environments consist of communication access points (i.e., beacons, e.g., using WLAN, WWAN communication technologies) and mobile devices (smart phones). Commonly, there is no globally visible organizational and network structure, i.e., agents cannot rely on a network world model to reach and identify specific devices. Each JAM node provides connectivity information, i.e., a list of all connected JAM nodes. Commonly, JAM nodes are connected peer-to-peer in mesh-like networks via TCP/HTTP or UDP connections.

Agents can communicate with each other either by exchanging tuples via a tuple space data base or by using signals (lightweight messages). Tuple space access is generative communication, i.e., the lifetime of tuples can exceed the lifetime of the generating agent. Additionally, tuple space communication is data-driven and anonymous (sender and receiver need no knowledge about each other).

Mobile agents can be used to carry information from one location to another. Moreover, mobile devices carrying mobile agents can be used to collect, carry, and distribute data within large regions via tuple spaces, shown in principle in Figure 3. The virtual simulation world is just another region in the mobility regions of agents.

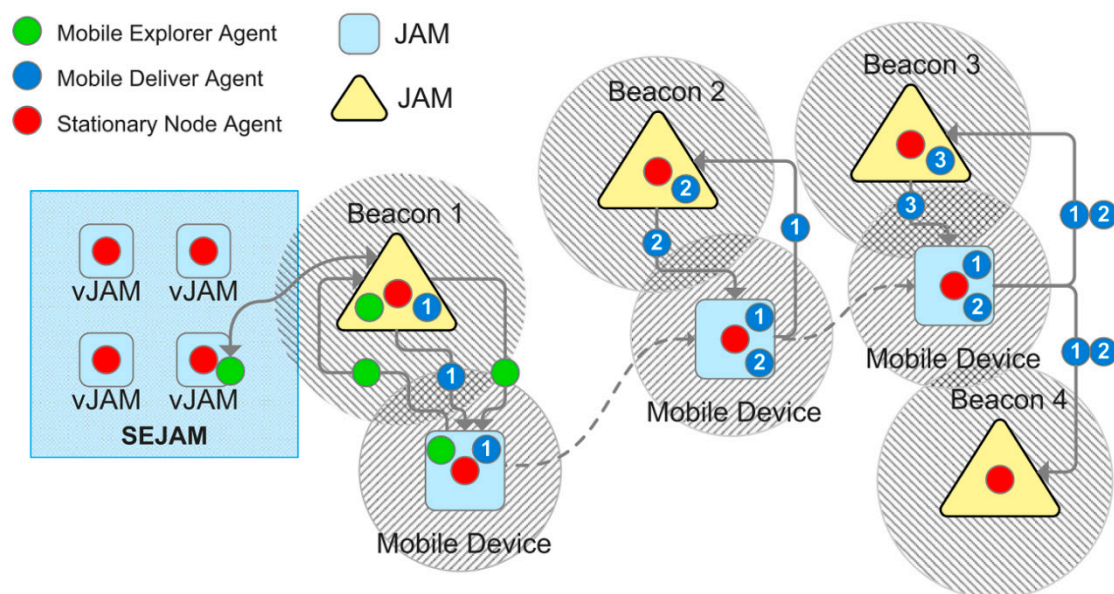


Figure 3. Principle network topology consisting of the virtual simulator world, connected to spatially distributed non-mobile beacons, mobile devices connected temporarily to beacons, and mobile agents used for wide-range interaction.

5. Virtual Sensors, Big Data, Data Mining, and Digital Traces

In Reference [9], a large-scale sensing application is composed of virtual sensors. A virtual sensor is a software component acting as a sensor data aggregator. Each sensor component is treated as a sensor, processing an input stream and computing an output data stream. Each physical sensor (data source) is also a “data stream” transformer, but based on physical principles. In this work, user-provided first-class data are also considered physical sensors. A virtual sensor is a processing system as well as a data storage system (data base). In this work, virtual sensors are represented by mobile agents, performing the sensing, aggregation, and application (or delivery) of accumulated and processed sensor data. The mobility enables self-organizing and adaptive mining systems controlled by environmental constraints rather than by individual users. In Reference [9], users using a smart phone application are considered agents. This role is replaced in this work by the deployment of agents that perform tasks autonomously.

A virtual sensor consists of different components. The environment of a sensor is a set of input streams of data generated from physical or virtual sensors. The environment defines the context within which the virtual sensor operates. An aggregator processes the input streams and performs sensor data fusion, shown in Figure 4 (left). A filter produces a set of output streams. Virtual sensors can be coupled in graph-like virtual networks via agent interaction (using signal messages and tuple exchange), shown in Figure 4 (right). These virtual sensor network compose processing chains (user defined or ad-hoc and self-organizing). Virtual sensors can operate in real-world environments (e.g., executed on mobile devices) or in virtual worlds (simulation). Agents implement the sensor data aggregator and filter function, performing the fusion, storage, and communication, and represent mobile transport entities. Agent processing is virtualized by a unified Agent Processing Platform.

Digital trace data are not error-free. Even unobtrusive measures may suffer from measurement error. Hox [10] points out that the “promise of big data collection is that expensive data collections are replaced by less costly ‘found’ data, and that sampling is replaced by analyzing all existing data, avoiding sampling error. If subjects are aware of being research subjects, this influences their behavior, including answer behavior in surveys. Such research is reactive; subjects react to being researched. On the other hand, unobtrusive measures “collect data already generated” and hence “cannot influence that behavior.” Shah et al. [11] also refer to the topic of “unobtrusive measures”. Crowd data can be considered as “naturally occurring” data in contrast to data from surveys and experiments and the typically associated biases through “experimenter effects” and “self-

Each agent migrating to a device operated by humans (i.e., smart phones, car board computer, or fixed terminals) can ask questions via the APP tuple space by inserting a question tuple. This question request tuple is evaluated and assessed by the mediator agent performing security and privacy checks, and finally passing the question to the chat blog waiting for human participation [13]. The answer (if any) is passed back to the original requesting agent by an answer tuple, again previously checking privacy and security concerns. In doubt, the question or answer is discarded or modified (e.g., filtered or annotated with additional warnings). The mediator agent performs question request scheduling to satisfy agent request and human interaction capabilities (high question rates will decrease the user interest and motivation to answer questions).

A typical chat dialog is shown in Figure 5. It consists of question–answer snippets processed and controlled by the mediator agent operating on each device with user interaction.

The mediator agent can pass questions (or just standalone messages) to the user chat via simple operations providing a question text, possible answers or input fields, and a callback function handling the result:

```
message('Mediator','Now I will mediate questions from agents!');
question('Guest','Can you make a guess of the room temperature?',{
    size: 14,
    icon: 'thermometer-empty',
    value: this.temperature,
    // show the current temperature as default
    sub_type: 'number',
    placeholder: '20'
},function (res) {
    this.temperature=res;
},timeout);
```

A survey agent arrived from an external location can ask questions via the tuple space, which are processed by the mediator agent and transformed in chat operations:

```
out(['Question','Guest','Can you make a guess of the room temperature?',
    {value: 20, type:'number', icon:'thermometer-empty'}]);
try_inp(timeout,
    ['Answer','Guest',
    'Can you make a guess of the room temperature?',_],
    function (x) { if (x) this.result = x[3] });
message(['Message','Guest',
    'Are you kidding me? Temperature is never '+this.result])
```

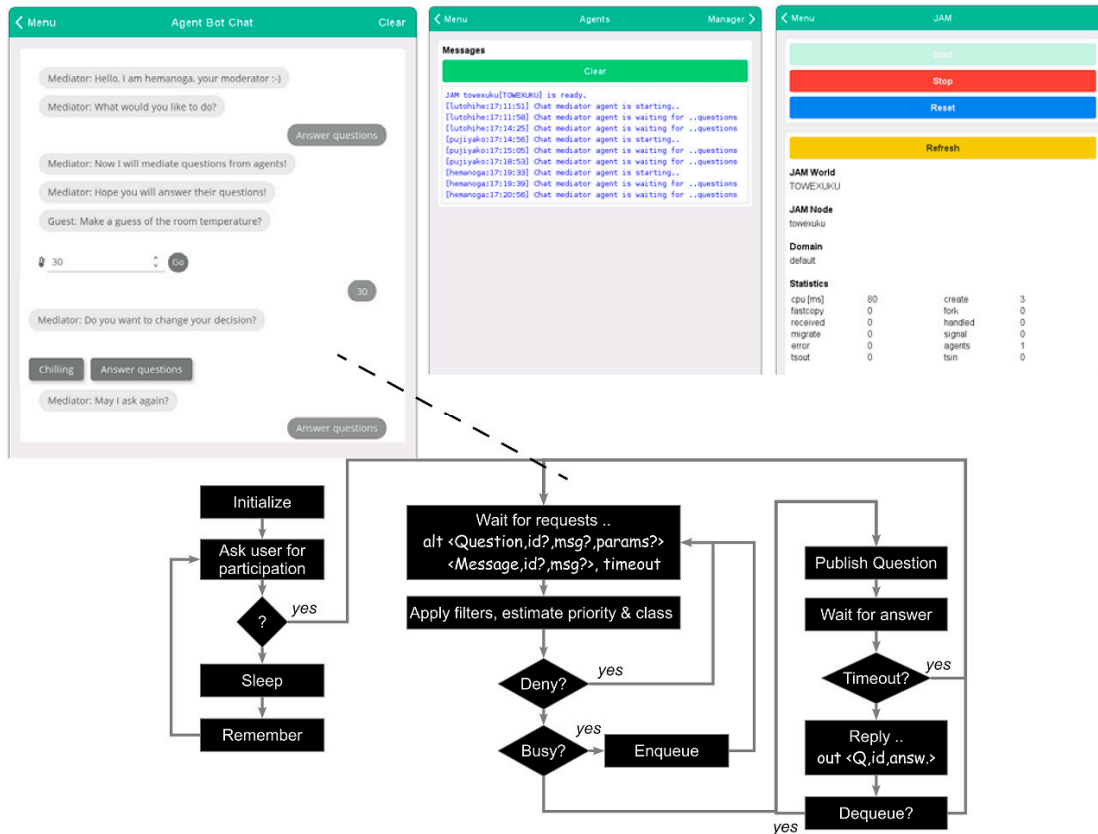



Figure 5. (Top, left) A typical human-agent chat dialog page in the App; (Top, center) agent message monitor page; (Top, right) JAM (JavaScript Agent Machine) control dialog page (Bottom) Interaction flow of Mediator agent.

7. Use Case

Although this work is in a preliminary stage, the tools for distributed Crowd Sensing coupled to simulation are available. The goal of a simple demonstrator is the simulation and investigation of Crowd Interaction with Smart Cities by implementing a self-organized control of ambient light conditions (e.g., in streets or buildings) using Crowd Sensing.

A simple demonstrator consists of an artificial street area with moving nodes (representing humans interacting with mobile devices or machines), beacons (access points) for sensor aggregation and distribution, and some external beacons connected to the Internet, enabling the connection to mobile devices via the Internet, and finally Smart Light Devices illuminating streets and buildings, shown in Figure 6.

The goal of explorer agents sent out by mobile devices is information mining in the outside world via Internet deployed agent processing platforms that can be accessed by simulation nodes and WEB browser Apps. The collected information is passed back to the root node (e.g., a mobile devices of a passenger) to assist decision making and navigation.

The explorer agents have to estimate the position of the root node by performing sensor mining from surrounding devices they are visiting and from the outside world (far away) by asking questions answered by humans via the WEB App chat dialog. Based on answered questions regarding the current user location, the satisfaction of ambient light condition, and an optional fusion with device sensor data (light, position, ...), actions are directed to smart light control devices to change the light condition in streets and buildings by using mobile notification agents. The action planning is based on the Crowd demands and energy saving constraints. If action is required, mobile notification agents are sent out to neighboring nodes to change light intensity based on directed diffusion, random walk, and divide-and-conquer.

The simulation with the current SEJAM2 simulator was able to be performed with more than 1000 nodes, hundreds of beacons, and more than 10,000 explorer agents. Real time values of one simulation step depends on the number of active agents to be processed, node and agent mobility (graphics and communication), and ranges from 1 ms to 1 s.

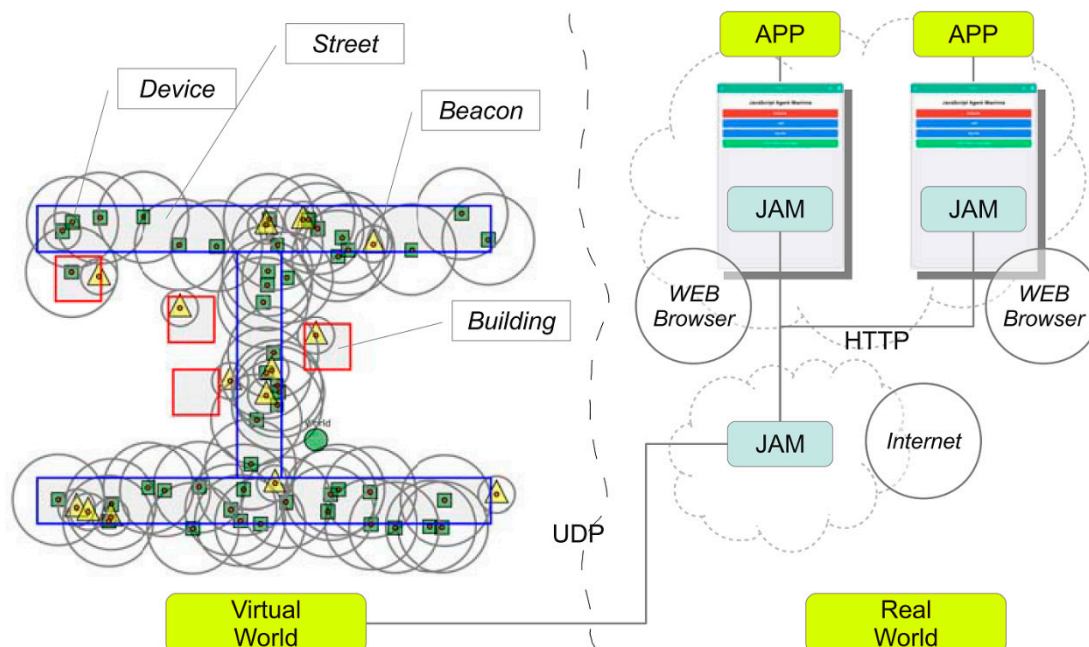


Figure 6. A demonstrator: An artificial street area consisting of street segments (blue rectangle) and buildings (red rectangles), beacons (yellow triangles), populated with stationary and mobile devices (green squares). The grey circles around beacons and nodes show the wireless communication range. Only overlapping circles connect.

8. Conclusions and Outlook

Mobile agents can be used to fusion real and virtual worlds composing augmented virtuality. The deployment of the JAM agent processing platform enables large-scale simulation of extended and complex socio-technical systems, e.g., complex traffic situations and smart wide-range traffic management, large-scale social media interaction, Crowd Sensing, and many more complex systems with unknown emergence behavior. Moreover, from the point of view of social science, decision making, opinion formation, and voting processes can be studied with a unified simulation and data mining framework. The tools (the big machine) are available and the suitability can be shown with demonstrators. Future work has to find suitable research questions that can be investigated with the machine from social sciences, smart city and facility management, production and logistics environments, warehouse systems, and many more fields of applications. Additionally, real-time issues by agent interaction in real and virtual worlds differing in real time scale have to be investigated.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ganti, R.K.; Ye, F.; Lei, H. Mobile Crowd Sensing: Current State and Future Challenges. *IEEE Commun. Mag.* **2011**, *49*, 32–39.
2. Van Dam, K.H.; Lukszo, I.; Zofia, N. (Eds.) *Agent-Based Modelling of Socio-Technical Systems*; Springer: Berlin/Heidelberg, Germany, 2013.
3. Leppäne, T.; Lacasia, J.Á.; Tobe, Y.; Sezaki, K.; Riekk, J. Mobile Crowd Sensing with mobile agents. *Auton. Agents Multi-Agent Syst.* **2017**, *31*, 1–35.

4. Bosse, S. Mobile Multi-Agent Systems for the Internet-of-Things and Clouds using the JavaScript Agent Machine Platform and Machine Learning as a Service. In Proceedings of the IEEE 4th International Conference on Future Internet of Things and Cloud, Vienna, Austria, 22–24 August 2016; doi:10.1109/FiCloud.2016.43.
5. Bosse, S. Smart Micro-scale Energy Management and Energy Distribution in Decentralized Self-Powered Networks Using Multi-Agent Systems. In Proceedings of the FedCSIS Conference, 6th International Workshop on Smart Energy Networks & Multi-Agent Systems, Poznan, Poland, 9–12 September 2018.
6. Bosse, S.; Pournaras, E. An Ubiquitous Multi-Agent Mobile Platform for Distributed Crowd Sensing and Social Mining. In Proceedings of the FiCloud 2017: The 5th International Conference on Future Internet of Things and Cloud, Prague, Czech Republic, 21–23 August 2017.
7. Bellavista, P.; Berrocal, J.; Corradi, A.; Foschini, L. Mobile Crowd Sensing as an Enabler for People as a Service Mobile Computing. In Proceedings of the ADHOC-NOW 2017: Ad-hoc, Mobile, and Wireless Networks, Messina, Italy, 20–22 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10517.
8. Zambrano-Martinez, J.L.; Calafate, C.T.; Soler, D.; Cano, J.-C.; Manzoni, P. Analysis and Classification of the Vehicular Traffic Distribution in an Urban Area. In Proceedings of the ADHOC-NOW 2017: Ad-hoc, Mobile, and Wireless Networks, Messina, Italy, 20–22 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10517.
9. Pournaras, E.; Moise, I.; Helbing, D. Privacy-preserving Ubiquitous Social Mining via Modular and Compositional Virtual Sensors. In Proceedings of the IEEE 29th International Conference on Advanced Information Networking and Applications, Gwangju, Korea, 25–27 March 2015.
10. Hox, J.J. Computational Social Science Methodology, Anyone? *Methodology* **2017**, *13*, 3–12, doi:10.1027/1614-2241/a000127
11. Shah, D.V.; Capella, J.N.; Neuman, W.R. Big Data, Digital Media, and Computational Social Science: Possibilities and Perils. *Ann. Am. Acad. Political Soc. Sci.* **2015**, *659*, 6–13, doi:10.1177/00027162155720810.
12. Mahrt, M.; Scharkow, M. The Value of Big Data in Digital Media Research. *J. Broadcast. Electron. Media* **2013**, *57*, 20–33, doi:10.1080/08838151.2012.761700.
13. Musciotto, F.; Delpriori, S.; Castagno, P.; Pournaras, E. Mining Social Interactions in Privacy-preserving Temporal Networks. In Proceedings of the Advances in Social Networks Analysis and Mining (ASONAM), San Francisco, CA, USA, 18–21 August 2016.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).