A METHODOLOGY FOR DESIGN SPACE EXPLORATION OF REAL-TIME LOCATION SYSTEMS

D.A. Kirov[a], R. Passerone[a], A.A. Ozhiganov[b]

[a] University of Trento, Trento, 38123, Italy
[b] ITMO University, Saint Petersburg, 197101, Russian Federation
Corresponding author: redgyy@gmail.com

Abstract

Scope of Research. This paper deals with the problem of design space exploration for a particular class of networked embedded systems called Real-Time Location Systems (RTLS). **Methods.** The paper contains a clear and detailed plan of an ongoing research and could be considered as a review, a vision and a statement of objectives. Analytical and formal methods, simulation and automated verification will be involved in the research. **Main Results.** Analysis of the state of the art (current design flow, existing simulation tools and verification techniques) has revealed several limitations for performing efficient design space exploration of RTLS, especially for safety-critical applications. The review part of the paper also contains a clear problem statement. The main outcome of this research is the proposed vision of a novel methodology for determining the best-suited technology and its configuration from the space of potential solutions. In particular, it is planned to extend an existing simulation framework and apply automated verification techniques. The latter will be used for checking simulation results and also for exploring different system configuration alternatives, that is, to optimize the design, which is a novel approach. A case study for validating the methodology is also proposed. **Practical Significance.** The proposed methodology will highly increase the breadth of design space exploration of RTLS as well as the confidence on taken design decisions. It will also contribute to optimizing the design.

Keywords

design space exploration, localization, positioning, real-time location systems, RTLS, simulation, automated verification, statistical model checking, networked embedded systems, embedded systems, cyber-physical systems, CPS.

УДК 004.0316, 004.942, 004.78

МЕТОДИКА ИЗУЧЕНИЯ ПРОСТРАНСТВА ПРОЕКТНЫХ РЕШЕНИЙ ДЛЯ СИСТЕМ ПОЗИЦИОНИРОВАНИЯ ОБЪЕКТОВ В РЕАЛЬНОМ ВРЕМЕНИ

Д.А. Киров[a], Р. Пассероне[a], А.А. Ожиганов[b]

[a] Университет Тренто, г. Тренто, 38123, Италия
[b] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация
Адрес для переписки: redgyy@gmail.com

Аннотация

Предмет исследования. В данной статье рассматриваются проблемы изучения пространства проектных решений для одного из классов сетевых встроенных систем, носящего название систем позиционирования объектов в реальном времени. **Методы.** Статья содержит подробный план проводимых в настоящий момент исследований и может рассматриваться как обзорная и как постановочная. В работе будут задействованы аналитические и формальные методы, имитационное моделирование и автоматизированная верификация. **Основные результаты.** Анализ имеющихся процессов проектирования, существующих симуляторов и методов верификации выявил ряд ограничений на пути эффективного исследования проектных решений для систем позиционирования. Данные проблемы сформулированы в обзорной части статьи. Главным результатом работы следует считать предлагаемую новую методику, позволяющую выбрать наиболее подходящую аппаратуру и ее конфигурацию из набора потенциальных проектных решений. В частности, планируется расширить существующий инструментарий имитационного моделирования и внедрить методы автоматизированной верификации моделей. Данные методы будут

использоваться для проверки результатов симуляций, а также для изучения различных альтернатив конфигурирования системы, т.е. для оптимизации проекта, что является одним из наиболее существенных критериев новизны. В работе также описан практический пример для проверки разрабатываемой методики. **Практическая значимость.** Предложенная методика значительно увеличит эффективность анализа проектных решений для систем позиционирования объектов в реальном времени и уверенность в их принятии. Также будет внесен существенный вклад в оптимизацию проектируемых систем.

Ключевые слова

изучение пространства проектных решений, локализация, позиционирование, системы позиционирования объектов в реальном времени, имитационное моделирование, симуляция, автоматизированная верификация, статистическая проверка моделей, сетевые встроенные системы, встраиваемые системы, киберфизические системы.

Introduction

Recent advances in microelectromechanical systems (MEMS), hardware and communication technologies have led to a significant change in understanding embedded systems both in scientific and industrial communities and led to a more global and interdisciplinary concept, that of Cyber-Physical Systems (CPS). In such systems, one distinguishes a physical part, which could be a mechanical device, such as a robot, a car, an aircraft or something else, and a "cyber" part that performs sensing, computation, communication and control tasks [1]. These parts are tightly coupled, i.e., embedded controllers and networks monitor and control physical processes, which, in turn, affect the computation and actuation.

Real-Time Location Systems (RTLS) are a class of networked embedded systems, which perform the tasks of localization and positioning of objects or people in real-time. Nowadays, RTLS become an integral part of the cyber-physical world because localization is necessary in the majority of CPS applications, such as smart homes, logistics, manufacturing processes control, assets tracking, smart agriculture, entertainment and so on.

Accordingly, RTLS face the same challenges as CPS and most of them are related to design. As for any other embedded system, the design process of RTLS requires a lot of interdisciplinary knowledge, time and money investment. The developed system is expected to completely meet its requirements (such as accuracy and precision of localization, real-time constraints, cost and so on) and moreover the design process should meet the time-to-market deadlines.

In particular, one of the most important design stages is called *design space exploration*. It is a process of analyzing different alternatives of system construction and configuration and selecting the optimal ones for a particular application according to the given requirements. The outcome of the process, i.e., the design decision, depends on the specific system aspect being designed and the level of abstraction (for example, it could be a circuit layout, a set of hardware components to be used, an algorithm, a configuration of the network protocol, etc.). In other words, the set of available components and solutions forms a "design space", which a system developer should "explore".

Real-time location systems may include different communication technologies (e.g., RFID, WiFi, wireless sensor networks, etc.) with various configuration parameters and protocols and different localization methods (e.g., range-based, proximity-based and so on), which form the high-level design space of these systems. In many applications it is very hard and at the same time crucial to make a proper choice among them to satisfy the requirements. Moreover, each technology can be configured differently in terms of hardware parameters, used protocols and so on. In many cases, this problem is partly avoided by designers due to its complexity and limited development time of the project. At the same time, incorrect design decisions made on early stages can result either in catastrophic consequences for system safety or in a significant loss of money for re-design and fixing or both. All aforementioned issues make the design space exploration a fundamental challenge for modern RTLS and, more generally, cyber-physical systems.

Our motivation for starting this research has the following key points. First, existing tools are not sufficient for RTLS in many respects. Moreover, in most cases developers rely only on prototyping and real-world deployments to evaluate their systems, and, therefore, explore design alternatives in a considerably limited manner. Second, many safety-critical applications for RTLS currently appear. Their design requires more exhaustive verification than prototype experiments could provide. Automated verification is currently not applied to RTLS design. Finally, the future vision of CPS (and RTLS as their part) is related to high autonomy, heterogeneity and self-organization. Therefore, such concepts as "design in runtime" [2] arise and encourage many researchers to develop efficient instrumentation for their support.

Our main goal in this research is to develop a methodology for RTLS design space exploration, which combines such important steps as modeling (simulation), requirements specification and verification into a single efficient process. First, we aim to provide an extensible and flexible simulation framework for RTLS with a particular emphasis on environment description, wireless channel representation and mobility components. As we discuss in this paper, an accurate combination of these aspects is missing in most existing simulators applicable for RTLS. Second, we are to provide a way of formally define design requirements and constraints for the systems under study. Finally, we plan to apply automated verification techniques both for checking simulation results and for exploring the space of system parameters. The former will increase the confidence that

the design, which goes to the prototyping phase, is correct. The latter is related to guiding the simulator by automatically adjusting system parameters and, therefore, evaluating different points in the design space. Using formal symbolic techniques will allow us to allocate regions of feasible parameter combinations, which could be explored. In other words, it is a significant contribution to optimizing the design. We consider this approach very promising and novel.

By applying such methodology to RTLS, designers will be more confident on their decisions, i.e., determining the best-suited technology and configuring it (e.g., selecting a particular node placement, protocol configuration, localization method, etc.). In our approach we aim to use an automated verification technique called statistical model checking (SMC) that will provide more guarantees than real-world deployments and prototyping. We propose to use an SMC tool in an unconventional way, i.e., to guide and adjust the simulator via a feedback loop.

Overview of RTLS

Real-time location systems are very versatile. Basically, two main classes of systems exist: for outdoor and indoor localization. The latter is generally more complex, because common solutions like GPS are practically impossible to use indoors. Presence of a large amount of obstacles like walls is an important factor as well, because it highly affects the signal propagation. Also, most applications of indoor localization are more strict in accuracy requirements. Both indoor and outdoor RTLS solutions are usually based on a single hardware technology and single localization method. However, their combinations are also being developed by researchers and, moreover, are already available on the market[1]. Below we discuss existing RTLS technologies, localization methods and overview their current market. Providing an exhaustive classification of these systems requires a lot of additional effort and space and lies out of scope of this paper. Instead, our primary concern is to demonstrate that the current design space of RTLS is vast in order to prove that our research problem is important. Therefore, we limit ourselves to the aforementioned viewpoints. We also describe the constituents of the RTLS design space and classify the primary requirements of these systems.

Localization methods. A lot of different methods currently exist, but all of them can be divided into two classes. The first one, called *range-based*, uses information received from the wireless channel to estimate the distance between communicating nodes. One of these nodes is called an anchor, which means that its coordinates are assumed to be known. Many solutions are based on using received signal strength indication (RSSI) to estimate the distance [3]. Another important group relies on measuring the time between sending and receiving the signal [4]. Methods, such as time-of-arrival (TOA) and time-difference-of-arrival (TDOA), belong to this group. Another powerful method is angle-of-arrival (AOA) [5], though it typically requires complex antenna hardware. A detailed survey of range-based methods can be found in [6]. Several performance improvements have been developed, for example, Least Squares [4] and Kalman filtering [7]. Distances provided by range-based methods are used in calculating moving object's coordinates, for example, by using trilateration [6].

Another class of methods does not use distances for object's localization and is called *range-free*. These methods use statistical data and some a-priori information about the localization domain. For example, the scene analysis method uses a technique called fingerprinting to collect features of a scene and then uses these fingerprints to estimate object's location by matching online measurements (e.g., RSSI or number of hops) to them. Such methods as Bayesian estimation and kNN [8] are based on fingerprinting. Also, the Gaussian process technique [9] has been recently proposed to address the RSS localization problem. One more range-free set of methods is called *proximity-based*. They provide symbolic relative location information. Such methods are used, for example, in RFID localization [10]. Range-free methods typically do not provide the object's position but only some symbolic information, which is referred to as "location". Exact coordinates, however, can also be provided, for example, by referring to the nearest anchor node with known position. A range-free system can tell, for example, that the object is in a particular part of the room, which is accurate enough for certain applications. Authors of [11] evaluate several range-free RSS-based techniques in an ambient assisted living application and conclude that they provide sufficient accuracy and relatively low power consumption at the same time.

In general, range-based localization techniques provide greater accuracy of localization for the price of more complex and expensive hardware components (e.g., enhanced transceivers). Their calibration might be a longer process as well. At the same time, range-free methods are usually easier to implement on existing platforms, as they do not require any specific hardware. However, the accuracy that they provide is typically low or moderate. Programming effort for increasing it (e.g., applying a complex filtering technique) could be considerably bigger. For a more detailed comparison we refer the interested reader to [6, 8].

Technologies. Several communication technologies have been created or adapted for localization. One could refer to [6] or [8] for a comprehensive survey. Currently wireless sensor networks (WSN) are very promising in this domain. WSN consist of small battery-powered devices having different sensors on board (e.g.,

---

[1] Zebra technologies [Electronic resource]. – Access mode: https://www.zebra.com/us/en/products/location-solutions.html, free access, in English. (last access – 20.05.2015)

temperature, humidity, light etc.) and communicating via wireless channel. Being an important part of CPS, sensor networks are starting to be used as an underlying infrastructure for RTLS. From a high level point of view, localization can be considered as a single sensor that monitors the object's location. Many existing localization methods are applied to WSN. Most of them are range-based. For example, the Cricket system [12] uses both radio and ultrasonic waves together with the TDOA method for efficient indoor localization. NanoLOC TRX from Nanotron Technologies [13] is a high performance transceiver with localization capabilities using an advanced range-based method called SDS-TWR (Symmetrical Double-Sided Two-Way Ranging).

Radio frequency identification (RFID) is another technology widely used for localization purposes. It plays a key role in pervasive networks and services [14]. Typical applications include locating goods in warehouses and supermarkets, personnel tracking, security systems and so on. Passive, semi-active and active RFID systems are available on the market [15]. They differ in price, communication range and performance. In comparison to other wireless technologies, like WSN or Wi-Fi, passive RFID tags provide much shorter communication range, which is compensated by the number of deployed tags (the price of a single tag can be as cheap as 0.1$). At the same time, active battery-powered RFID tags could cover a much bigger area, comparable to other aforementioned technologies. Many RFID-based RTLS systems are related to mobile robots localization and navigation. For example, authors of [16] propose to combine RSSI fingerprints provided by a WSN infrastructure with passive RFID tags to ensure efficient obstacle avoidance. Several localization algorithms are also overviewed in [17, 18].

Ultra-Wide Band (UWB) is a localization technology, which is based on short (less than 1 ns) pulses in a very wide spectrum [19]. Average pulse frequency is 5 GHz and so UWB signals do not interfere with other wireless infrastructure and provide highly accurate distance estimation. ScenSor DW1000 from Decawave[1], a novel UWB-based transceiver for sensor networks, is one example. In comparison to other technologies, UWB hardware is much more expensive.

Many localization systems exploit some existing Wireless Local Area Network (WLAN) infrastructure based on the IEEE 802.11 standard. WLAN provides relatively long communication range (50-100 m). Many existing vendors provide Wi-Fi-based solutions (examples are provided in the next subsection). Another solution is Bluetooth (IEEE 802.15.3 standard), which provides shorter range and lower bit-rate in comparison to WLAN but is more ubiquitous and autonomous. Bluetooth modules are embedded in most portable devices. Also the 4th version of the 802.15.3 standard (Bluetooth LowEnergy) is currently one of the most promising wireless technologies. Many systems based on WLAN and Bluetooth are described in [8]. The most prominent advantage of WLAN and Bluetooth-based localization systems is the possibility of utilizing the existing infrastructure: Wi-Fi hardware (e.g., routers) is nowadays deployed practically everywhere; similarly, Bluetooth modules are embedded in most modern communication devices (e.g., smartphones).

Global Positioning System (GPS) is well known and widely used in different applications. It is not an exaggeration to say that GPS is the dominant technology for outdoor localization (e.g., car navigation). Despite this fact, it has very poor performance in indoor environments due to the bad coverage of satellite signal. Several solutions for indoor localization exist like A-GPS [6, 8].

Several other approaches for real-time localization exist. They include Inertial Navigation Systems (INS), laser-ranging, acoustic ranging and mobile cameras [6]. In many systems they are used in combination with technologies discussed above.

Current market of RTLS. Several off-the-shelf solutions for RTLS are currently available on the market. Application domains for deploying these systems are mostly the same among all existing vendors. The first one covers the industrial domain, like automotive or aerospace. It includes warehouse monitoring, efficient logistics, tracking of assets, people and machinery, work-in-process (WIP) and so on. The second one is related to the medical sphere: tracking doctors, nurses and patients in a hospital as well as medical assets localization. Several systems are also used in retail.

An interesting aspect to note is that most solutions are based on one of the two main technologies, namely Wi-Fi and RFID, or on the combination of those. For example, Ekahau[2] and Zebra provide Wi-Fi based RTLS solutions for health-care, logistics, manufacturing and retail. One of the key benefits, as claimed by these companies, is the possibility of integrating readers or anchor nodes into the existing WLAN infrastructure without the need of additional wiring or communication media. For being able to do this, they have established a partnership with the top providers of WLAN equipment. Such companies as Identec Solutions[3] and Savi[4] use the

---

[1] Decawave [Electronic resource]. – Access mode: http://www.decawave.com, free access, in English. (last access 20.05.2015).
[2] Ekahau solutions [Electronic resource]. – Access mode: http://www.ekahau.com, free access, in English. (last access 20.05.2015).
[3] Identec solutions [Electronic resource]. – Access mode: http://www.identecsolutions.com, free access, in English. (last access 20.05.2015).
[4] Savi technology [Electronic resource]. – Access mode: http://www.savi.com, free access, in English. (last access – 20.05.2015).

active RFID technology (433 MHz) for real-time localization in the industrial domain, while Radianse[1] provides RFID solutions for the health care.

All aforementioned vendors claim that their solutions are suitable both for indoor and outdoor environments and provide an accuracy of several meters. Most of used protocols are proprietary. There are some companies, like Ubisense[2] and Zebra, that also utilize the UWB technology (on the base of a sensor network) to achieve greater accuracy.

However, so far only few companies, for instance Identec and Ekahau, include in their vision the future smart environment, which is able to perform various tasks (e.g., localization, different kinds of sensing, etc.) dynamically mapping them on an existing heterogeneous infrastructure that might include different technologies like Wi-Fi, WSN, RFID and others. Also only few of them are focused on safety-critical systems that require very high accuracy.

Design space and requirements. A critical choice to be made at the beginning of RTLS design is to select a communication technology (or a combination of those). Available alternatives form the high-level design space of RTLS. Exploring this space could be called "technology selection". In some cases, the localization method can also be selected at this stage, because some of them, like RSSI, can be utilized with different technologies. At this point, the designer decides to use, for example, a WSN with TDOA localization, a WiFi infrastructure with signal strength measurements, a combination of WSN and passive RFID tags with a fingerprinting method and so on.

A significant part of the design is also concerned with finding the right configuration for a selected technology that is to tune the hardware, to select a proper protocol and its parameters and so on. This stage could be called "technology optimization" and it typically follows the previous one though they could also be done simultaneously to a certain extent (e.g., when one needs to compare a WSN solution with a particular protocol and localization scheme to a WiFi solution with another protocol). Examples of the optimization part include selecting a MAC protocol (e.g., TDMA-based, contention-based, etc.) and its parameters like radio duty-cycle and backoff time, choosing a routing scheme (for instance, AODV [20], CTP [21] or static), deciding on node/tag placement, etc.

Overall we can conclude that design space exploration of RTLS is related not only to selecting a proper technology and localization technique, but also to optimally configuring them. It could be said that the design space has a layered structure with technologies at the top and protocols, hardware parameters and configurations at a lower level.

There are several specific functional requirements for RTLS. One is called *accuracy*, and it is typically expressed as a mean error between measured and real location (could be, for instance, 1m or 20cm). Another one is *precision*, which defines how consistently the system works, i.e., it is a distribution of distance error over many trials [8]. There is also a metric called *resolution*, which expresses the minimum difference between measurements (for example, several centimeters). Designers should take these metrics into account when they reason about system performance. A qualitative functional requirement is the *type of localization* that can be either symbolic or exact (e.g., xyz coordinates) or hybrid. Among others, one could name possibility of localizing multiple targets and scalability. For example, combined functional requirement for an RTLS could be "*The system should provide exact coordinate measurements with 10cm resolution and 90% precision*".

Non-functional requirements also play an important role and in fact they are similar to other embedded and cyber-physical systems: real-time constraints (average delays, worst case execution times and so on), power consumption, cost.

RTLS are being applied to several safety-critical domains like emergency systems and, therefore, some related requirements arise. In particular, real-time constraints are significant, because their violation might affect safety. In some systems it is crucial for an operator to receive successive location measurements without any time gaps, i.e., in real time. The respective requirement is called *continuous localization*. To meet it, the system should have acceptable networking and computational delays as well as the possibility of making location measurements at any time. The latter means the necessity of having certain distinct measurements (e.g., distances) at any time to allow location calculation. The possibility of the system to operate normally when some nodes go down, i.e., fault-tolerance, should also be considered. All aforementioned requirements ensure system safety and reliability.

State of the art in RTLS design

Moving objects localization is becoming necessary in various applications, from logistics and security to interactive entertainment, and hence in the nearest future real-time location systems tend to be deployed everywhere: homes, roads, offices, warehouses and so on. As stated above, RTLS become an integral part of

[1] Radianse: Intelligent real time location solutions [Electronic resource]. – Access mode: http://www.radianse.com/products_overview.html, free access, in English. (last access 20.05.2015).
[2] Ubisense group [Electronic resource]. – Access mode: http://www.ubisense.net/en, free access, in English. (last access 20.05.2015)

modern cyber-physical systems and thus they should follow their design process. So far, it is still a fundamental problem for CPS because many new challenges arise, such as system heterogeneity and verifying non-functional requirements [22].

The goal of a system developer is to provide a design, which performs all necessary functions correctly. Likewise, the system should meet its non-functional requirements. Comparing different solutions and properly configuring them, i.e., design space exploration, is, therefore, a very important step. We have already analyzed what the design space for RTLS includes and what requirements exist. In the following subsections we briefly review the existing system-level design methodologies, describe the current design flow and then focus on such approaches as modeling and verification, discuss their role and review several tools.

Methodologies. Several design methodologies have emerged over recent years. Among the most topical one could name *Platform-Based Design* (PBD) [23, 24] and *Contract-Based Design* (CBD) [25]. The main concept of PBD is related to defining several levels of abstraction, each of which hides unnecessary details of the underlying implementation, to limit the design space to a set of available components at each level. In CBD, a new form of system specification is introduced which defines both the properties of the system as well as its assumptions on the context, in which the system is used. These specifications take the name of *interfaces* or *contracts*. Several other methodologies also exist, for example aspect-oriented and actor-oriented [26]. Their review in the scope of wireless sensor networks design is given in [27].

Both PBD and CBD can and should be adapted to real-time location systems. However, a clear methodology for RTLS design is still to be derived from these fundamental concepts. One should be able to make a choice between available components as well as to define their assumptions and guarantees. In particular, it should be possible to select a proper technology to use as well as the most suitable localization method, i.e., to efficiently explore the design space.

Current design flow. At present moment, design of many RTLS is based solely on prototyping. Developers typically select a communication technology and several algorithms for location estimation. By performing series of real-world tests (e.g., by trying different movement patterns, or changing transmit power) they intend to calibrate the prototype and come up with an optimal configuration. The choice of technology and localization method, therefore, is done empirically at the very beginning of development and afterwards designers stick to that decision. If the latter is incorrect, it might entail the development of a new prototype. Designers do not have sufficient tools to try out different alternatives without violating time-to-market or running out of budget because prototyping might be both expensive and time-consuming. At the same time, real-world experiments are of crucial importance, because it is impossible to consider and predict all possible factors and events in a model. What we are trying to emphasize is that an intermediate stage between specification and prototyping is required, which would highly increase the design space exploration quality and designer's confidence on their decisions. Such stage could include building models of different system configurations, simulating them and analyzing results using some verification techniques. The latter could be used in a feedback loop and affect the design in the sense of adjusting several model parameters for further simulations.

Modeling. Currently, among existing modeling approaches, *simulation* is the most widely used for exploring and evaluating the design. Any simulation framework used in a design process must obey the following basic requirements. First, it must capture the necessary aspects of the designed system and surrounding environment so that the provided models are sound. In case of RTLS, a flexible tool for describing mobility (moving objects, their trajectories and other parameters) and environment (walls, obstacles, etc.) is required. As the object location is estimated using wireless media, an accurate channel model is also very essential as well as the transceiver model (supporting different types of modulation, encodings, error checking and so on) [28]. This is highly relevant for both range-based and range-free methods in localization systems, because certain measurements are made in both these classes (signal strength, timings and so on). At the same time, absolute accuracy is typically unreachable (for achieving it, real-world experiments are necessary) and, therefore, a simulator should be "accurate enough" to achieve its goals in the context it is used.

Availability of measuring energy consumption is necessary as well. Also, the tool must be able to provide simulation traces with all necessary data required for the verification process, i.e., the tracer component should be very flexible.

There are several simulators, which partly satisfy the aforementioned requirements. Castalia [29] and MiXiM [30] are focused mainly on wireless sensor networks. They are based on the OMNET++ framework. The main feature of Castalia is the accurate and highly customizable wireless channel and radio transceiver with signal-to-noise ratio (SNR) calculation, different types of modulation, collision models, etc. For achieving high accuracy in estimating signal attenuation, path loss maps are calculated. Moreover, temporal variation based on different mathematical models is considered. The implementation of the radio channel model in MiXiM significantly differs from Castalia but is also promised to be accurate and flexible. However, both tools lack support for energy-aware aspects and a GUI. Also, only little documentation is provided for MiXiM.

Ptolemy II [31] and its extension VisualSense [32] provide an actor-oriented framework for modeling heterogeneous systems. Their distinguishing feature is the possibility of creating hierarchical models using different

models of computation at different levels. This allows more natural investigation of various system parts. A drawback of these tools is that they do not provide dedicated primitives for careful environment description, mobility and node placement. With existing components these tasks are doable but fairly complex and unnatural.

Another powerful tool is PASES [33]. It is an energy-aware design space exploration framework focused on wireless sensor networks. However, its extensible architecture allows using other technologies in simulations. Low-level hardware models are written in SystemC and so it is possible to add custom components, which can form another technology. The implementation is based on the TLM (Transaction Level Modeling) standard [34]. The main feature of PASES is a set of accurate battery models. Their dynamic lifetime estimation is provided as one of the simulation results and can be conveniently visualized using a rich GUI. Some basic primitives for environment description and mobility are provided in PASES, but currently they are not sufficient for effective RTLS modeling. Another drawback is a very simple wireless channel model, which is based on the SCNSL library [35] and supports only basic collision mechanisms.

Cooja [36] is a simulator for the Contiki operating system. It has a well-organized radio model, which supports signal reflections from walls, collisions and SNR calculation. Cooja can be used as a simulator for Contiki code or as a cycle-accurate emulator for MSP430 architecture (using MSPSiM tool). However, this very limited choice of hardware platforms makes Cooja insufficient for RTLS design space exploration, where different technologies and hardware should be evaluated.

Finally, IDEA1 [37] and ATLeS-SN [38] are another two extensible simulators for sensor networks, which are based on the SystemC core. Networking in IDEA1 is based on the same version of the SCNSL library as in PASES and thus does not provide sufficient accuracy for RTLS simulation. Developers of ATLeS-SN have adapted the channel and radio models from Castalia by wrapping them into SystemC modules and providing the necessary infrastructure. Therefore they achieved much greater accuracy in this aspect.

Verification. Another important method used in many design routines is called verification. It aims to ensure the correctness of the model as well as of simulation results and, therefore, helps to answer the question "Are we doing the thing right?" In particular, automated model verification techniques are used for checking simulation traces and also the model itself against the requirements. As we already discussed, design space exploration for RTLS is currently done only with empirical analysis and prototyping, i.e., simulations are rarely involved and, consequently, verification is not applied. Below we review several verification techniques and methods, which we find very promising for RTLS.

A well-known approach is *model checking*. It is a process of automated verification of models against the requirements, which are expressed in some formal language. It examines all possible system scenarios in a systematic manner [39]. Temporal Boolean logic is typically used for requirements specification, though a combination of theories is also possible, for example, difference logic and linear arithmetic over rationals. For theory combinations, *Satisfiability Modulo Theories* (SMT) solvers are used [40]. As output, model checking tools provide information whether requirement formulas are satisfied or not. In the latter case, many checkers are able to return a counterexample, which is very useful for debugging and analysis. Infinite executions and traces are typical for model checking, but a bounded approach also exists [41].

RTLS are very nondeterministic. This means that due to the probabilistic nature of object's movement as well as of other environmental conditions the amount of possible simulation traces is huge even for a small number of steps. This is a general problem of model checking and it is called the *state space explosion*. With increasing amount of traces the state space grows rapidly and its verification requires enormous computational resources. One of the possible ways of avoiding state space explosion is *Statistical Model Checking* (SMC). Instead of exhaustive verification of all simulation traces, it analyzes a certain amount of them (estimated or predefined) and uses statistical methods for deciding whether requirements are satisfied or not. The verification task is therefore reduced to running a certain number of simulations and checking their traces, which are deterministic. As output, SMC tools provide True (satisfied) or False (not satisfied) and the probability of this decision [42].

Several SMC tools currently exist. One of such tools aimed at verification of probabilistic real-time systems is PRISM [43]. It provides a discrete-event simulation engine and a model description language called RML (Reactive Modeling Language). System properties can be described in several temporal logics.

PLASMA-lab [44] is a high-level SMC platform that allows the usage of custom model checking tools and simulation frameworks. Basically, they can be integrated into the platform as plugins. PLASMA-lab is able to calculate the number of simulations to run based on Monte-Carlo method or several other approaches and then trigger the simulation tool. Then it provides output simulation traces to the model checker. Several simulation engines and languages (e.g., RML) have been already integrated to PLASMA. In other words, PLASMA implements different SMC algorithms and acts as a controller of simulation and verification.

Application examples of RTLS

Below we provide several application groups and examples, which have different primary requirements, and, therefore, the main design criteria for them will be different as well. We plan to use these applications as references during our research and expect our methodology to be applicable to all of them.

A particular example is related to the industrial domain, which is one of the primaries for existing RTLS vendors (e.g., Identec, Savi, Ekahau, Ubisense). The purpose of the system is to localize and track goods, machinery or people in a huge warehouse or in docks. Such systems typically require average accuracy (several meters could be enough). But due to their large-scale size, cost and power consumption are the primary metrics to be optimized, because using expensive hardware for a large network as well as performing frequent battery replacement routine is very costly. Design space for such class of systems typically includes WLAN, active/passive RFID and WSN. Suitable localization methods are usually proximity-based (fingerprinting, nearest neighbors, etc.).

Another class of applications is safety-critical. A good example is an emergency system that guides people to an evacuation point or helps the rescue team get to the source of fire or other hazardous places [45]. The aforementioned safety requirements and real-time constraints play the most important role: the system should provide continuous localization without any gaps or delays that might violate the rescue mission. Emergency mode of system operation is typically expected to last several minutes or hours, and hence power consumption optimization is not the primary task during design though still very important. While many such systems (like the ones deployed on an oil rig, for example) are mission-specific, others switch to emergency mode on demand, performing tasks like ambient monitoring the rest of the time. Many technologies might be involved: WSN, WiFi, RFID, Bluetooth and so on. An important emerging domain of CPS and RTLS, which has many safety-critical requirements for localization applications, is Vehicular Ad Hoc Networks (VANET) [46]. Applications requiring high-accuracy and real-time localization in VANET include vehicle collision warning, cooperative cruise control and driving, blind crossing and some others [47].

One more example is precision agriculture. Such systems are used for accurate cultivation and fertilizing. For instance, there could be a robotic vehicle in a greenhouse which handles the aforementioned operations for different plants based on a particular schedule or other parameters like humidity, light and so on. This could be done automatically without human interaction but location of the robot is required in relation to plants so it could do these operations accurately without harming others. Similar systems are also used outdoors both for helping a human control some machinery on the field and for complete automation. Typically GPS is used outdoors, but solutions with another supportive technology also exist [48]. Such systems are expected to be optimized for accuracy, which might be several centimeters for indoor and up to one meter for outdoor. Other systems, which primarily require accuracy/precision optimization, are manufacturing (e.g., localizing machinery and particular manipulators during a car construction process), rescue (remote control of a robot moving in a hazardous area), medical (precisely positioning robotic tools during a surgery) and so on. Such systems might require a very accurate but expensive technology like UWB, or a highly optimized range-based solution (e.g., RSSI, TOA) for WSN or also some combined approach.

Problem statement

Having many different technologies and methods available, designers have to make a careful choice among them to meet the requirements and find optimal designs. In other words, design space exploration for RTLS becomes very challenging. Incorrect design decisions as well as the lack of verification may lead to huge time losses for bug fixing, implementation of another method, re-testing or even moving to another technology. Most likely this will result in failure of meeting time-to-market constraints, which, in turn, entails significant money losses. The situation might get even worse if the incorrect decision violates the system safety.

To the best of our knowledge, a clear methodology for design space exploration of RTLS is still missing. As discussed above, capabilities provided by existing simulation tools are not sufficient for RTLS in different aspects: mobility, environment description, accurate wireless channel and radio modeling, GUI support, ease of use and so on. Efficient methods of building models of RTLS and simulating them are, therefore, still to be defined. Other important unsolved tasks include formal specification of requirements and using automated verification techniques for checking and optimization. A design space exploration methodology should introduce strong guidelines on how to solve the aforementioned tasks. Such methodology as well as a powerful tool that supports it will make a considerable impact on RTLS design process.

Another important reason for using modeling and verification alongside rapid prototyping and deployment comes from safety-critical domains. Avionic and automotive systems, for example, must guarantee safety and therefore cannot rely only on extensive prototype testing. Additional certification is required. This certification (several standards currently exist in different domains) proves that safety-critical properties are not violated under any possible conditions. To verify a system in such a way, real-world tests are very important but definitely not enough. This is a task for automated verification techniques.

Methodology overview

So far in this paper we have outlined the most challenging problems, which a design space exploration methodology for RTLS should overcome. In this section we will propose our hints on their solution. Our ultimate goal is to introduce a methodology that not only consists of these solutions but also integrates them into

a single and efficient process. In this section we also describe Povomon, a deployed wireless sensor network for low-power indoor monitoring, and introduce a case study for this system, based on which we plan to validate our methodology.

As already mentioned, there are several significant tasks in the design space exploration process, which our methodology is going to solve. First, a designer should create an adequate model of the system using an appropriate tool. Second, a formal specification of requirements should be done, perhaps in parallel with model construction. Finally, he/she needs to run simulations and analyze the provided traces to check whether the requirements are met or not. Based on the analysis he/she can proceed further with the design, for example, lower the level of abstraction and refine the model or change some of its parameters to try a new configuration. Regarding the latter, another motivation to introduce verification in the design flow is related to guiding the simulator configuration, i.e., requesting additional simulation runs (particularly with updated parameters) to increase the confidence level of verification results and optimize the design. We provide more details further on.

The envisioned design space exploration methodology for RTLS is briefly summarized on Fig. 1. There are three stages: simulation, requirements specification and verification/optimization. By integrating them into a single process means, first of all, the compatibility of supporting tools is meant. Another advantage in this sense is the possibility of introducing a high degree of automation for the process by having several feedback transitions (run more simulations, update/change the model and requirements).

Modeling and simulation. Our first aim henceforth it to provide an extensible and flexible framework for simulation of RTLS. Implementation of a new standalone simulation tool is an extremely difficult task but, as we discussed previously, some existing solutions already provide many useful capabilities. Therefore, we will focus on extending an existing framework with important aspects of modeling RTLS.

We have already discussed the necessary capabilities for an RTLS simulator, namely the mobility, environment description tools, accurate wireless channel and radio models. A suitable tool for our purposes should either meet these criteria or provide convenient extension possibilities for integrating these components. Accurate battery lifetime estimation and other energy-aware aspects are also favorable. Moreover, the simulator should have acceptable performance and provide sufficient GUI support.

After analyzing several simulators we concluded that PASES [33] is the most suitable for our task. More specifically, it currently encompasses the full design space exploration solution for wireless sensor networks with particular emphasis on energy-aware aspects. Hardware components in this framework (e.g., MCUs, radio transceivers, sensors) are provided as separate SystemC modules and the complete platform is assembled from these parts. PASES provides an extensible and highly customizable environment and so it is possible to supplement it with other hardware models like RFID tags or Bluetooth transceivers. Another advantage of SystemC and the TLM standard in particular is the high simulation speed as many communication aspects are abstracted away. At the same time, it is always possible to refine certain parts of the hardware model to study them more carefully (e.g., go down to register transfer level). PASES can provide very accurate battery lifetime estimation, because it tracks the power consumption of each component of the platform and even takes battery microstates into account. PASES also has several GUI tools to view and analyze simulation results.
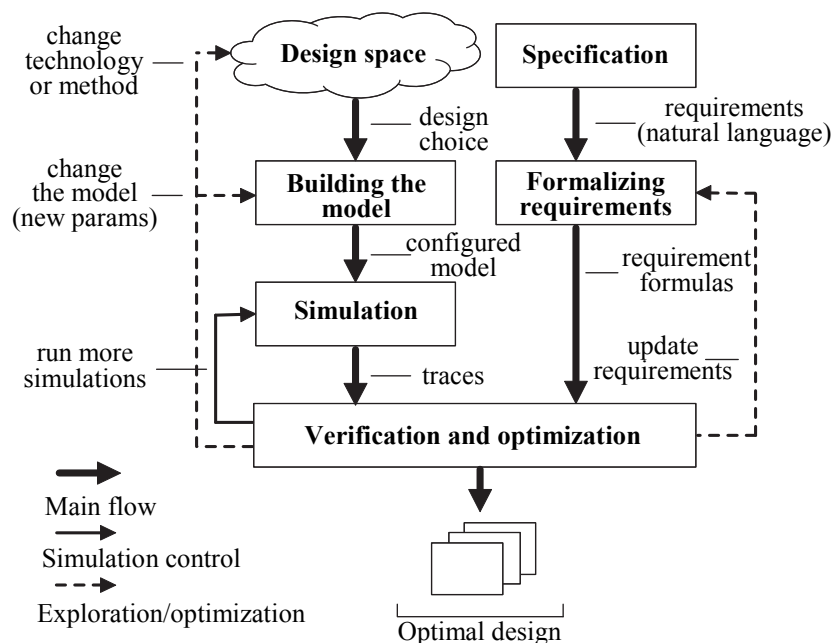

Fig. 1. Workflow of the design space exploration methodology for RTLS

The current channel and radio models of PASES are insufficient for RTLS. Its radio component is implemented in SystemC and provides a very basic API for application developers (limited to send/receive commands, state switching, calculating received signal strength, simple clear channel assessment and some others). Features like SNR calculation, changing modulation type or encoding are missing. Also the wireless channel model, which is based on SystemC Network Simulation Library (SCNSL) [35] is very simplistic. It provides a simple collision mechanism and signal attenuation calculation. SCNSL was extended in PASES with several methods supporting node mobility. Nevertheless, existing components are not accurate enough, because most localization solutions highly depend on the wireless channel properties. Lack of detalization in discussed components entails building inadequate models, which in turn will provide unreliable simulation results.

However, during our analysis we have found a way of overcoming these drawbacks of PASES. In brief, it is required to improve the radio transceiver model (an internal component of PASES) and extend its API as well as to switch to another wireless channel implementation (SCNSL is an external component and so it can be changed to something else). In order to comply with the concept and architecture of PASES, improved and new components should be written in SystemC. As discussed previously, in the ATLeS-SN simulator [38] channel and radio transceiver methods from Castalia have been wrapped into the SystemC TLM infrastructure. The source code for this tool is publicly available. We believe that integration of radio components from ATLeS-SN and Castalia into PASES will result in highly accurate simulations in the aspect that we require for RTLS.

Another important problem is the improvement of environment description and mobility components of PASES. Regarding the first, we should clarify what we consider relevant for RTLS and what we plan to implement in this research. The most important feature that we need is to be able to describe the space (e.g., a room) by putting walls and obstacles and also to handle the node placement. It should also be possible to set signal attenuation parameters for obstacles based on their size and material. Of course, related logic of PASES will require improvements. In particular, we will need to implement an algorithm that for each communication between two nodes will check the obstacles in proximity, determine those which cross the path of transmitted signal and affect it, and re-calculate signal parameters accordingly. Node mobility is also to be improved with different movement patterns and probabilistic scenarios.

One more task is the modeling of hardware components different from wireless sensor networks, i.e., RFID, Bluetooth, Wi-Fi and so on. For achieving this, their hardware representations in SystemC will be required as well as implementations of common protocols and possible physical layer improvements. Finally, we plan to review the tracer mechanism of PASES and ensure that provided simulation traces include the state of all necessary variables for the subsequent verification process. It should be configurable enough to return different properties of interest for different nodes.

We intend to provide a version of PASES that is capable to simulate RTLS. Wireless channel and transceiver models will be improved as described above by integrating with components from Castalia. Despite the fact that WSN, RFID, WiFi and so on obviously have diverse channel and radio models, the key thing is to provide a flexible infrastructure to be able to configure different models later on (in particular by using different mathematical models, modulations, frequencies and so on). The same is related to hardware models of different technologies. For validation of the methodology that we discuss later in the proposal we will primarily require WSN models (which are already present in PASES but will need certain improvements) and RFID hardware models that we intend to implement. So the design space of our initial version will consist of two technologies, but, as we discussed previously, there are various configuration alternatives based on them in terms of different localization methods, protocols and parameters. We will also implement the aforementioned environment description features that we find most relevant for an RTLS simulator. Our main contribution, therefore, is related to providing a simulation infrastructure that is independent of the technology being used and allows extensibility.

An important remark to be made is related to the simulation accuracy. One could argue that it is critical to have absolutely accurate results in our methodology. However, this is not the case. Instead, we require the ability of distinguishing solutions and having bounds and confidence levels, i.e., to broaden the design space exploration. Certain features, therefore, which are related to environment modeling, we leave out of scope (e.g., multipath effects, or temperature and humidity distributions) because these should be observed during real-world tests. Also, running experiments having another possibly interfering wireless infrastructure is important, but we plan to rely on prototyping in this aspect. This is justified by the fact that trying to account for all possible factors in a model could be incredibly complex, resource and time-consuming, i.e., in some cases prototyping could give results much faster. We do not aim to replace real-world experiments with simulations. Instead we plan to introduce our methodology as an intermediate stage that could efficiently extend the design space exploration as well as increase the confidence level on selecting a particular technology and configuration. We will extend the capabilities of PASES in order to achieve acceptable accuracy for acquiring conservative verification results. For instance, several aforementioned factors have a purely probabilistic nature and, therefore, introducing some randomization would considerably improve our results. As an example, the temporal variation of the wireless signal introduced in the Castalia channel model (we have already mentioned it above) allows the small-scale fading effects (e.g., multipath) to be taken into account by applying specific random distributions like, for

instance, Rayleigh fading [49]. It would also be possible to consider certain other probabilistic environment factors by supplementing the calculations with random variables. As a result, with our methodology we will be able, in particular, to evaluate the design with worst/best values of some parameters (e.g., temperature conditions or noise level) and, therefore, get bounds on the parameter space, which will drastically reduce regions of feasible system configuration alternatives. Also, using the probability distributions will give the possibility to evaluate the system behavior between these bounds and, in particular, tighten them. We provide more details on verification and optimization below.

Requirements specification. Our second task is to provide a clear specification of RTLS properties in a formal language so they could be used in automated verification. A prospective language of expressing them is temporal logic [39], because we aim to check their satisfiability on each simulation step. In particular, bounded linear temporal logic (BLTL)[1] could be used, because our simulation traces are going to be finite. In BLTL, one introduces a bound (in simulation steps) up to which a property should be checked. This is done because in the base case the "G" ("globally") quantifier of linear temporal logic entails that a property is satisfied on an infinite trace.

For instance, a continuous localization property could be expressed in natural language like the following: "*The moving object will always stay in connection with at least 3 anchor nodes*". The BLTL version is:

$$G^{\leq L}(nConnected \geq 3)$$

where *nConnected* is the number of anchor nodes, which are currently on the line of sight with the mobile object and thus could provide reliable distance calculation, L is the trace length, $G^{\leq L}$ means "globally within the simulation trace".

As a more complex example, let us write down the requirements formula for a mobile node in an emergency localization system (it could be a PDA of a fireman, a robot etc.). We will use a weaker version of continuous localization constraint: "*If a moving object is connected to less than 3 anchors, it will be connected to some third anchor no later than in 10 seconds*". We will also assume that a range-based method is used for localization and use the absolute distance measurement error as an accuracy constraint. Finally, we will put a constraint on the average power consumption of the node. The final BLTL formula is:

$$G^{\leq L}\left[\begin{array}{l}\left((nConnected < 3) \rightarrow F^{\leq 10}\left(nConnected \geq 3\right)\right) \wedge \left(\left|d_1 - d_{1real}\right| \leq 0,8\right) \\ \wedge \left(\left|d_2 - d_{2real}\right| \leq 0,8\right) \wedge \left(\left|d_3 - d_{3real}\right| \leq 0,8\right) \wedge \left(avgPower \leq 250\right)\end{array}\right]$$

where d_1, d_2 and d_3 are the measured distances to three anchor nodes, d_{1real}, d_{2real} and d_{3real} are real distances (all expressed in meters), *avgPower* is the average power consumption of the mobile node (in mW), $F^{\leq 10}$ means "finally within 10". The latter as well as the trace length can have units of time (e.g., seconds) or they can be expressed as a number of simulation steps. Accuracy constraint here is expressed as a simple difference between acquired and real distances between nodes. Of course, this formula is not complete. We did not specify such properties as transmission delay, packet delivery ratio (PDR) as well as others for the sake of space. It could be done in a similar manner.

It is worth mentioning that most variables in our expected simulation traces will have a non-Boolean value, but are either real or integer. This entails that atoms in our BLTL formulas will be expressed in some underlying theory, e.g., difference logic or linear arithmetic. Consequently, for deciding the satisfiability of requirement formulas we will probably need an SMT-solver. We discuss this in the next subsection.

Verification in design. The most significant step in our methodology is to introduce verification in the design flow of RTLS. To check a single trace one needs to implement a program that takes the trace and requirement formulas as an input and provides the verification result as an output. A custom script could possibly be enough to solve this, though this requires additional investigation and complexity analysis. Another alternative is SMT. Examples of requirements provided in the previous subsection are expressed in the theory of rationals. These formulas together with the trace, that would play the role of a truth assignment, will form the SMT-problem. It would then be possible to use an SMT-solver for verification and the main task will be related to implementing a proper encoder into SMT.

However, verifying a deterministic trace does not make a sufficient contribution per se. First of all, we need to ensure that not just a single trace but the whole system satisfies some property. It means that we should be able to verify all possible traces. However, this ideal case might result in state space explosion and, as we already discussed, the statistical model checking (SMC) approach is very promising to avoid it. SMC is capable to reduce the initial problem to verification of a finite amount of deterministic simulation traces. This means the possibility of a verification tool to guide the simulator and entails having a feedback loop. The statistical approach involves the estimation of simulations required to reach the desired confidence level. After checking a trace, the tool should add the result to its statistical data, decide on whether it requires additional simulations or not and in the first case

---

[1] Bounded Linear Temporal Logic – PLASMA lab. [Electronic resource]. – Access mode: https://project.inria.fr/plasma-lab/documentation/languages/property-language/, free access, in English. (last access - 20.05.2015)

trigger the simulator. Moreover, by applying some specific control logic it could adjust certain model parameters at the same time. In other words, the SMC tool, relying on current verification results, could try/suggest different parameter values or system configurations, i.e., explore different points in the parameter space. Therefore, we propose to apply verification technique not only for trace checking but also for optimizing the design.

Another, more formal technique, which we aim to apply, is performing parameter space exploration with SMT. In contrast with using the SMC platform for adjusting parameters, here the exploration process can be much more exhaustive. In brief, the workflow is the following. First, error traces (i.e., those which do not satisfy the constraints) are obtained from simulations. Second, each trace is expanded symbolically and projected onto the parameter space in order to find the region of unfeasible parameters (i.e., those combinations, with which it might also be possible to follow the same trace and reach the error state). Such regions are excluded from the parameter space. As a result, one could get a region of feasible parameter combinations, that is, symbolically expand a single point to a region instead of successively exploring each point one by one. Moreover, such approach could drastically decrease the amount of required simulations for exploring different parameter alternatives. Also this is another argument that advocates for applying an SMT-solver as a checker in our methodology. More background can be found in [50] and [51].

There are three main contributions that we will make by applying verification to design space exploration of RTLS: checking simulation traces, controlling the simulation and performing parameter space exploration. Regarding the latter, we will first experiment with the SMC tool as a controller of parameters variation, and as a second step apply the technique that combines simulations and symbolic approach with SMT. Our goal, therefore, is to find a proper verification instrument and integrate it to our toolchain.

As a result of our investigation, we decided to use the PLASMA-lab platform [44]. Its biggest advantage is the possibility of integrating different simulation frameworks with different model checkers and explicitly controlling the verification process. Therefore, we need to provide a rich simulation control API for our extended PASES framework and make it compatible with PLASMA interfaces. The latter means that we will have to wrap the simulator into a plugin for PLASMA. Its authors encourage using Java Simple Plugin Framework (JSPF) for this purpose[1].

The same must be ensured for the checker. We have already discussed anticipated alternatives above: a custom script that checks traces and a combination of SMT-solver and encoder. The first one is simpler to implement and use for trace checking, but the second one is more promising in other aspects of verification that we aim to introduce.

We should also note that "verification in design", i.e., having a feedback from the verification tool to the simulator, is an unconventional way of using an SMC platform. We strongly believe that it is novel and fully applicable to RTLS. The same could be claimed for using SMT for finding regions of feasible parameters in the scope of RTLS. These approaches could highly increase the degree of process automation, breadth of design space exploration and the confidence on correctness of system configuration. In other words, our proposal is related to using verification not only to check, but also to design.

Resume and final remarks. The expected result of our research is summarized on Fig. 2. It is a combination of a simulation framework for modeling RTLS and a statistical model checking platform. We are to provide the simulation part by extending the PASES framework, devise a requirements specification formalism, select a checker for simulation traces and ensure the interface compatibility of these parts with the PLASMA platform. In other words, we are to provide two plugins for the SMC platform and use them together. PLASMA will return the satisfiability of the model together with its probability value (e.g., "*The property is satisfied with a probability of 95 percent*"). This will be the primary experimental result for designers. They will be able to use different technologies and localization methods in their RTLS models, run simulations and extensively verify traces of many successive runs against the clearly specified requirements. Moreover, the toolchain will give the possibility for automatically adjusting system parameters and, therefore, evaluating many different configurations. Hence, it will be much easier to see if the requirements are still satisfied or not while developers explore the design space and so they will be more confident on their decisions.

The toolchain that we will provide could be viewed as an integration of existing instruments. However, there are two important contributions to the state of the art that should be emphasized. First, these instruments require sustainable improvements. In particular, the improved PASES framework will gather all necessary features for RTLS design space exploration. We have analyzed them previously. At present moment all existing tools are missing certain aspects and a complete solution is still absent. While some of these aspects we will implement during the research, we will also provide infrastructure for the others. Our second contribution is related to introducing automated verification in design. We will use it to check simulation traces and also to control and configure the simulation by having a feedback loop. The latter is an unconventional way of using a verification tool. Moreover, we aim to introduce a more formal method of exploring the parameter space, that is SMT.

---

[1] PLASMA Lab book [Electronic resource]. – Access mode: http://sparika.gitbooks.io/plasma-lab-book, free access, in English. (last access - 20.05.2015)
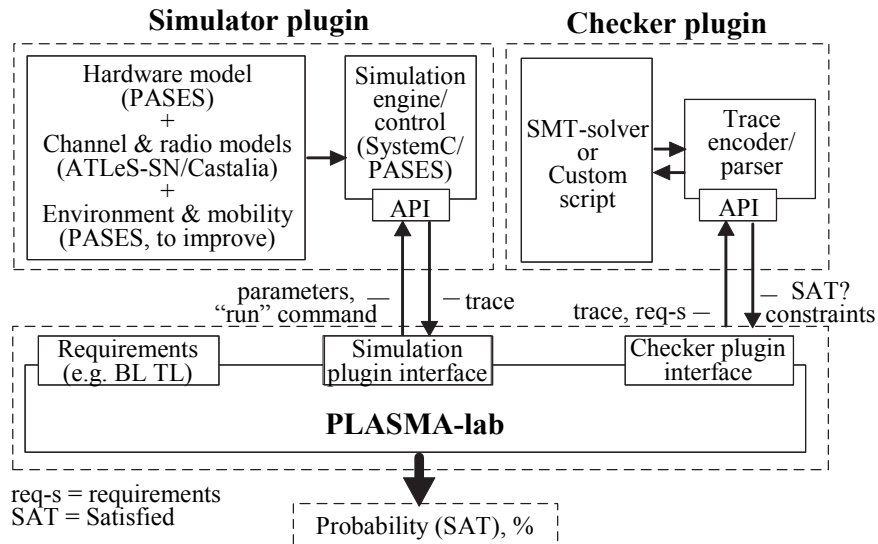
Fig. 2. Expected toolchain for RTLS design space exploration

We strongly believe that it would be possible to use our methodology for designing other aspects of cyber-physical systems, not necessarily related to localization. The approach has a generic background that is of simulating system models, obtaining deterministic traces and verifying properties and constraints by applying statistical model checking.

Case study

Having described the development plan of the methodology we also require a case study for its validation. We propose to use our approach and toolchain in the scope of the Povomon project [52]. It is a wireless sensor network of 29 nodes, which was deployed on the floor of the ICT doctoral school in February 2014. Its current main task is low-power monitoring of various ambient conditions like temperature, light, humidity, etc. The ultimate goal of the system is the optimization of power consumption in the building by controlling HVAC, lighting and so on. Therefore, Povomon falls into the "smart buildings" application, which is one of the current primary research directions in CPS [53].

The communication protocol of Povomon is based on the IEEE 802.15.4 physical and MAC standards, though several of its parts (e.g., time synchronization) are proprietary. Currently, all nodes run a TDMA-based schedule. They wake up every 5 minutes, transmit sensor readings and switch the radio off going back to low-power mode. This definitely does not correlate with localization applications, where anchor nodes should continuously communicate with moving objects in order to provide frequent location updates. On the other hand, having the radio always on entails huge power consumption and, in turn, very short battery lifetime. Consequently, designers should very carefully select the nodes that will participate in the localization (i.e., task mapping), their placement, duty cycles and so on to satisfy power consumption constraints.

In our case study, we propose to design and evaluate another mode of operation for the Povomon system, i.e., the emergency mode that we described previously. It presumes that in case of fire or substantial smoke emission the network switches from ambient monitoring to another task - localization. Its purpose will be to guide firemen or some automatic machinery in low visibility conditions. This could be of great help especially if the floor plan is complex (e.g., with many rooms, parallel corridors and so on). The system could guide people to the nearest evacuation point (fire door, staircase) as well. Such emergency mode together with the current Povomon application will be an efficient and extensible prototype for a smart building infrastructure.

The resulting system should satisfy the following requirements. It should continuously provide 2D coordinates of a moving object with an accuracy of at least 1 meter. Maximum delay of localization is 3 seconds. Power consumption of the system should be minimized, i.e., several parameters like transmit power should be properly optimized.

By applying our methodology we expect to optimize many parameters of the current deployment in the scope of the described use case. Our design space for "technology selection" will consist of WSN and RFID (active and passive). We therefore plan to evaluate how the system will operate if based only on WSN and also if complemented by an RFID infrastructure. There are several range-based and range-free localization methods that we want to try: RSSI, TOA, fingerprinting with RSS, proximity-based technique with RFID tags. To ensure reliable communication we should also select certain protocols. For MAC we will try contention-based (e.g., S-MAC [54] or simple CSMA) and scheduled-based (TDMA) alternatives. For routing classical solutions like CTP [21] and static scheme will be evaluated. It is also interesting to evaluate specific routing solutions for networks with mobile

nodes, for example MobiQuery [55], which detects the nodes that do not participate in localization at current moment and puts them to sleep. This is a very promising energy-aware solution for localization systems.

In order to achieve acceptable performance we are going to do parameter space exploration with our provided tools. List of parameters includes node placement, radio duty-cycle, transmit power, frequency, backoff time and some others. In particular, we should optimize the placement of nodes in order to ensure continuous localization as in examples that we have shown above. Of course, simulations will only allow us to restrict possible locations of nodes to certain areas because many environmental factors influence the connectivity. The same applies to several other parameters. However, as discussed previously, the possibility to shrink the space of possible alternatives and to distinguish them is a significant benefit per se. After using our toolchain we will perform real-world tests on configurations that we select and do further calibrations if required.

Conclusion

In this paper we presented our vision of the design space exploration methodology for real-time location systems. We proposed to extend an existing simulation tool with additional components representing the hardware used in RTLS. Likewise, our simulator will provide accurate wireless channel and radio transceiver models and improved components for environment description and mobility. We also introduced the way of formalizing requirements of RTLS. Finally, we proposed to integrate automated verification into design, not only to check simulation traces, but also to control the simulation process, adjust its configuration and optimize system parameters.

The proposed methodology together with provided instrumentation will contribute to the design space exploration process, which is one of the most challenging parts of modern cyber-physical systems design, RTLS in particular. The level of confidence on design decisions (selecting localization methods, wireless technologies and their configurations) will increase and, therefore, related risks will be lowered. The methodology will be validated by designing a typical application for RTLS, that is, an emergency system, but will be applicable also for other classes of applications, which have different primary design criteria (e.g., power consumption or localization accuracy), because our simulator will provide an accurate radio channel model as well as the power consumption model. We also expect our approach to be extensible enough to address design space exploration problems in a more generic sense, i.e., for other classes of cyber-physical systems.

References

1. Lee E.A., Seshia S.A. *Introduction to Embedded Systems: a Cyber-Physical Systems Approach*. Lulu.com, 2011, 516 p.
2. Lee E.A., Kubiatowicz J.D., Rabaey J.M., Seshia S.A. et al. The Terraswarm Research Center (TSRC) (A White Paper). *Technical Report no. UCB/EECS-2012-207*. EECS Department, University of California, Berkeley, 2012, 14 p.
3. Lee J.H., Buehrer R.M. Fundamentals of received signal strength-based position location. In *Handbook of Position Location: Theory, Practice, and Advances*. John Wiley and Sons, 2012, pp. 359–394. doi: 10.1002/9781118104750.ch11
4. Buehrer R.M., Venkatesh S. Fundamentals of time-of-arrival-based position location. In *Handbook of Position Location: Theory, Practice, and Advances*. John Wiley and Sons, 2012, pp. 175–212. doi: 10.1002/9781118104750.ch6
5. Peng R., Sichitiu M.L. Angle of arrival localization for wireless sensor networks. *Proc. 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*. Reston, USA, 2006, vol. 1, pp. 374–382. doi: 10.1109/SAHCN.2006.288442
6. Zekavat S.A., Kansal S., Levesque A.H. Wireless positioning systems: operation, application and comparison. In *Handbook of Position Location: Theory, Practice and Advances*. John Wiley and Sons, 2012, pp. 3–23. doi: 10.1002/9781118104750.ch1
7. El Madani B., Yao A.P., Lyhyaoui A. Combining Kalman filtering with ZigBee protocol to improve localization in wireless sensor network. *ISRN Sensor Networks*, 2013, art. 252056. doi: 10.1155/2013/252056
8. Liu H., Darabi H., Banerjee P., Liu J. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, 2007, vol. 37, no. 6, pp. 1067–1080. doi: 10.1109/TSMCC.2007.905750
9. Aravecchia M., Messelodi S. Gaussian process for RSS-based localization. *Proc. 10th IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communications*. Larnaca, Cyprus, 2014, pp. 654–659. doi: 10.1109/WiMOB.2014.6962240
10. Bouet M., dos Santos A.L. RFID tags: positioning principles and localization techniques. *Proc. 1st IFIP Wireless Days*. Dubai, United Arab Emirates, 2008, art. 4812905. doi: 10.1109/WD.2008.4812905
11. Raza U., Murphy A.L., Picco G.P. Embracing localization inaccuracy: a case study. *Proc. IEEE 8th Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing*. Melbourn, Australia, 2013, vol. 1, pp. 207–212. doi: 10.1109/ISSNIP.2013.6529790

12. Nissanka B.P. *The Cricket Indoor Location System. PhD thesis*. Massachusetts Institute of Technology, 2005.
13. *Real-Time Location Systems (RTLS). Technical Report*. Nanotron Technologies GmbH, 2006, 20 p.
14. Want R. An introduction to RFID technology. *IEEE Pervasive Computing*, 2006, vol. 5, no. 1, pp. 25–33. doi: 10.1109/MPRV.2006.2
15. Mitrokotsa A., Douligeris C. Integrated RFID and sensor networks: architectures and applications. In *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations*. CRC Press, 2009, pp. 511–536.
16. Enriquez G., Park S., Hashimoto S. Wireless sensor network and RFID fusion approach for mobile robot navigation. *ISRN Sensor Networks*, 2013, art. 157409. doi: 10.1155/2013/157409
17. Liu H., Bolic M., Nayak A., Stojmenovic I. Taxonomy and challenges of the integration of RFID and wireless sensor networks. *IEEE Network*, 2008, vol. 22, no. 6, pp. 26–32. doi: 10.1109/MNET.2008.4694171
18. Lei Z., Zhi W. Integration of RFID into wireless sensor networks: architectures, opportunities and challenging problems. *Proc. 5th Int. Conf. on Grid and Cooperative Computing*. Hunan, China, 2006, pp. 463–469. doi: 10.1109/GCCW.2006.58
19. Yang L., Xu H. Wireless localization using ultra-wideband signals. In *Handbook of Position Location: Theory, Practice, and Advances*. John Wiley and Sons, 2012, pp. 245–277. doi: 10.1002/9781118104750.ch8
20. Chakeres I.D., Belding-Royer E.M. AODV routing protocol implementation design. *Proc. 24th Int. Conf. on Distributed Computing Systems Workshops*. Hachioji, Japan, 2004, vol. 24, pp. 698–703.
21. Gnawali O., Fonseca R., Jamieson K., Kazandjieva M., Moss D., Levis P. CTP: an efficient, robust and reliable collection tree protocol for wireless sensor networks. *ACM Transactions on Sensor Networks*, 2013, vol. 10, no. 1, art. 16. doi: 10.1145/2529988
22. Lee E.A. Cyber-physical systems: design challenges. *Proc. 11th IEEE Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC)*. Orlando, USA, 2008, pp. 363–369. doi: 10.1109/ISORC.2008.25
23. Sangiovanni-Vincentelli A. Quo vadis, SLD? Reasoning about the trends and challenges of system level design. *Proceedings of the IEEE*, 2007, vol. 95, no. 3, pp. 467–506. doi: 10.1109/JPROC.2006.890107
24. Davare A., Densmore D., Guo L., Passerone R., Sangiovanni-Vincentelli A., Simalatsar A., Zhu Q. MetroII: a design environment for cyber-physical systems. *Transactions on Embedded Computing Systems*, 2013, vol. 12, art. 49. doi: 10.1145/2435227.2435245
25. Sangiovanni-Vincentelli A., Damm W., Passerone R. Taming Dr. Frankenstein: contract-based design for cyber-physical systems. *European Journal of Control*, 2012, vol. 18, no. 3, pp. 217–238. doi: 10.3166/EJC.18.217-238
26. Lee E.A., Neuendorffer S., Wirthlin M.J. Actor-oriented design of embedded hardware and software systems. *Journal of Circuits, Systems and Computers*, 2003, vol. 12, no. 3, pp. 231–260. doi: 10.1142/S0218126603000751
27. Kirov D.A., Ozhiganov A.A. Analysis of wireless sensor and actuator networks design methods. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2013, no. 1 (83), pp. 129–135. (in Russian)
28. Zekavat S.A. Channel modeling and its impact on localization. In *Handbook of Position Location: Theory, Practice, and Advances*. John Wiley and Sons, 2012, pp. 105–135. doi: 10.1002/9781118104750.ch4
29. Boulis A. Castalia: revealing pitfalls in designing distributed algorithms in WSN. *Proc. 5th ACM Conference on Embedded Networked Sensor Systems*. Sydney, Australia, 2007, pp. 407–408. doi: 10.1145/1322263.1322318
30. Koepke A., Swigulski M., Wessel K., Willkomm D., Klein Haneveld P.T., Parker T.E.V., Visser O.W., Lichte H.S., Valentin S. Simulating wireless and mobile networks in OMNET++: the MiXiM vision. *Proc. 1st Int. Conf. on Simulation Tools and Techniques for Communications, Networks and Systems, SIMUTOOLS'08*. Marseille, France, 2008, pp. 263–284.
31. *System Design, Modeling and Simulation using Ptolemy II*. Ed. C. Ptolemaeus. Ptolemy.org Publ., 2013, 685 p.
32. Baldwin P., Kohli S., Lee E.A., Liu X., Zhao Y. VisualSense: visual modeling for wireless and sensor network systems. *Technical Memorandum UCB/ERL M05/25*. University of California, Berkeley, 2005, 68 p.
33. Minakov I., Passerone R. PASES: an energy-aware design space exploration framework for wireless sensor networks. *Journal on Systems Architecture,* 2013, vol. 59, no. 8, pp. 626–642. doi: 10.1016/j.sysarc.2013.05.020
34. *Transaction-Level Modeling with SystemC*. Ed. F. Genassia. Springer, 2005, 272 p. doi: 10.1007/b137175
35. Fummi F., Quaglia D., Stefanni F. A systemC-based framework for modeling and simulation of networked embedded systems. *Proc. Forum on Specification, Verification and Design Languages, FDL'08*. Stuttgart, Germany, 2008, pp. 49–54. doi: 10.1109/FDL.2008.4641420
36. Eriksson J., Osterlind F., Finne N., Tsiftes N, Dunkels A., Voigt T., Sauter R., Marron P.J. COOJA/MSPSim: interoperability testing for wireless sensor networks. *Proc. 2nd Int. Conf. on Simulation Tools and Techniques*. Rome, Italy, 2009. doi: 10.4108/ICST.SIMUTOOLS2009.5637

37. Du W., Mieyeville F., Navarro D., O'Connor I. IDEA1: a validated SystemC-based system-level design and simulation environment for wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2011, pp. 143. doi: 10.1186/1687-1499-2011-143

38. Lizarraga A., Ding L., Hiner J., Lysecky R., Lysecky S., Gordon-Ross A. ATLeS-SN: a modular simulator for wireless sensor networks. *Design Automation for Embedded Systems*, 2013, vol. 16, no. 4, pp. 265–291. doi: 10.1007/s10617-013-9109-2

39. Bayer C., Katoen J.-P. *Principles of Model Checking*. Cambridge, MIT Press, 2008, 984 p.

40. Sebastiani R. Lazy satisfiability modulo theories. *Journal on Satisfiability, Boolean Modeling and Computation*, 2007, vol. 3, pp. 141–224.

41. Biere A. Bounded model checking. *Frontiers in Artificial Intelligence and Applications*, 2009, vol. 185, no. 1, pp. 457–481. doi: 10.3233/978-1-58603-929-5-457

42. Basu A., Bensalem S., Bozga M., Caillaud B., Delahaye B., Legay A. Statistical abstraction and model checking of large heterogeneous systems. *Lecture Notes in Computer Science*, 2010, vol. 6117 LNCS, pp. 32–46. doi: 10.1007/978-3-642-13464-7_4

43. Kwiatkowska M., Norman G., Parker D. PRISM 4.0: verification of probabilistic real-time systems. *Lecture Notes in Computer Science*, 2011, vol. 6806 LNCS, pp. 585–591. doi: 10.1007/978-3-642-22110-1_47

44. Boyer B., Corre K., Legay A., Sedwards S. PLASMA-lab: a flexible, distributable statistical model checking library. *Lecture Notes in Computer Science*, 2013, vol. 8054 LNCS, pp. 160–164. doi: 10.1007/978-3-642-40196-1_12

45. Bertinato M., Ortolan G., Maran F., Marcon R., Marcassa A., Zanella F., Zambotto M., Schenato L., Cenedese A. *RF localization and tracking of mobile nodes in wireless sensor networks: architectures, algorithms and experiments*. University of Padua, Italy, 2007.

46. Guerrero-Ibanez J., Flores-Cortes C., Zeadally S. Vehicular ad-hoc networks (VANETs): architecture, protocols and applications. In *Next-Generation Wireless Technologies*. Springer, 2013, pp. 49–70.

47. Boukerche A., Oliveira H.A., Nakamura E.F., Loureiro A.A. Vehicular ad-hoc networks: a new challenge for localization-based systems. *Computer Communications*, 2008, vol. 31, no. 12, pp. 2838–2849. doi: 10.1016/j.comcom.2007.12.004

48. Drenjanac D., Tomic S., Aguera J., Perez-Ruiz M. Wi-fi and satellite-based location techniques for intelligent agricultural machinery controlled by a human operator. *Sensors (Switzerland)*, 2014, vol. 14, no. 10, pp. 19767–19784. doi: 10.3390/s141019767

49. Rappaport T.S. *Wireless Communications: Principles and Practice*. 2nd ed. Prentice Hall, 2002, 736 p.

50. Cimatti A., Palopoli L., Ramadian Y. Symbolic computation of schedulability regions using parametric timed automata. *Proc. Real-Time Systems Symposium*. Barcelona, Spain, 2008, pp. 80–89. doi: 10.1109/RTSS.2008.36

51. Simalatsar A., Ramadian Y., Passerone R., Lampka K., Perathoner S., Thiele L. Enabling parametric feasibility analysis in real-time calculus driven performance evaluation. *Proc. 14th Int. Conf. on Compilers, Architectures and Synthesis for Embedded Systems*. Teipei, Taiwan, 2011, pp. 155–164. doi: 10.1145/2038698.2038723

52. Brunelli D., Minakov I., Passerone R., Rossi M. POVOMON: an ad-hoc wireless sensor network for indoor environmental monitoring. P*roc. 6th IEEE Workshop on Environmental, Energy and Structural Monitoring Systems*. Naples, Italy, 2014, pp. 175–180. doi: 10.1109/EESMS.2014.6923287

53. Kleissl J., Agarwal Y. Cyber-physical energy systems: focus on smart buildings. *Proc. 47th Design Automation Conference*. Anaheim, USA, 2010, pp. 749–754. doi: 10.1145/1837274.1837464

54. Ye W., Heidemann J., Estrin D. An energy-efficient MAC protocol for wireless sensor networks. *Proc. 21st IEEE INFOCOM*. NY, 2002, vol. 3, pp. 1567–1576. doi: 10.1109/INFCOM.2002.1019408

55. Lu C., Xing G., Chipara O., Fok C.-L., Bhattacharya S. A spatiotemporal query service for mobile users in sensor networks. *Proc. 25th IEEE Int. Conf. on Distributed Computing Systems, ICDCS*. Columbus, USA, 2005, pp. 381–390.

Dmitrii A. Kirov	–	PhD student, University of Trento, Trento, 38123, Italy, redgyy@gmail.com
Roberto Passerone	–	PhD, Professor, Associate professor, University of Trento, Trento, 38123, Italy, roberto.passerone@unitn.it
Aleksander A. Ozhiganov	–	D.Sc., Professor, Professor, ITMO University, Saint Petersburg, 197101, Russian Federation, ojiganov@mail.ifmo.ru
Киров Дмитрий Александрович	–	аспирант, Университет Тренто, Тренто, 38123, Италия, redgyy@gmail.com
Пассероне Роберто	–	PhD, профессор, доцент, Университет Тренто, Тренто, 38123, Италия, roberto.passerone@unitn.it
Ожиганов Александр Аркадьевич	–	доктор технических наук, профессор, профессор, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, ojiganov@mail.ifmo.ru

Dmitrii A. Kirov received his BS and MS degrees in Informatics and Computer Science from the National Research University of Information Technologies, Mechanics and Optics in 2009 and 2011 respectively (with Networks and Telecommunications curriculum track in the Master program). He is currently a PhD student of the Information and Communication Technologies (ICT) Doctoral school at the University of Trento, Italy, and a member of Embedded Electronics and Computing Systems (EECS) research group. His research interests include embedded systems design methodologies (in particular, cyber-physical systems), real-time location systems (RTLS) and wireless sensor networks.

Киров Дмитрий Александрович получил степени бакалавра и магистра в области информатики и вычислительной техники в Университете ИТМО в 2009 и 2011 г.г. (специализация магистерской программы – сети ЭВМ и телекоммуникации). В настоящее время он является аспирантом подразделения информационных и телекоммуникационных технологий в Университете Тренто, Италия, а также членом исследовательской группы встраиваемой электроники и вычислительных систем (EECS). Его научные интересы включают в себя методики проектирования встроенных систем (в частности, киберфизических систем), системы позиционирования объектов в реальном времени (RTLS) и беспроводные сенсорные сети.

Roberto Passerone is an Assistant Professor at the Department of Information Engineering and Computer Science at the University of Trento, Italy. He received his MS and PhD degrees in Electrical Engineering and Computer Sciences from the University of California, Berkeley, in 1997 and 2004, respectively. Before joining the University of Trento, he was Research Scientist at Cadence Design Systems. Prof. Passerone has published numerous research papers on international conferences and journals in the area of design methods for systems and integrated circuits, formal models and design methodologies for embedded systems, with particular attention to image processing and wireless sensor networks. He was track chair for the Real-Time and Networked Embedded Systems at ETFA from 2008 to 2010, and general and program chair for SIES from 2010 to 2015. He has participated to several European projects on design methodologies, including SPEEDS, SPRINT and DANSE, and was local coordinator for ArtistDesign, COMBEST, and Cy-PhERS.

Роберто Пассероне является профессором кафедры информатики и вычислительной техники в Университете Тренто, Италия. Он получил степени магистра и PhD в области вычислительной техники в университете Беркли (США, Калифорния) в 1997 и 2004 годах. Перед началом работы в Университете Тренто он являлся научным сотрудником Cadence Design Systems. Р. Пассероне является автором многочисленных публикаций в международных изданиях в области проектирования систем и интегральных схем, формальных моделей и методик проектирования встроенных систем, в частности, систем обработки изображений и беспроводных сенсорных сетей. С 2008 по 2010 гг. он был председателем секции систем реального времени и сетевых встроенных систем на Международной конференции перспективных технологий и промышленной автоматизации (Emerging Technologies and Factory Automation – ETFA). С 2010 по 2015 г.г. являлся председателем организационного и программного комитетов Международного симпозиума по промышленным встроенным системам (IEEE International Symposium on Industrial Embedded Systems – SIES). Он участвовал в ряде европейских проектов по методикам проектирования систем и аппаратуры, таких как SPEEDS, SPRINT и DANSE, а также являлся локальным координатором проектов ArtistDesign, COMBEST и CyPhERS.

Aleksandr A. Ozhiganov, doctor of technical sciences, professor, professor of the Computer Science department of the National Research University of Information Technologies, Mechanics and Optics (ITMO) and leading researcher of the JSC "Avangard". He is the author of numerous published scientific works, 14 educational materials, more than 15 patents. Prof. Ozhiganov is an honorable fellow of higher professional education of Russian Federation. He is a member of the editorial board of the Scientific and Technical Journal of Information Technologies, Mechanics and Optics and the deputy chief editor of the "Izvestija VUZov - Priborostrojenie" (Proceedings of higher education institutions – Instrumentation engineering) journal. He is also a member of two dissertation councils in the University ITMO and a member of joined dissertation council in the JSC "Avangard".

Ожиганов Александр Аркадьевич, доктор технических наук, профессор, профессор кафедры вычислительной техники Университета ИТМО, главный научный сотрудник ОАО «Авангард». Автор более 80 опубликованных научных работ, 14 учебно-методических пособий, более 15 патентов. Почетный работник высшего профессионального образования Российской Федерации. Член редакционной коллегии журнала «Научно-технический вестник информационных технологий, механики и оптики». Заместитель главного редактора журнала «Известия высших учебных заведений. Приборостроение». Член двух диссертационных советов при Университете ИТМО и объединенного диссертационного совета при ОАО «Авангард».