Model-Based Analysis of Secure and Patient-Dependent Pacemaker Monitoring System

Leonidas Tsiopoulos[1,2(✉)], Alar Kuusik[1], Jüri Vain[1], and Hayretdin Bahsi[1]

[1] Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia
{leonidas.tsiopoulos,alar.kuusik,juri.vain,hayretdin.bahsi}@taltech.ee
[2] Åbo Akademi University, Tuomiokirkontori 3, 20500 Turku, Finland

Abstract. Pacemakers' safety, security and reliability are of utmost importance for patient's life quality in various daily situations. An integral characteristic of the pacemaker that depends on all of these attributes is its lifetime. In current medical practice the pacemaker's expected lifetime is estimated relying on manufacturer's data sheet and expert knowledge that may result in quite rough approximations if patient's specifics are not taken into account. In this paper we perform a model-based quantitative analysis of pacemaker lifetime that takes into account patient specific factors, including general health condition, acting environment, remote reporting and others. We demonstrate that including these factors in analysis can provide drastically different results compared to that of average approximating estimates.

Keywords: Cardiac implanted electronic devices · Pacemaker · UPPAAL timed automata.

1 Introduction

The embrace of sensors and advances in communication and cloud technology has enabled us to develop effective monitoring applications in different areas including the health sector. Implantable medical devices such as cardiac implanted electronic devices (CIEDs), insulin pumps and gastric stimulators can continuously track a patient and transfer the data to the medical institution regardless of the location. CIEDs, namely permanent pacemakers (PPM), implantable cardioverter-defibrillators (ICDs) and cardiac resynchronization therapy (CRT) devices exceed 500 units per million inhabitants worldwide [21] and their number is increasing in correlation with the expected lifetime and national wealth. In well-developed countries the average prevalence is over 1000 implants today and reaches 10000 devices per 1M elderly people (ibid). For such remarkable deployment rate the reliability and operational lifetime of devices is an important question to prevent fatalities. According to patient support organizations the complete

M. M. Alam et al. (Eds.): BODYNETS 2020, LNICST 330, pp. 77–91, 2020.
https://doi.org/10.1007/978-3-030-64991-3_6

failure of a modern pacemaker is rare [11] and most frequently devices need just timely reprogramming for proper operation [1]. Acquiring timely information of patient condition changes and the discovery of possible mismatch of device settings requires periodic device operation analysis to discover irregularities in the heart behavior. Regardless the maintenance related usability concerns it is important to underline that remote monitoring of pacemakers and other CIEDs has improved the patient survival rate [25] and not only. The accurate predicting of device maintenance dates not only prevents malfunctioning in conditions when the monitoring is urgently needed but also improves the user trust and convenience of wearing the devices without compromising patient's life quality.

Permanent pacemaker communication is required for device adjustments according to patient's physical condition and its deviations over the time. Due to technological limitations, mainly because of the high energy consumption of legacy wireless communication, frequent data transmissions of pacemakers were not possible till recent times. Inductive link coupling used for legacy pacemakers made data exchange - device programming and event report offloading possible only at hospital environments. Therefore, not much attention has been paid to pacemaker communication security and energy efficiency measures. Bluetooth connectivity for PPMs was proposed already in 2002 by Hsu and Avudaiappan [14]. However, Bluetooth Classic technology available at this time was not suitable for long duration battery operation. The energy efficient Bluetooth Low Energy (BLE) connected pacemakers were actually introduced to the market in 2016 and are considered simple to use and helpful for both sides, patients and cardiologists [24]. Today, there is a selection of modern PPM and ICD devices from companies Medtronic and Abbott that provide BLE connectivity with an acceptable reduction of the device operation lifetime. However, the impact of wireless communication to device energy consumption is still remarkable; an active modern Bluetooth transmitter consumes around one milliamper current while modern microcontrollers in pacemakers can operate with few microamps per pacing in average. Therefore it is important to optimize wireless communication frequency and data amount to be transmitted heavily depending on patient's clinical needs. It is also essential to pay attention to personal data protection and protection against external operational vulnerabilities to CIEDs as life supporting medical devices.

Deployment of advanced wireless communication made possible periodic transmission of heart data to cardiologist which significantly improved the patient monitoring quality but it also added security risks and reduced device battery lifetime. As argued in [20] wireless interfaces of such devices are vulnerable to cyber attacks with even life-threatening impact. Various passive and active attacks targeting the communication medium are possible [17]. The upgrade mechanisms of the PPMs and the vulnerabilities of the cloud server could be also used for compromising the devices remotely [2]. Moreover, pacemakers of St. Jude Medical were found to have insufficient cyberprotection measures that made possible battery drain, firmware downgrade and malfunctioning attacks over the wireless interface [16]. As various nodes such as home routers, mobile

phones, enterprise gateways, would take part in the communication, the range of security threats gets very wide. Therefore, it is essential to establish end-to-end security that covers the whole communication medium encompassing the device, cloud system, and all intermediate nodes. On the other side, the security countermeasures create an additional burden to resource-constrained devices with the additional energy consumption affecting significantly the device lifetime. From the clinical point of view, requirements to communication frequency and payload depend on patient's profile (health condition, daily activities, environment etc.) that in combination with the security countermeasures on the communication can result in largely varying expectations to device lifetime.

To reduce threats and possibly infeasible design decisions (that ignore the factors coming from different use cases) a model-based formal analysis can be used to justify design and runtime configuration decisions. The formal analysis in this paper is targeted to taking into account the effects of pacemaker remote monitoring and secured dynamic episode alerting to the device lifetime, depending on different patient profiles and security countermeasures adjusted to daily living environment. We apply probabilistic modeling, simulation experiments and model-checking with UPPAAL Timed Automata to estimate the impact of various factors to pacemaker expected lifetime. In particular, we focus on factors such as averaged patient-dependent condition episode alerting frequency and the cost of secure communication (in terms of energy consumption) in three possible patient environments during a day. The impact of listed factors is characterized with numerical data extracted from model experiment that justifies the need for refined model-based analysis of pacemaker operational lifetime. The choice of UPPAAL Timed Automata as the formalism for our work is justified by the facts that the checking of behavioral, stochastic and real-time aspects of systems is efficient and the UPPAAL tool family includes several extensions of the standard modeling and verification tool which will allow us in the future to address different development aspects of pacemakers.

2 Related Work

In the context of wireless Body Area Sensor Networks (BASN) extensive research has been done with application of formal methods. Ahmed et al. [6] used higher-order-logic theorem proving to formally analyse energy consumption of BASNs by verifying the mathematical relations for energy, delay and distortion of a given BASN. The results of this analysis can then be used to determine the parameters of optimal energy consumption of BASN algorithms. In a closely related work, Dai et al. [12] proposed synthesizing power management strategies for wireless sensor network nodes with UPPAAL STRATEGO by taking into account the various power states of a node device during the runtime in order to achieve a tradeoff between power consumption and performance.

Regarding verification of wireless body area network protocols, several works exist in the literature. For example, timed automata have been used in [8] to verify a recently proposed Medium Access Control protocol called STDMA (Statistical frame based TDMA protocol). Taking also security into accout, Chen

et al. [10] proposed a formal modeling and verification method using the PAT model-checker for wireless BASN-specific authentication security protocols.

However quite few results are available on the verification of pacemaker operation correctness using model-checking tools. An operational model verification with UPPAAL tool is conducted by Pajic et al. [19]. As stated in the Introduction, pacemakers were found to have insufficient cyberprotection measures [2,16]. This triggered research work to present solutions for protection against cyber attacks under the assumption the communication has been already compromised. For example, Rao et al. [22] presented a model for a multi-modal design approach for risk assessment of pacemaker devices and they propose an adaptive remediation scheme to mitigate security threats. The approach is integrated into the hardware-software development with a middleware for dynamic switching between the modes based on risk values assigned to the different functions each mode has.

The analysis of the recent related work, with some of the example papers discussed above, shows the lack of satisfying solution in the pacemaker literature and motivates the authors for the work presented in this paper. To the best of authors' knowledge, a formal model-based analysis on the effects of pacemaker remote monitoring and secured dynamic episode alerting to the device lifetime, taking into account different patient profiles regarding heart condition and security measures adjusted to daily living environment, has not been presented yet.

3 Pacemaker Background

In general, an implantable pacemaker monitors and regulates the patient's heart rate continuously by providing single or dual chamber rate-responsive corrective bradycardia and atrial tachyarrhythmia electrical pacing. As a reference device of current study we use Medtronic dual chamber permanent pacemaker model Azure XT DR W1DR01 [4].

3.1 Pacing

Dual chamber and single chamber pacing modes address different cardiac conditions. Dual chamber pacing restores AV synchrony by sensing and stimulating two chambers of the heart, the right atrium and right ventricle. Single chamber pacing supports patients with infrequent asystole or patients with chronic AT/AF and for whom dual chamber pacing is not justified [4]. For some of the dual chamber pacing modes pacing occurs at the programmed lower rate and for some modes the pacing is occuring at the sensor rate. Rate-responsive pacing adapts the pacing rate to changes in patients' physical activity due to some patients exhibiting heart rates that do not adapt to changes in their physical activity. The device uses an activity sensor to measure the patient's movement and to determine the appropriate pacing rate.

3.2 Monitoring and Alerting

Data collected by the pacemaker is encrypted and sent to the CareLink network through the MyCareLink Heart mobile app [3], providing clinicians with alerts on clinically-relevant patient events. The app also makes selected pacemaker data, such as transmission success history, pacemaker battery information and updates on physical activity, easily accessible to patients. If a clinical or system performance event occurs and Medtronic CareAlert Monitoring is programmed to respond with an alert, the device automatically attempts to establish wireless communication with the mobile phone. After communication is established, the mobile phone receives the alert data from the device, and then transmits the alert data to the CareLink Network wherefrom the data is accessible to clinicians.

Examples of clinically-relevant alerts are "*Average Ventricular Rate During AT/AF larger than Threshold*" indicating that the average ventricular rate during a selectable duration of AT/AF exceeds the programmed threshold, and "*Monitored VT Episode Detected*" indicating that one or more monitored VT episodes were detected.

The clinician may also configure the device to send periodic reports with the frequency depending on the patient. The information included in such reports is, e.g., "episode data and EGM storage", providing an arrhythmia episode log that enables to view the summary and detailed diagnostic data, including stored EGM, for the selected arrhythmia episode, and "rate drop response episodes data" displaying beat-to-beat data that is useful in analyzing Rate Drop Response episodes and the events leading up to these episodes.

3.3 Pacemaker Security Countermeasures

We assume that the communication between pacemaker and cloud is secured with end-to-end mechanisms. From the connectivity perspectives, device performs internal data encryption and communicates with smartphone or tablet PC gateway over BLE communication using Medtronic BlueSync technology. Gateway device runs Medtronic MyCareLink Heart application that acts as a pass-through element for the encrypted pacemaker data.

In a typical scenario, three main security tasks should be performed in a system: (1) relevant cryptographic keys should be generated, (2) each device should be authenticated and authorized, (3) based on the generated keys, secure end-to-end communication should be established. In typical end-to-end communication, a gateway acts as an intermediary node between sensor nodes and the cloud. Gateways obtain the sensor data with short-range local area network protocols (i.e., wireless or wired) and relay them to the cloud server over wide area network infrastructures. In addition to relaying responsibility, these gateways could take over some cryptographic operations to relieve the sensors from resource-intensive tasks [13,15,18]. However, this functionality comes with a compromise in the end-to-end security property, requiring more trust to the gateway.

In most cases, these gateways do not have resource restrictions like sensors have but may be subject to different threats or security controls. In stationary locations, a patient can use the existing network infrastructures, thus, a pacemaker can benefit from hospital gateway or home router. Hospital gateways could be more trustable as they could be secured by IT staff of the hospitals, and we can assume that more physical security is guaranteed. This can eliminate the cyber threats requiring physical proximity to the target. On the other side, an ordinary user generally relies on the default configuration of the home routers and ignores the security hardening. Thus, it is reasonable to delegate some cryptographic tasks to the hospital gateways but not to home routers.

A mobile phone could be a perfect fit as a gateway while the patient visits non-stationary places. However, it is not trustable for sensitive tasks as mobile malware poses an important threat to the secure communication passing over the mobile phone. These devices are prone to physical loss or theft. Most of the public places may enable the attackers to come close to the target and conduct man-in-the-middle (MitM) or denial of service (DoS) attacks. Comparing with the home environment, despite the existence of similar threats in home environment, the likelihood of threats could be considered higher for communication over the mobile phone in public places. Therefore, we assume that a stronger security configuration should be enabled in this option and any compromise in end-to-end security is not tolerable.

In resource-constraint scenarios such as our pacemaker remote monitoring case, security mechanisms are required to be use-case adaptable and lightweight, meaning that they should operate with less storage, computing, and energy resources. The current study does not follow proprietary BlueSync implementation that, according to our knowledge, supports a single security model for the pacemaker data communications. The strength of lightweight security mechanisms is that they can be deployed according to the patient environments with varying threat profiles. As we assume in this study the patient profile is determined by being in different locations such as hospital, home, and other places during a day, we selected a distinct security configuration setting for each location. Hospital and home are more stationary ones whereas the third option covers all other places a user can visit or stay at (e.g., shops, public transport, etc.).

We assume that the pacemaker application uses Datagram Transport Layer Security (DTLS) which can be considered a UDP-based alternative of transport layer security (TLS) protocol [23]. A DTLS session starts with a handshake for authenticating the parties and exchanging the session keys. The parties use the agreed session key to perform secure communication. However, in the case of using a certificate, large messages should be fragmented into various packets exchanged between parties, ending up with huge energy consumption for the handshaking phase. Therefore, delegation schemes in which gateways conduct handshake operation on behalf of sensors are proposed in the literature [13,15,18].

We consider the security configuration options given in the benchmarked study, [18], as the main baseline. The results in this study are very relevant for our cases as it provides a detailed performance analysis of a health-

care IoT system and proposes a solution that uses delegation idea (i.e., namely DTLS session resumption) which is compared with other configuration options. We assume that in a hospital setting, DTLS session resumption using elliptic curve operations for certificates and transferring handshake responsibility to the gateway is enabled. In home environment, symmetric key-based DTLS (DTLS_PSK_WITH_AES_128_CCM_8) is utilized. The pacemaker is assumed to initiate a certificate-based handshake using Elliptic Curve Digital Signature Algorithm in non-stationary places (i.e., security option is DTLS_ECDH_ECDSA_WITH_AES_128_CCM_SHA_256) which has a higher cost but provides more assurance about the authentication of the pacemaker. We omit the details about the key generation algorithms for simplicity.

In our model, as the frequency of the data transmission is so low, we consider that a handshake happens for each transmission of monitoring data or alert. As the data sizes are also low, one or two messages are enough after agreeing on the session key, considering the available payload sizes after excluding all headers. Therefore, we assume that energy consumption is heavily determined by handshake operation and the sensor side energy consumption values given in [18] are considered as the baseline for the calculation and correlation of energy costs of each security configuration for the model (i.e., DTLS session resumption costs 1/6 less than Symmetric key-based DTLS and certificate-based DTLS costs 32 times more than Symmetric key-based DTLS).

4 UPPAAL Probabilistic Timed Automata

UPPAAL Timed Automata (UTA) [7] address the behavioral and timing aspects of systems providing efficient data structures and algorithms for their representation and analysis through simulation and model checking.

An UTA is given as the tuple $(L, E, V, CL, Init, Inv, T_L)$, where L is a finite set of locations, E is the set of edges defined by $E \subseteq L \times G(CL, V) \times Sync \times Act \times L$, where $G(CL, V)$ is the set of constraints in guards, $Sync$ is a set of synchronization actions over channels and Act is a set of sequences of assignment actions with integer and boolean expressions as well as with clock resets. V denotes the set of integer and boolean variables. CL denotes the set of real-valued clocks ($CL \cap V = \varnothing$). $Init \subseteq Act$ is a set of assignments that assigns the initial values to variables and clocks. $Inv : L \rightarrow I(CL, V)$ is a function that assigns an invariant to each location and $I(CL, V)$ is the set of invariants over clocks CL and variables V. $T_L : L \rightarrow \{ordinary, urgent, committed\}$ is the function that assigns the type to each location of the automaton.

UPPAAL Probabilistic Timed Automata (UPTA) [9] is a stochastic and statistical modeling extension of UTA. UPTA preserves the standard UTA constructs such as integer variables, data structures and user-defined C-like functions. Additionally, UPTA support branching edges where weights can be added to define a probability distribution on discrete transitions. The weights may be general expressions that depend on the states and not just simple constants. For the work in this paper we use the branching edges with probability weights.

The requirement specification language (in short, query language) of UTA, used to specify properties to be model checked, is a subset of Timed Computation Tree Logic (TCTL) [7]. The query language consists of path formulae and state formulae. State formulae describe properties that can be interpreted in individual states, whereas path formulae quantify over paths or traces of the model and can be classified into reachability, safety and liveness [7]. For this paper we consider safety properties that are specified with path formula $\mathsf{A}\Box\phi$ stating that state formula ϕ should be true in all reachable states.

5 Pacemaker Monitoring and Dynamic Alert Model

For the objective of this paper the model concerns only the continuous monitoring and possible episode alerting from the pacemaker in a probabilistic manner depending on three different patient profiles listed as Type 0, Type 1 and Type 2.

5.1 Patient Profiles

The "best" case patient profile (Type 0) is away from home (traveling to work and back, being at work and any possible shopping) 10 h per day. Hence, he is at home 14 h per day. He visits the hospital once per year and this visit is 2 h long. Regarding dynamic alerts, he has 1 alert per year in any of the three possible environments.

The "medium" case patient profile (Type 1) is away from home 6 h per day. He visits the hospital 2 times per year and each visit is 4 h long. Regarding dynamic alerts, he has 4 alerts per year in any of the three possible environments.

The "worst" case patient profile (Type 2) is away from home 1 h per day. He visits the hospital 4 times per year and each visit is 2 d (48 h) long. Regarding dynamic alerts, he has 10 alerts per day in any of the three possible environments.

5.2 Modeling Alerting Cost, Parameters and Constants

The device battery characteristics are as follows. The mean usable capacity is 1.2 Ah. Depending on the device setting/programming per patient needs, the lifetime of the device varies from 7.4 to 15.8 years (see Tables 4, 5, 8 in technical manual [5]). For our model let us choose the value 13.7 years/5000 d for the lifetime of the device regardless the patient profile. Note that all numbers for the model can be changed accordingly to different patient profiles and concrete living scenarios. One of the main assumptions is that daily reporting is active for all patients. This reduces lifetime of the device battery by 14.4% which translates to 564 d less lifetime. After applying simple calculations in order to have reasonable numbers for the model parameters to work on a day basis for the lifetime reduction, we know that 189 daily reporting sessions are needed to reduce battery lifetime by 1 day.

The dynamic alerts have a different cost in terms of battery lifetime depending on the environment the patient is in when the alert is triggered. This is due to different security protocols applied for communication in different environments. An additional assumption is that the alert triggered when at home costs the same as sending the daily report meaning that the security countermeasure cost is the same for both. Then, we assume that the cost of sending an alert from hospital is 1/6 less than the cost of an alert from home and the cost of the alert while traveling is 32 times higher than the cost of the alert from home, taking into account recommendations in Sect. 3.3. Thus, 220 hospital alerting sessions and 6 travel alerting sessions are needed in order to reduce battery lifetime by 1 day, respectively.

Figure 1 shows the model parameters, variables and constants. P is the patient type index with domain $\{0, 1, 2\}$. N is the number of patient profile types, 0 for "best", 1 for "medium" and 2 for "worst". M is the normalization to 1000 of the probability weight in order to accommodate all model probability weights into approximating integer scale from 0 to 1000. On this scale Ph denotes the probability weight of being at home during any hour around the clock. Similarly Pt is the probability weight of traveling and Pv is the probability weight of visiting the hospital during any hour around the clock. Pr is the probability weight of daily reporting and it is the same for all patients. Pa is the probability weight of alerting at any hour. Alerting may occur in any of the three different environments. Variable *env* is for the environment the patient is currently located in and it can be assigned symbolic values *home*, *travel* and *hospital* for the environment options.

```
const int P = 0;
const int N = 3;
const int M = 1000;

const int [0,M] Ph[N] = {583, 748, 938};
const int [0,M] Pt[N] = {417, 250, 41};
const int [0,M] Pv[N] = {0, 2, 21};
const int [0,M] Pr[N] = {42, 42, 42};
const int [0,M] Pa[N] = {1, 4, 417};

int env = 0;
const int home = 0;
const int travel = 1;
const int hospital = 2;
```

```
int lifetime = 5000;

const int deltaLTN = 1;
const int deltaLTH = 1;
const int deltaLTT = 1;
const int deltaLTO = 1;

int crD;
const int cD = 189;
int crO, crT, crH;
const int cO = 220, cT = 6, cH = 189;
```

Fig. 1. Pacemaker UPPAAL model parameters, variables and constants.

Variable $lifetime$ expresses the battery lifetime estimate in days which is calculated iteratively in every hour (model time) depending on different environment and communication events that can occur with probability weights

specified as described above. The initial value of battery lifetime estimate (without any battery use) is 5000 d. $deltaLTN$ is a constant showing the decrease of battery lifetime by 1 day after 189 daily reporting sessions. $deltaLTH$ is a constant for the decrease of battery lifetime by 1 day after 189 home dynamic alerting sessions. $deltaLTT$ is a constant for the decrease of battery lifetime by 1 day after 6 dynamic alerting sessions during travel. $deltaLTO$ is a constant for the decrease of battery lifetime by 1 day after 220 hospital dynamic alerting sessions. crD is a variable for counting sessions of daily reporting. This counter is reset after reaching constant value cD which, in turn, triggers the subtraction operation $lifetime - deltaLTN$. Similarly reaching values crO, crT and crH with corresponding constants cO, cT and cH trigger resets of variables crO, crT and crH, respectively. Letters O, T and H stand for hospital, traveling and home, respectively.

5.3 Pacemaker Model

Figure 2 shows the complete model composed of interacting automata Patient, Device and HourClock. The hour clock of the system model is depicted in the lower part of the figure. The state updates of Patient and Device are triggered every 1 h with synchronization channel chH. The automaton Patient that models possible moves of the patient between different environments is shown in the upper part of the figure and the pacemaker device automaton is shown in the middle part of the figure. The initial location of the patient is his/her home. Probability Ph indicates that he/she stays at home and probability $M - Ph$ indicates that he/she is traveling. Probability Pt indicates that he/she keeps traveling, while probability Pv indicates that he/she will visit the hospital and probability $M - Pt - Pv$ indicates that he/she will return back home. The value vectors of the probabilities calculated from assumptions can be seen in Fig. 1 where the elements of vectors correspond to different patient profiles 0, 1 and 2. A concrete element is chosen from the vectors Ph, Pt, Pv and Pa depending on what value of Patient template parameter P (denoting profile) is selected for the model experiment. Variable env is updated according to the patient location in current state and is used within branching conditions in Device automaton. Location value affects the type of dynamic alerts which in turn affect differently the battery lifetime as described above and shown in the device model template.

In the device automaton template there are two main locations, one modeling the regular Monitoring and the other for Alerting. From Monitoring with probability Pr daily reporting occurs and whenever variable crD (for counting sessions of daily reporting) reaches constant value cD the battery lifetime is decreased by constant $deltaLTN$. From Monitoring with probability $M - Pr$ daily reporting does not occur. During Monitoring dynamic episode alerting occurs with probability Pa being different for each patient type as it can be seen in Fig. 1 and battery lifetime is decreased depending on the environment the patient is currently located. With probability $M - Pa$ dynamic alerting does not occur in current hour. Since the minimal time step in the model is one hour, sub-hour activities' timing is ignored and modeled using *Committed* locations

in which the model time is not progressing, i.e., all state updates of transitions from *Committed* locations are instantaneous (in model time).

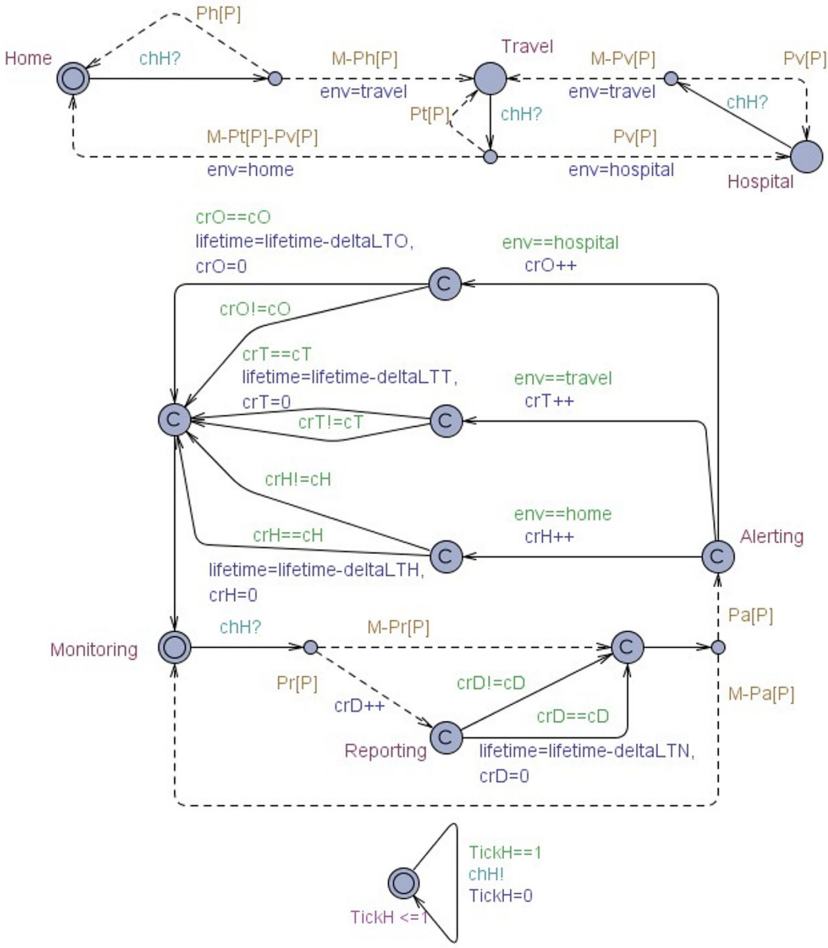

Fig. 2. Pacemaker UPPAAL probabilistic model

5.4 Pacemaker Model Analysis

The model depicted in Fig. 2 allows for checking critical battery lifetime properties depending on the different patient profiles and patient activities. For example, let us prove with model-checking that the expected lifetime is *safe*, i.e., the battery level never gets below a critical value before planned maintenance time, which could be 15% of its initial level. Assuming that the battery nominal lifetime is 5000 d and for simplicity we approximate the depletion rate being

Table 1. Experimental estimates of device lifetime for different patient profiles

Environment	**Depletion rates in env.** (days per alert)	**Impact of the environment**	**Patient Type 0**	**Patient Type 1**	**Patient Type 2**
Home	0,12714	Hrs in env. per day	14	18	22,475
		No of alerts in env. per day	0,0016	0,0082	9,3646
		Depletion per day (days per day of operation)	0,0002	0,00105	1,19063
Travel	4,06853	Hrs in env. per day	10	6	1
		No of alerts in env. per day	0,00114	0,00274	0,41667
		Depletion per day (days per day of operation)	0,00012	0,00029	1,69522
Hospital	0,10595	Hrs in env. per day	0,00548	0,0219	0,526
		No of alerts in env. per day	0,0000006	0,00001	0,21918
		Depletion per day (days per day of operation)	0,0000025	0,00004	0,02322
Total no of alerts per day			0,00274	0,01097	10,0004
Total expected lifetime			4435	4431	1239
Lifetime till 15% remaining battery lifetime			3770	3766	1053

constant the critical threshold will be reached by 4250 d (15% of 5000 d is 750 d). Also we use an auxiliary clock variable GCl in the query to refer to the time instances in the interval from 0 to $EXLT$, which stands for the expected lifetime until reaching the critical threshold value. The TCTL formula (1) expresses the property that in all states of all possible scenarios within closed time interval [0,$EXLT$] the calculated battery level never gets less than the $Critical$ value (750 d of remaining lifetime).

$$\mathsf{A}\Box\ \mathsf{GCl}\ <=\ EXLT\ \mathsf{imply}\ lifetime\ >=\ Critical \tag{1}$$

In addition to model-checking the correctness of pacemaker's maintenance schedule, concrete numerical estimates of battery lifetime can be generated by

UPPAAL simulation experiments under various scenarios. Table 1 exemplifies the results of simulation experiments for all three patient profiles described in Sect. 5.1. As can be seen from Table 1 the pacemaker expected lifetime average estimates for patient type 0 and type 2 differ drastically (over 3 times), and claiming common rough estimate regardless the specifics of patient profile is obvious risk to patient's safety. The factors, such as, frequency of alerting and security protocols used in different environments have substantial effect on pacemaker expected lifetime. While one would expect that the expected lifetime and maintenance deadline estimates for patient types 0 and 1 can be proved to be correct (Query 1 would be satisfied) and for patient type 2 the Query 1 would be clearly unsatisfiable, it turns out that if more than daily reporting is taken into account the query is not satisfiable for all patient types for this case study. That confirms our main hypothesis that patient profile specific analysis using patient and device models incorporating all influential factors is inevitable.

6 Conclusions

In this paper we have presented a model-based quantitative analysis of pacemaker lifetime that takes into account patient specific factors, including general health condition, acting environment, remote dynamic reporting of vital patient data and alternative security protection measures for these data communications. The study did not follow proprietary pacemaker BlueSync implementation that, according to our knowledge, supports a single security model. Instead, we proposed a multi-level security model that, depending on the security context, allows to select different security levels. Moreover, due to the fact that current patient safety regulations do not foresee remote modifications of the pacemaker firmware, our proposed cryptographic implementation should be applicable for the secure implantable device firmware upgrade as well in the further.

The analysis of the recent related work on development of BANs and specifically on implantable pacemakers motivated us for the work presented in this paper. We performed probabilistic modeling and model analysis with UPPAAL Timed Automata on a fully parameterizable model regarding some of typical patient profiles. For that we took into account averaged patient-dependent data for episode alerting frequency which is additionally assigned different security costs based on three possible patient environments during a day. The focus of our work was on pacemaker operational lifetime prediction with UPPAAL model-checking tool with different remote observation needs according to cardiovascular disease severity and location-dependent security requirements. The proposed approach to fine-grained battery lifetime analysis using probabilistic timed models is flexible in the sense that it can be easily adjusted by incorporating in the model only the factors that have impact to device endurance and reliable maintenance planning.

Acknowledgement. This work has been supported by the ERDF funded centre of excellence project EXCITE (2014–2020.4.01.15-0018), the Estonian Ministry of Education and Research institutional research grant no. IUT33-13 and supported in part by the Estonian Research Council grant PRG 424.

References

1. https://www.fairview.org/patient-education/116363EN, Accessed 15 August 2020
2. https://www.wired.com/story/pacemaker-hack-malware-black-hat/
3. https://www.medtronic.com/us-en/healthcare-professionals/products/cardiac-rhythm/managing-patients/improved-outcomes/remote-monitoring-options.html
4. Medtronic Azure MRI Surescan Pacemaker Reference Manual, https://www.medtronic.com/us-en/healthcare-professionals/products/cardiac-rhythm/pacemakers/azure.html
5. Medtronic Azure MRI Surescan Pacemaker Technical Manual. https://www.medtronic.com/us-en/healthcare-professionals/products/cardiac-rhythm/pacemakers/azure.html
6. Ahmed, A., Hasan, O., Tahar, S., Mohamed, A.: Formal verification of energy consumption for an eeg monitoring wireless body area sensor network. In: 2016 International Conference on Open Source Systems Technologies (ICOSST), pp. 18–22. IEEE (2016)
7. Behrmann, Gerd., David, Alexandre, Larsen, Kim G.: A tutorial on Uppaal. In: Bernardo, Marco, Corradini, Flavio (eds.) SFM-RT 2004. LNCS, vol. 3185, pp. 200–236. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30080-9_7
8. Ben Hamouda, R., Ben Hafaiedh, I.: Formal modeling and verification of a wireless body area network (WBAN) protocol: S-tdma protocol. In: 2017 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC), pp. 72–77. IEEE (2017)
9. Bulychev, P.E., et al.: UPPAAL-SMC: statistical model checking for priced timed automata. In: Wiklicky, H., Massink, M. (eds.) Proceedings 10th Workshop on Quantitative Aspects of Programming Languages and Systems (2012)
10. Chen, T., Yu, Z., Li, S., Chen, B.: From wireless sensor networks to wireless body area networks: formal modeling and verification on security using pat. J. Sens. **2016**, 1–11 (2016). https://doi.org/10.1155/2016/8797568
11. Cronin, B., Maus, T.M., Khoche, S., Rozner, M.A.: Chapter 4 - cardiovascular implantable electronic device management in noncardiac surgery. In: Kaplan, J.A., Cronin, B., Maus, T.M. (eds.) Essentials of Cardiac Anesthesia for Noncardiac Surgery, pp. 70–99. Elsevier, New York (2019). https://doi.org/10.1016/B978-0-323-56716-9.00004-7, http://www.sciencedirect.com/science/article/pii/B9780323567169000047
12. Dai, S., Hong, M., Guo, B.: Synthesizing power management strategies for wireless sensor networks with UPPAAL-STRATEGO. Int. J. Distrib. Sens. Networks **13**(4) (2017). https://doi.org/10.1177/1550147717700900
13. Granjal, J., Monteiro, E., Silva, J.S.: End-to-end transport-layer security for internet-integrated sensing applications with mutual and delegated ECC public-key authentication. In: 2013 IFIP Networking Conference, pp. 1–9. IEEE (2013)
14. Hsu, C.L., Avudaiappan, R.: Bluetooth technology. Academia Sinica Computing Centre **18**, (2002)
15. Hummen, R., Shafagh, H., Raza, S., Voig, T., Wehrle, K.: Delegation-based authentication and authorization for the IP-based internet of things. In: 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), pp. 284–292 (2014)
16. Kramer, D.B., Fu, K.: Cybersecurity concerns and medical devices: lessons from a pacemaker advisory. JAMA **318**(21), 2077–2078 (2017). https://doi.org/10.1001/jama.2017.15692

17. Li, C., Raghunathan, A., Jha, N.K.: Hijacking an insulin pump: security attacks and defenses for a diabetes therapy system. In: 2011 IEEE 13th International Conference on e-Health Networking, Applications and Services, pp. 150–156. IEEE (2011)
18. Moosavi, S.R., Nigussie, E., Levorato, M., Virtanen, S., Isoaho, J.: Performance analysis of end-to-end security schemes in healthcare IoT. Procedia Comput. Sci. **130**, 432–439 (2018)
19. Pajic, M., Jiang, Z., Lee, I., Sokolsky, O., Mangharam, R.: From verification to implementation: a model translation tool and a pacemaker case study. In: 2012 IEEE 18th Real Time and Embedded Technology and Applications Symposium, pp. 173–184. IEEE (2012)
20. Pycroft, L., Aziz, T.Z.: Security of implantable medical devices with wireless connections: the dangers of cyber-attacks. Expert Rev. Med. Devices **15**(6), 403–406 (2018). https://doi.org/10.1080/17434440.2018.1483235
21. Raatikainen, M.P., et al.: Statistics on the use of cardiac electronic devices and electrophysiological procedures in the European Society of Cardiology countries: 2014 report from the european heart rhythm association. EP Europace **17**(suppl-1), i1–i75 (2015). https://doi.org/10.1093/europace/euu300
22. Rao, A., Rozenblit, J., Lysecky, R., Sametinger, J.: Composite risk modeling for automated threat mitigation in medical devices. In: Proceedings of the Symposium on Modeling and Simulation in Medicine, MSM 2017, pp. 1–10. Society for Computer Simulation International, San Diego, CA, USA (2017)
23. Rescorla, E., Modadugu, N.: Datagram transport layer security version 1.2. RFC **6347**, 1–32 (2012)
24. Tarakji, K., et al.: P577 Early experience with the first pacemakers to directly connect with smart devices for remote monitoring. Eur. Heart J. **40**(Supplement_1) (2019). https://doi.org/10.1093/eurheartj/ehz747.0188, ehz747.0188
25. Varma, N., et al.: The relationship between level of adherence to automatic wireless remote monitoring and survival in pacemaker and defibrillator patients. J. Am. Coll. Cardiol. **65**(24), 2601–2610 (2015). https://doi.org/10.1016/j.jacc.2015.04.033