Your Mobility can be Injurious to Your Health: Analyzing Pervasive Health Monitoring Systems under Dynamic Context Changes

Ayan Banerjee* *Student Member, IEEE* and Sandeep K. S. Gupta* *Senior Member, IEEE.*

***Abstract*—The advent of smart phones has enabled health care anywhere and anytime. With pervasive health care, a person can perform necessary day to day tasks while his health is recorded, controlled, and processed continuously using on body sensors and actuators to capture abnormalities, trends, and causes. In such a scenario, seamless operation of the pervasive health management system (PHMS), given dynamic changes in the context induced by mobility of the user is of utmost importance. Such context changes dynamically affect many aspects such as the processing requirements of the health management application, the available energy sources, the interaction between a medical device with the human body. For social acceptability of PHMSes, they have to be tested and verified for their safe, energy sustainable (long term) and reliable operation under such dynamically changing environment. This paper proposes a novel technique to analyze PHMSes under mobility induced dynamic changes in the context and constant interaction of the medical device with the human body. Results show that human mobility induced context changes can cause unsafe conditions such as drug overdose.**

I. Introduction

With the aging population, those older than 65 expected to double by 2050 [14], and increased life expectancy the need for health care "anywhere" and "anytime" is imminent. Embedded computing devices such as smart phones, iPads, and other personal electronic devices are being increasingly used to support health related applications such as tracking calorie intake [6], weight control [7], and pulse oximetry [2]. Trends indicate a fast progression towards pervasive health monitoring systems (PHMSes) (Figure 1), where sensors are deployed on the human body to monitor environmental and physiological signals; a smart phone acts as a computation and communication hub, or base station, for these sensors - it gathers sensor data and processes them to detect contexts, such as location. activities, or health emergencies, and it may upload information to a cloud to archive a history of user's health. With the smart phone being recently considered as a medical device [16], PHMSes are predicted to become critical infrastructures to support health and wellbeing of the populace.

The operation of a PHMS is characterized by dynamic changes in the user context [1] e.g., change in user location from home to outdoors for a jog or to a hospital in case of medical emergency (Figure 1). Such changes in context affect the operation of every subcomponent of a PHMS. To illustrate this claim let us consider the example of the Ayushman

* The authors are affiliated with School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Email: {abanerj3,sandeep.gupta}@asu.edu

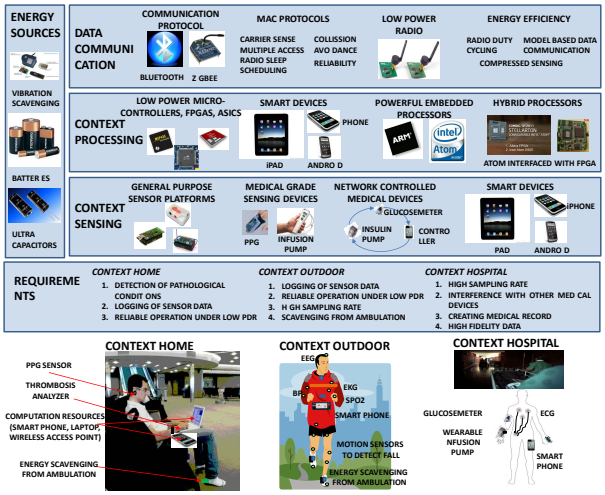

Fig. 1. **PHMS under different contexts with varied requirements and several options for a designer. Choosing the right set of alternatives to match the system requirements is a challenging problem.**

PHMS [20] shown in Figure 1 developed at the IMPACT Lab at ASU. The Ayushman PHMS has five subcomponents: 1) context sensing, which senses environmental and physiological signals to facilitate the detection of user context, 2) context processing, which processes the sensed signals to determine user context and perform necessary computations for the PHMS services, 3) data communication, which deals with the transfer of information from the sensors via the base station to the server, 4) energy sources, which determines the technology for energy supply to the sensors, base station, and server, and 5) last but not the least, the human body with which the PHMS devices interact either through sensing and actuation or through harmful side effects of operation such thermal harms.

In case of the network controlled infusion pump in the Ayushman PHMS, the controller sends control inputs to infusion pump over the wireless channel to maintain the analgesic drug concentration to a particular level. The controller gets feedback from a pulse oximeter on the human body on the blood oxygen level. The pump should stop infusing immediately when the blood oxygen level falls below a certain level to prevent respiratory distress [23]. Since the wireless channel is prone to errors, the packets containing control information can get corrupted or dropped at random. If control informations do not reach the infusion pump the pump maintains the previous control information for a preset time and then shuts down. Further, if the controller does not obtain an accurate

estimation of the blood oxygen level it can cause unstable or oscillatory infusion rates, which may be harmful for the user. To this effect, the communication protocol is modified to consider dynamic power control for avoiding packet drop. The design of the technique considered a home environment and modulates radio power to increase the packet delivery rate (PDR) from 0.6 to an acceptable level of 0.8. However, if the user now moves from home to an outdoor open space such as the balcony, the PDR drops drastically [12], and the dynamic power control may not increase the PDR to the acceptable level. If this happens too frequently - a property governed by the user's mobility model - several control packets may be dropped, which can lead to overshoot, undershoot, or oscillations in the analgesic concentration in the blood as result causing pathological conditions such as respiratory distress. Hence, in such a scenario the user's mobility pattern may be unsafe for his health! The operation of a PHMS involves such dynamic context driven interaction between the embedded computing device and the environment including human physiology.

Further, contexts such as onset of epilepsy [19] can trigger excessive computation in the smart phone, which may cause it to be thermally unsafe. Dynamic changes in context can also be beneficial when with a change in location context opportunities to scavenge energy comes forth e.g., sunlight.

Under such varied effects of dynamic contextual changes a PHMS design should be pre-verified to meet their operational requirements to be socially acceptable. It is of utmost importance to prove that a PHMS is - a) *truly pervasive*, i.e., not obtrusive to the user, easily wearable, and does not restrict the mobility of the user, b) *safe*, i.e., the sensors and the smart phones are risk free and safe from faulty operation, harmful thermal, and electromagnetic effects on the human body, and c) *sustainable*, i.e., capable of providing uninterrupted and long term operation. Rigorous analysis should be conducted before any clinical study to minimize any harm to the users. Further, any analysis technique will have to consider the dynamic context driven interaction of the PHMS devices with the human body.

This paper provides a systematic non-invasive analysis methodology for PHMS under dynamic context changes and continuous interaction of sensors, actuators, and smart phones with the ever changing environment.

A. Approach

In this paper, we take a model based engineering (MBE) approach for PHMSes. Such an approach, enables non-invasive analysis of PHMSes, where experimentation is not required to obtain an early feedback on the design. Such an approach is also useful to analyze the feasibility of the design at an initial stage. The proposed methodology has four phases (Figure 2):

1) *Specification phase:* In this phase, the designer specifies the *ContextFSM*, denoting the different contexts and the events that cause changes in the context. For each state in the *ContextFSM*, the designer specifies the architecture of the PHMS hardware and software configuration. Further, for each state there are different sets of requirements that the PHMS should satisfy. In addition, the user also specifies the model of the physical process of the environment and the human body. This phase also allows the specification of the cooperation-operation of the computing system with the human body through information and control message exchange (Section III).
2) *Profiling Phase*: In this phase, we perform experimentation on the PHMS hardware devices to determine properties such as power dissipation and thermal hot spots, essential for the safety and sustainability analysis (Section IV).
3) *Modeling Phase*: The information gathered from the profiling phase are then used to build models of the PHMS for each context. The model building process is hierarchical wherein the PHMS is represented as a collection of models for the sensors and the smart phone (Section V).
4) *Analysis & Design Phase*: The models developed in the previous phase are then used to perform mathematical analysis on the system properties to check conformance with the requirements (Section VI).

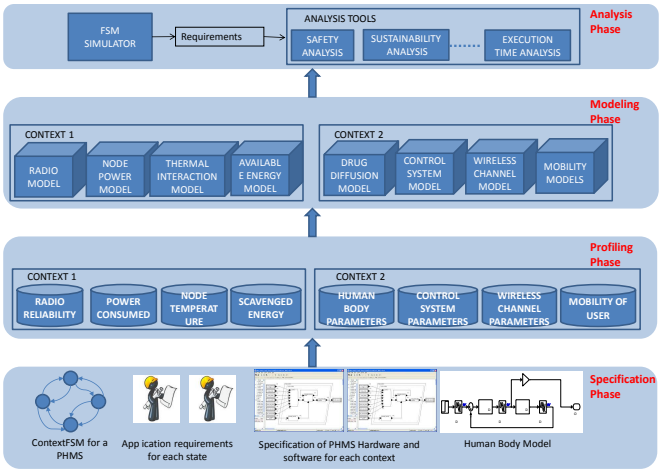

Fig. 2. **Approach for analysis of PHMS under dynamic context changes.**

MBE has been widely used for embedded systems. A large number of tools are available that model and analyze hardware of computing systems such as Pspice [18] and AADL (http://www.aadl.info/), and application software such as UML (http://www.uml.org/) and Petrinets [5]. The authors in [22] have proposed a tool called *ANDES* that introduces the concept of MBE in WSNs to ensure accuracy and low latency of WSN operations. The MBE approach is also used to study the behavior of physical systems through tools such as SysML (http://www.sysml.org/), Simulink (http://www.mathworks.com/), and Flovent (http://www.mentor.com/). However, none of these tools can model dynamic context and their effect on the interaction of the PHMS devices with the environment. Further, context awareness in embedded computing environments has been considered in detail [24], however, the cooperation of the embedded devices with their environment is limited to context sensing. This paper considers the Ayushman PHMS under three contexts: a) in a home environment, b) out for a jog, and c) in a hospital under medical supervision and shows

TABLE I
PHMS CONFIGURATIONS FOR DIFFERENT CONTEXTS IN AYUSHMAN (EXPLAINED IN MORE DETAIL IN THE APPENDIX).

Context	Requirements	Hardware Config.	Software Config.	Energy Source	Radio Protocol	Human Body
Home	Thermal safety, low energy consumption, and detection of arrhythmia within 5 seconds of onset	Shimmer ECG, PPG, and GSR sensors, TelosB motes for temperature and humidity, Intel Atom based smart phones with 6 sleep and 8 frequency states	Ayushman Workload shown in Figure 3, with sensing, communication and pairwise secure key agreement using physiological signals (PKA)	Batteries	Bluetooth and ZigBEE, with model based communication.	Thermal dynamics
Roaming	Reliable data communication under high wireless channel error, battery less operation for 6 hrs	Same as Home Context	Radio sleep scheduling, processor duty cycling, frequency control	Scavenge from body heat, respiration, ambulation, and sunlight	Retransmission and dynamic power control to ensure reliable data communication	Model of the intermittent behavior of scavenging sources.
Hospital	High fidelity data, thermal and drug overdose safety	Medical grade infusion pumps, pulse oximeters	Infusion control algorithm	Batteries or mains	Bluetooth, WiFi, ZigBEE, wired	Drug diffusion dynamics

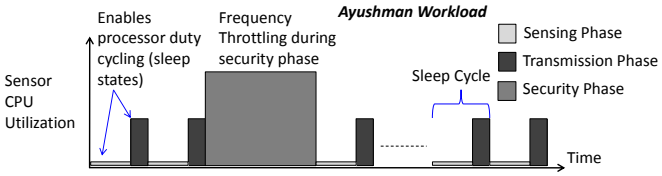

Fig. 3. **Ayushman workload.**

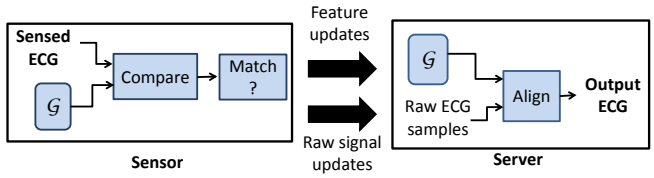

Fig. 4. **Model based data communication, which achieves 42:1 compression for ECG [10] and 300:1 for PPG [11].**

the usage of the tool to model and analyze context driven interactions and test compliance to requirements. In summary, the main contributions of this paper are:

- development of an integrated specification logic for PHMS, which enables specification of dynamic user context changes and the cooperation of the computing system with the user environment.
- integrating models of computation and human physiology to develop analysis algorithms for requirements such as safety, and sustainability,
- providing a comprehensive case study showing the usage of the proposed methodology on the Ayushman PHMS.

Using the infusion pump example we show how the mobility pattern of a user can harm the drug diffusion safety. Thermal safety analysis shows how context triggered health emergency detection algorithms can cause harmful thermal effects. Finally, we show how the mobility of an user can beneficially affect the sustainability of the PHMS. We use industry standard Architecture Analysis and Design Language (AADL) (www.aadl.info) to implement the specification phase and the OSATE JAVA software platform to develop the analysis phase. AADL is a widely used language in the industry and academia to specify system behavior models of wireless sensor networks [22] and analyze network related performances (throughput, packet drop ratio etc). Further, AADL allows extensions to the language by enabling introduction of new language constructs as language *annex*.

II. AYUSHMAN PHMS

Ayushman is a smart health infrastructure developed in the IMPACT Lab for privacy ensured continuous health monitoring of ambulatory individuals. It has a multi-tier architecture enabling management of sensors, secure storage and dissemination of data, access control of user health history, query processing, service discovery and context processing. At its core is a body sensor network (BSN) consisting of a number of physiological as well as environmental sensors such as photoplethysmogram, electrocardiogram, temperature, and humidity sensors and a smart phone serving as the computation and communication hub. In Ayushman we consider three different contexts, which vary in their hardware software configurations, communication protocols, power management techniques, and energy sources. These contexts are summarized in the Table I, and are discussed in more detail in the Appendix (available online at http://impact.asu.edu/publication/PercomBanerjee.pdf).

A. Context changes

Context changes occur due to random events triggered by: 1) user mobility, modeled using mobility models such as random or Levy walk [15] and Markovian models, 2) emergency events such as detection of fall, epileptic seizure, arrhythmia, and 3) user settings, such as activation and deactivation.

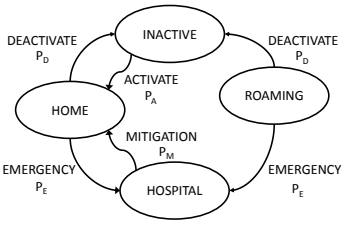

Fig. 5. **Finite State Automata representation of contexts and context changes *ContextFSM*.**

The different contexts can be represented as states in the *ContextFSM* and the events can cause state transitions. The events are assumed to be random with an associated probability distribution. The *ContextFSM* for the Ayushman PHMS is shown in Figure 5.

III. Specification Phase

The specification of a PHMS is done using the industry standard AADL language (www.aadl.info). AADL is a hierarchical model specification tool that provides constructs dedicated to modeling embedded software and hardware. However, AADL inherently does not support specification of context and context transitions and physical dynamics of the human body. We use the behavioral annex to specify context as states and context changes as a finite state automata, *ContextFSM*. Further, we extend AADL to incorporate specification of complex physical processes as a series of differential equations through the development of annex (language extensions). The AADL specifications shown in this paper denotes the AADL specific constructs in bold. A PHMS specification has five subcomponents:

PHMS specification: It has three different contexts and their dynamic transition logic, AADL Spec 1 (*PervasiveHealthMonitoringSystem*). For each context, there is a different hardware and software implementation indicated by the subcomponents in the AADL Spec 1, and are specified using the **system implementation** construct. Each context is specified using the **mode** construct, and the context transitions are specified as shown in the AADL Spec 1. The transitions take place on occurrence of events, which are specified in the **features** section of the specification. The events are generated from context sensors specified using the *ContextSensor* **system** component. In this component, mobility models can be specified, which can be later used to generate random events that cause state transitions. In this particular example, there are *four* states - home, roaming, hospital, and inactive, and six events - RoamingActive, AtHome, Emergency, Mitigate, Activate and DeActivate. In each context, the PHMS consists of context sensor nodes, energy source, the human body specification, and specification of the coordination between the devices and the human body.

Context sensor node: The context sensor node specification, AADL Spec 2 (*PHMSnode.ContextSensorNode*1), has the processor, application algorithm, and the radio. The processor (*Processor.Atom*) power for different sleep states and the idle power consumption at various operating frequencies are obtained from the experiments performed in the profiling phase. The busy power consumption of the processor for different operating frequencies are dependent on the application threads, modeled in the *Application.Ayushman* subcomponent. The model of the application workload consists of three main threads: 1) Sensing, 2) Data Transmission and 3) PKA Execution, each of which models the power consumption and thread execution time for different frequency of operation of the processor. The *Radio.CC*2420 subcomponent models the power consumption of the radio for three different modes of operation: 1) Radio transmission, 2) Reception and 3) Radio turned off. The model based communication algorithm can be specified as a state machine using the **mode** construct, where there will be three modes: 1) model match, when the model matching thread will be executed, 2) feature update, when the thread will compute features from the signal and send it back to the base station, and 3) raw signal update, when the sensor will just blindly send sample by sample data. For each mode

AADL Spec 1 PHMS context specification.

```
system PervasiveHealthMonitoringSystem
  features
  RoamingActive: in out event port;
  AtHome: in out event port;
  DeActivate: in out event port;
  Activate: in out event port;
  Emergency: in out event port;
  Mitigation: in out event port;
end PervasiveHealthMonitoringSystem;

system ContextSensor
  subcomponents
  process RandomWayPointModel
end ContextSensor;

system implementation PHMS.HomeContext
  properties
  TemperatureThreshold::313 K
  PDRThreshold::0.8
  NoOfNodesToSustain::7
  subcomponents
  S1: system PHMSNode.ContextSensorNode1;
  S1: system ScavengingSource.BodyHeat;
  S1: system HumanBody.skin;
end PHMS.HomeContext;

system implementation PHMS.OutdoorContext
     ...
end PHMS.OutdoorContext;

system implementation PHMS.HospitalContext
     ...
end PHMS.HospitalContext;

system implementation PervasiveHealthMonitoringSystem.imp
  subcomponents
  P0: system PervasiveHealthMonitoringSystem;
  P1: system PHMS.HomeContext in modes Home;
  P2: system PHMS.OutdoorContext in modes Outdoor;
  P3: system PHMS.HospitalContext in modes Hospital;
  R1: system ContextSensor.RoamingActive;
  H1: system ContextSensor.AtHome;
  E1: system ContextSensor.Emergency;
  M1: system ContextSensor.Mitigation;
  UA: system UserInput.Activate;
  UA: system UserInput.DeActivate;
  connections
  event port R1.SensorOutput → P0.RoamingActive;
  event port H1.SensorOutput → P0.AtHome;
  event port E1.SensorOutput → P0.Emergency;
  event port M1.SensorOutput → P0.Mitigation;
  event port UA.UserEvent → P0.Activate;
  event port UD.SensorOutput → P0.DeActive;
  modes
  Home: initial mode;
  Roaming: mode;
  Inactive: mode;
  Hospital: mode;
  Home: -[ P0.RoamingActive ] → Roaming;
  Roaming: -[ P0.AtHome ] → Home;
  Home: -[ P0.DeActivate ] → Inactive;
  Roaming: -[ P0.DeActivate ] → Inactive;
  Inactive: -[ P0.Activate ] → Home;
  Home: -[ P0.Emergency ] → Hospital;
  Roaming: -[ P0.Emergency ] → Hospital;
  Hospital: -[ P0.Mitigate ] → Home;
end PervasiveHealthMonitoringSystem.imp;
```

the power consumption and data sent properties will capture the compression obtained for each mode. The wireless channel properties for the radio can be specified using the properties construct, which can change for different modes.

Energy scavenging source: The available scavenged energy can be modeled as a property of the system implementation as shown in the AADL Spec 2. Further, to calculate the available energy, established models such as ANNOVA for ambulation [8] can be specified and executed in the analysis.

AADL Spec 2 PHMS sensor node specification.

```
system implementation PHMSnode.ContextSensorNode1
  subcomponents
  S1: system Processor.Atom;
  S2: system Radio.CC2420;
  A1: process Application.Ayushman;
     ...
end PHMSnode.sensorNode1;

system implementation Processor.Atom
  modes
  SleepState1 : mode;
  Frequency1 : mode;
     ...
  properties
  IdlePower => 0.160 W in modes SleepState1;
  Temperature => 43 °C in modes Frequency1 ;
     ...
end Processor.Atom

system implementation Radio.CC2420
  modes
  RadioOnTx : mode;
     ...
  properties
  Power => 0.058 W in modes RadioOnTx;
     ...
end Radio.CC2420

system implementation ScavengingSource.Ambulation
  properties
  Model => ANNOVA;
  Available Power => 0.058 W;
     ...
end ScavengingSource.Ambulation

process implementation Application.Ayushman
  subcomponents
  T1: thread Sensing;
  T2: thread PKA;
  T2: thread Transmission;
end Application.Ayushman

thread implementation Sensing
  modes
  Frequency1 : mode;
     ...
  properties
  Power => 0.030 W in modes Frequency1;
  ExecutionTime => 5 s in modes Frequency1;
     ...
end Sensing
```

Human body specification: Modeling of the human body is complex and AADL is not geared towards it. Property constructs can be used to specify the physical properties (thermal conductance, specific heat, blood perfusion as shown in AADL Spec 3). However, the complex physical dynamics of human body that controls its thermal behavior requires specification of differential equations in AADL which it does not support. In this regard, an annex (*CPS _annex*) was developed which extends AADL to incorporate specification of differential (both partial and total) equations in the model itself. Specific constructs for denoting differentials were developed in the annex. See Appendix for annex implementation details.

Specification of coordinated operation: The coordinated operation results in changes in the complex physical processes with events occurring in the computing domain. In the infusion pump example, the diffusion of drug is governed by the pharmacokinetic (PKA) process [23], which can be modeled as a set of differential equations.

AADL Spec 3 Human Body Specification.

```
system implementation HumanBody.skin
  properties
  SpecificHeat => 1.6 J/(Kg.K);...
  annex
  Del1<Temperature><Time> =
  K(Pdel2<Temperature><x>+Pdel2<Temperature><y> +
  Pdel2<Temperature><z>) + ...
end HumanBody.skin;
```

AADL Spec 4 Infusion Control Algorithm Specification.

```
system implementation NetworkControlledDevice.InfusionPump
  properties
  DiffusionCoefficient (D) => 1.6 J/(Kg.K);
     ...
  annex CPS_annex {**
  PDE1: Del1<DrugConc><Time> =
  D(Pdel2<DrugConc><x>+Pdel2<DrugConc><y>) +
  InfusionRate + ...
  PDE2: Del1<DrugConc2><Time> =
  D(Pdel2<DrugConc2><x>+Pdel2<DrugConc2><y>) + ...
**}
  annex behavior_specification {**
  states
     s0: initial state;
     s1, s2, s3: state;
  transitions
  s0: -[ Bolus ?] → s1 {PDE1;};
  s1: -[ OverShoot ? {PDE1;}] → s2 {PDE2;}; **}
end NetworkControlledDevice.InfusionPump;
```

However, the equations change with the change in controller's state. The controller algorithm takes the drug concentration predicted by the PKA process as input and varies the infusion rate to keep the drug concentration at a given level. Such an algorithm can be represented using a state machine, which captures both the computing and physical behavior of the infusion pump. However, the **mode** construct cannot be used to specify states in this case, since there is no provision to specify equations for a given state and transitions only depend on random events and not on variation of system parameters. Instead we use a combination of the *behavior _annex* and the *CPS _annex* in AADL to specify the control algorithm as a hybrid automata, AADL Spec 4 (*NetworkControlledDevice.InfusionPump*). As shown in the specification, we can specify the partial differential equations using *CPS _annex*, *PDE*1 and *PDE*2 and associate them with states *s*1 and *s*2 in the *behavior _annex*. Further, the events in the *behavior _annex* can occur when a variable in the implementation goes over threshold (*Overshoot* event).

TABLE II
POWER PROFILING OF SENSORS (TELOSB, IMOTE, BSN NODE V3, SHIMMER).

Tasks	Consumed Power (mW)	Execution Time (ms)
Mean, STD	5, 162, 6.7, 6.73	230, 220, 207, 200
Fourier Transform	5.1, 162, 6.5, 6.66	435, 102, 425, 415
Peak Detection	5.6, 156.6, 6.8, 6.6	100, 160, 90, 88

IV. PROFILING PHASE

The available energy profiles of the scavenging sources are already obtained from [13] while the profiling of human body for its thermal properties are obtained from previous literature. Thus, we derive the power and thermal profiles of the PHMS node under the Ayushman workload.

A. Power Profiling

For power measurement of the Atom processor we use the Mobile Intel 945 GSE (GMCH) chipset. The power measurement setup provides the board power consumption, which includes the CPU power as well as power for driving the chipset and other associated components. In order to isolate the CPU power consumption during Ayushman execution, we first measured the idle power of the board for each throttling mode by allowing the CPU to run idle for three minutes (for stabilizing the watt meter). Then the Ayushman workload is executed to measure the average platform power. The difference between the two power values gives the power consumed by the processor during the execution of the workload, which is shown in Table III for different throttling modes.

TABLE III
ATOM POWER USAGE FOR PKA COMPUTATION IN AYUSHMAN.

Percent Throttling	Power Consumption W
0	0.191
13	0.1864
25, 37	0.17
50, 62, 75, 87	0.167

The power consumption of the TelosB motes were experimentally obtained by running the BSNBench benchmarking suite [9]. The benchmarking suite has specific tasks for obtaining power consumption due to computation, sensing, and communication. The sensing and computation power consumption is listed in Table II for benchmark signal processing applications such as Fourier transform, and peak detection. The power consumption of the Chipcon radio was measured during transmitting packets at a bit rate of 250 kbps, standard for a sensor node (www.xbow.com). The current consumption of the CC2420 radio used in the PHMS was measured to be 18.41 mA during transmission and 19.20 mA during reception. Considering the operating voltage to be 3 V the maximum power consumption of the radio is 58 mW.

B. Thermal Profiling

The thermal effects of the TelosB sensors are negligible and hence are not considered in the analysis. However, the Intel atom based smart phone dissipates a greater amount of power and hence can cause thermal harm to the human skin. The temperature of the Intel Atom processor is measured using the on board thermal sensor of the Mobile Intel 945 GSE development platform (GMCH). The N270 processor has dedicated MSR registers (http://download.intel.com/design/processor/datashts/320032.pdf), which stores the digital thermometer reading. The Atom processor was first kept at sleep state (C6) to allow for the core temperature to stabilize to a low value. Then the processor was brought back to normal state and the Ayushman application was run at different throttling modes of the processor. Table IV shows the peak operating temperature attained by the Atom processor during Ayushman execution at different throttling modes.

TABLE IV
THERMAL PROFILE DATA FOR ATOM PROCESSOR.

Percent Throttling	Operating Temperature °C	Skin Temperature °C after 24 Hr of operation
0, 13, 25	43	39.4365
37, 50	42	39.3325
62, 75	41	39.2295
87	39	39.0264

V. MODELING PHASE

In this section, we discuss representative examples of the models developed from AADL specifications.

A. Power model of PHMS

Power consumption of hardware components (processor, radio) for execution of different threads ($Thread_power$) and their timings ($Thread_Time$) are extracted from the AADL meta-data. We contend that during the period of sensing $t_s = 5$ secs, the micro controller is in idle state, where it consumes P_{idle} amount of power (≈ 1 mW in TelosB motes). For a PHMS with n nodes the sensing process can be performed in parallel by all sensors. After each sensing period the sensed data is transferred to the base station. During this transmission period t_{Tx} the processor is in idle state, consuming P_{idle} amount of power (approximately for 0.39 secs to transmit five seconds of 32 bit data values 60 Hz sampling rate and a transfer rate of 24 Kbps [21]). The radio transmitter will also be active during this period ($P_{radio} \approx 58$ mW being its power consumption). In a 24 hr period there will be x number of sense and transmit periods (sleep cycles) for each sensor in the PHMS, with a duration of $(t_s + t_{Tx})$ secs each. Further, in a single day of operation of Ayushman the PHMS nodes under go pairwise PKA execution to maintain the freshness of the encryption key among two nodes. During this execution of PKA the processor should be in active state consuming P_{PKA} amount of power for the duration of execution of the PKA algorithm t_{PKA}. The value of P_{PKA} is around 10 mW and t_{PKA} is around 1 sec as obtained from actual measurements averaged over all the commercially available platforms. Further, during the transfer of the vault ($t_{Vault} = 6.75$ secs [21]), the radio is active. Thus total energy consumption is given by Equation 1:

$$\text{Total PHMS Energy} = \text{Sensing Energy} + \text{Data Transmission Energy} + \text{PKA computation energy} + \text{vault transfer energy}$$

$$\Rightarrow E_{PHMS} = nx(t_s)P_{idle} + nxt_{Tx}(P_{radio} + P_{idle}) + t_{PKA}P_{PKA}(\frac{n(n-1)}{2}) + t_{Vault}(P_{idle} + P_{radio})(\frac{n(n-1)}{2}) \quad (1)$$

x is the number of sleep cycle to be sustained in 24 hrs. The parameters of this model are experimentally derived and hence are applicable in practice.

B. Thermal model of PHMS

The operating temperature ($Operating_Temperature$) of the PHMS node, power consumption ($Component_Power$) of each component, human skin properties ($Skin_Properties$) and the distances of the components from the skin ($Skin_Distance$) are obtained from the metadata information after parsing the AADL specification. These parameters are used in the estimation of the temperature rise in the human body parts, which is governed by the following physical phenomenon: 1) radiative heat transfer from the Atom processor which depends on the operating temperature of the processor (modeled using the Stefan's Law), 2) conductive heat transfer from the processor, 3) electro-magnetic radiation absorption by the body part (modeled by calculating the Specific Absorption Rate (SAR) [17]) and 4) convective heat extraction by blood. These physical processes are combined in a partial differential equation known as Penne's bioheat equation which gives the temperature variation of the human body part over space and time (Equation 2):

$$\rho C_p \frac{dT}{dt} = K \bigtriangledown^2 T - b(T - T_b) + \rho \mathrm{SAR} + P_c + A \times \sigma(T_r^4 - T^4) \quad (2)$$

where ρ is the mass density, C_p is the specific heat, K is the thermal conductance and T is the temperature of the body part, P_c is the power generated by the processor, A is the surface area of contact with the device, T_r is its temperature, b is the blood perfusion constant, T_b is the blood temperature, and SAR is the specific absorption rate of the body part for electromagnetic radiation. The thermal model is verified against experimentation in our previous work [3].

C. Model of Infusion Control

The control algorithm of the pump considers an initial infusion rate of x_0. The control algorithm discretizes time and queries the pharmacokinetic model after each discretized step δt for an estimation of the drug concentration. It then either increases the infusion rate by δx or decreases it according to a linear approximation of the diffusion process. The initial infusion rate, time discretization step and the infusion increment step are the variables of the control system which can be tuned to obtain different behaviors. Further, the infusion pump can get random *bolus* requests (a step rise in infusion rate). The magnitude of the bolus is also a variable of the system. The pharmacokinetic model expresses the drug diffusion process as a set of multi-variable linear differential equations in state space form (Equation 3):

$$\dot{y_1} = A_1 y_1 + B_1 z_2 + E_1 u(t - T_i), z_1 = C_1 y_1(t - T_p) \quad (3)$$
$$\dot{y_2} = A_2 y_2 + B_2 z_1, z_2 = C_2 y_2(t - T_r)$$

Here y_1 and y_2 are the state space variables of the equation. A_1, A_2, B_1, B_2, C_1, C_2 and E_1 are constants. z_1 is the drug concentration in the blood while z_2 is the drug concentration of the tissue. The initial infusion rate $u = x_0$ is the input to the model and the output is the drug concentration in the blood. The differential equations in the model are time-variant. This is because they consider time delays related to the infusion input (T_i), cardio-pulmonary transport delay T_p and the arterial, capillary and venous transport delays T_r. The time-variant nature of the physical process comes from the consideration of the transport delays. This model is verified against experimentation and proved to hold in practice [23].

VI. PHMS Analysis

The AADL specification of the PHMS is first parsed to hierarchical XML based metadata. This XML metadata is used to extract information for analysis. The methodology for analyzing a PHMS for requirements verification under the dynamically changing environment is shown in Figure 6. The first step in the analysis procedure is to generate context transition test cases. In this step, a random sequence of events are generated according to context models such as mobility models, arrhythmia occurrence probability, and bolus request frequency. These events are classified into event types (ET_i) and are appended with an estimate of the time before next event ($TBNE_i$) and arranged into an event queue. The *ContextFSM* is then simulated starting from the initial state in accordance with the events. In each state, the context specific PHMS is parsed to obtain the requirements and analysis parameters. Depending upon the requirements different analysis plug-ins are employed to perform the simulation of the PHMS model. Further, domain specific tools such as Matlab can also be used to analyze the PHMS model. The execution of the appropriate plug-in for the correct analysis parameters and checking the compliance with the requirements is performed by the analysis execution unit (Figure 6). The output of the analysis execution engine is the compliance results. This analysis methodology is developed in the OSATE simulation environment, a JAVA based AADL specific software development platform. We next show the usage of the analysis framework to analyze the dynamic context driven interactions between the PHMS devices and the human body.

A. Effect of context change on medical control systems

We consider the infusion pump example discussed in Section I and show the usage of our analysis framework. The PHMS is in a hospital context. However, the patient wants to move around in the hospital and goes to the balcony to enjoy the view outside. This will trigger a context change in the PHMS and the system state will transit from hospital to outdoor. In such a scenario, specifically the wireless channel properties will change resulting in a different packet delivery ratio (PDR) for the radio communication. Since the infusion pump is controlled through the wireless channel by the controller, change in the PDR may cause a drop in communication quality between the controller and the pump. Low PDR may lead to packet loss from the controller to the infusion pump. This may cause delay or loss of control inputs to the pump. In the analysis framework, two different mobility models, random and Levy walk [15], were used to simulate the context change.

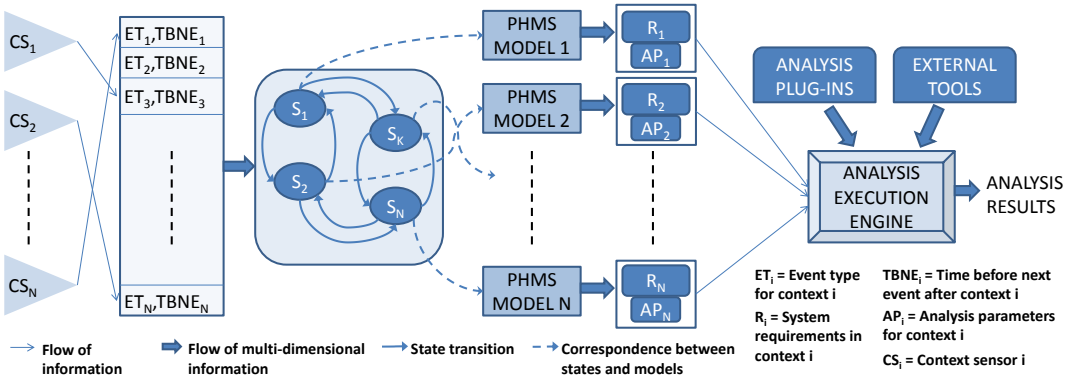

Fig. 6. **PHMS analysis methodology.**

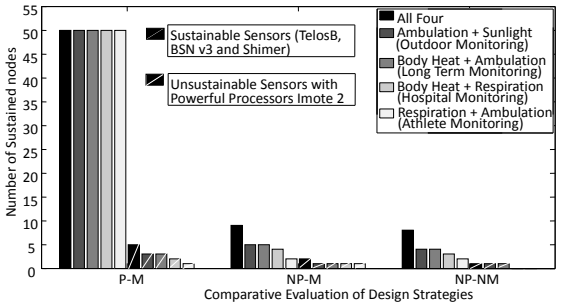

Fig. 7. **Sustainability Analysis Results.**

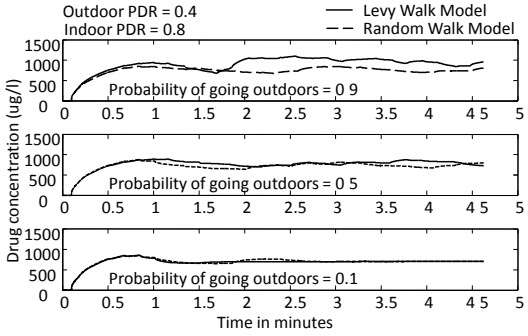

Fig. 8. **Analysis of infusion pump safety under dynamic context changes.**

The hospital region was divided into two parts: indoor (PDR = 0.8) and outdoor (PDR = 0.4). The contexts were simulated for 10 cases with probability of outdoor visits varying from 0.1 to 0.9. For each sequence of control inputs the control algorithm and the pharmacokinetic model were simulated in coordination. The results of the simulation is shown in Figure 8. The results show that a random mobility pattern is less harmful (causes lower overshoots in the average case) than a Levy walk pattern. This is because in the Levy walk pattern the patient is more inclined towards an outdoor visit. However, in random walk pattern the outdoor visits are more uniformly distributed. Such complex simulation of dynamic context changes and its effect on the medical device and human body coordination cannot be performed in contemporary simulation tools and is only facilitated by our methodology.

B. Context driven safety violation:

In this example, we consider the thermal impact of the Intel Atom based smart phone on the human body. We consider two contexts: 1) the smart phone is idle, and 2) when it receives data from the ECG sensor and starts online processing of the signal to detect onset of epilepsy [19]. The epilepsy detection algorithm involves the computation of Fast Fourier Transform (FFT) of the signal and peak detection. Once the smart phone transits to the epilepsy detection context it stays in that context for 1000 seconds. The thermal map for the operation of the smart phone is shown in Figure 9. The results show that when the smart phone was in the idle state, it was thermally safe. However, continuous execution of the epilepsy detection algorithm is not safe since after 1000 seconds of operation the skin temperature reaches safety limits of 312 K.

TABLE V
AVAILABLE SCAVENGING POWER [13].

Scavenging Source	Available Power	Scavenge Time
Body Heat	0.1 - 0.15 W	24 Hrs
Ambulation	1.5 W	2 Hrs
Respiration	0.42 W	6 Hrs
Sun Light	0.1 W	3 Hrs

C. Intermittent energy availability:

In this example, we consider two contexts: Home and Outdoor. The user is wearing an ECG, PPG, and accelerometer sensor to monitor his physiological health and exercise performance. Energy scavenging units harvest energy from body heat, respiration, sunlight, and ambulation. When at home energy can only be scavenged from respiration. However, in outdoor environments, energy can be scavenged from all four. Table V gives the available power from the scavenging sources and also the time duration for which they can perform the scavenging operation as discussed in [13].

We developed a sustainability analysis plug-in that used the power model of the context sensor and matched with the scavenging sources to compute the number of days a sensor can be sustained. We considered three combination of power management strategies: 1) no power management (NP-NM), 2) no processor level power management but with radio sleep scheduling (NP-M), and 3) with processor level power management and radio sleep scheduling (P-M). Figure 7 shows the time for which a PHMS node can be sustained using the different scavenging combinations design strategies. The analysis classified the sensors in the Ayushman BSN into

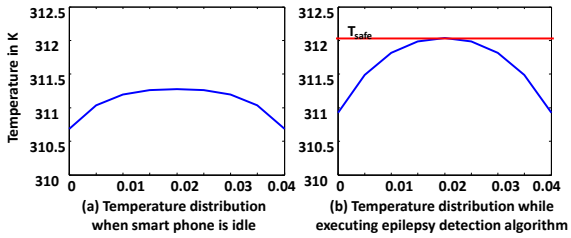

Fig. 9. **Skin thermal map for smart phone operation.**

two classes - a) sustainable sensors, such as TelosB, BSN node v3, and Shimmer, and b) unsustainable sensors, such as Imote 2, which have powerful processors (Intel XScale). In the subsequent experiments, we show how different mobility patterns of a patient can affect the sustainability of the sensor nodes. We find that when we consider context changes the time for which the nodes can be sustained decreases to 12.27 hrs on an average, due to intermittent nature of energy availability. Further, we observed that our analysis methodology could simulate the decrease in time before recharge with reduction in the outdoor excursion frequency. Further, if we employ the model based data communication it increases the sustainability of the sensors by a factor of 42 in case of ECG [10] and 300 in case of PPG [11]. However, context driven changes in the wireless channel characteristics causes higher packet loss resulting in loss in accuracy of reported data. Retransmissions and dynamic power control mechanisms can reduce the sustainability by a factor of nearly 2.

VII. Conclusions

In this paper, we have demonstrated a specification and analysis methodology for PHMSes, which takes into account the dynamic context changes and the effects on the computing infrastructure, their environment, and the interaction between them. The contexts are specified using a finite state machine, *ContextFSM*, which for each state considers a different PHMS configuration. The AADL language is enhanced with ability to specify physiological processes, which enables modeling the interaction between the medical devices and the human body. The operation of the methodology is demonstrated by analyzing the Ayushman PHMS. Results show that user mobility patterns can cause drug overdose or but can also help sustain sensors longer with more scavenged energy. Context changes may result in increased communication resulting in higher heat dissipation. Future work may include online formal verification of PHMS models using time bounded short run behavior analysis [4]. An appendix is included in the extended version of the paper, which describes the different PHMS configurations and the implementation of the annex (http://impact.asu.edu/publication/PercomBanerjee.pdf).

Acknowledgments

This research was funded in part by NSF grants CNS-0831544 and IIS- 1116385, and Intel Corporation. Special thanks to the OSEL group in FDA for providing infusion pump models, Mary Murphy-Hoye of Intel for Atom based development kits, and the reviewers and the shepherd Dr. Behrooz Shirazi for their useful inputs to improve the paper.

References

[1] F. Adelstein, S. K. S. Gupta, G. Richard, and L. Schwiebert. *Fundamentals of Mobile and Pervasive Computing*. McGraw Hill, 2004.

[2] AppBrain. Instant heart rate. http://www.appbrain.com/app/instant-heart-rate/si.modula.android.instantheartrate.

[3] A. Banerjee, S. Kandula, T. Mukherjee, and S. K. S. Gupta. BANDAiDe: A tool for cyber-physical oriented analysis and design of body area networks and devices. *ACM Transactions on Embedded Computing Systems (TECS), Special issue on Wireless Health Systems*, 2010.

[4] L. Bu, Q. Wang, X. Chen, L. Wang, T. Zhang, J. Zhao, and X. Li. Toward online hybrid systems model checking of cyber-physical systems' time-bounded short-run behavior. *SIGBED Rev.*, 8:7–10, June 2011.

[5] G.-M. Elena and M. José. Argospe: Model-based software performance engineering. In *ICATPN*, pages 401–410, 2006.

[6] iTunes. Calorie tracker. http://itunes.apple.com/us/app/calorie-tracker-livestrong.com/id295305241?mt=8.

[7] iTunes. Loose the belly. http://itunes.apple.com/us/app/lose-belly-weight-loss-for/id333182979?mt=8.

[8] S. Korla, R. A. Leon, I. N. Tansel, A. Yenilmez, A. Yapici, and M. Demetgul. Design and testing of an efficient and compact piezoelectric energy harvester. *Microelectron. J.*, 42:265–270, February 2011.

[9] S. Nabar, A. Banerjee, S. K. S. Gupta, and R. Poovendran. Evaluation of body sensor network platforms: a design space and benchmarking analysis. In *Wireless Health 2010*, WH '10, pages 118–127, New York, NY, USA, 2010. ACM.

[10] S. Nabar, A. Banerjee, S. K. S. Gupta, and R. Poovendran. GeM-REM: Generative model-driven resource efficient ecg monitoring in body sensor networks. In *Body Sensor Networks (BSN), 2011 International Conference on*, pages 1 –6, may 2011.

[11] S. Nabar, A. Banerjee, S. K. S. Gupta, and R. Poovendran. Resource-efficient and reliable long term wireless monitoring of the photoplethysmographic signal. In *Wireless Health (WH) accepted for publication, 2011 International Conference on*, pages 1 –6, October 2011.

[12] A. Natarajan, B. de Silva, K.-K. Yap, and M. Motani. To hop or not to hop: Network architecture for body sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks, SECON '09. 6th Annual IEEE Communications Society Conference on*, pages 1 –9, june.

[13] J. A. Paradiso. Energy scavenging for mobile and wireless electronics. *Pervasive Computing, IEEE*, 4(1):18–27, Jan.-March 2005.

[14] Population Reference Bureau. Aging population. http://www.prb.org/Publications/PopulationBulletins/2011/americas-aging-population.aspx.

[15] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong. On the levy-walk nature of human mobility. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 924 –932, april 2008.

[16] Singularity Health Law. FDA to regulate minor subset of smart phone medical device apps. http://gregpiche.typepad.com/blog/2011/09/fda-to-regulate-minor-subset-of-smartphone-medical-device-applications.html.

[17] Q. Tang, N. Tummala, S. K. S. Gupta, and L. Schwiebert. Communication scheduling to minimize thermal effects of implanted biosensor networks in homogeneous tissue. *Biomedical Engineering, IEEE Transactions on*, 52(7):1285–1294, July 2005.

[18] P. W. Tuinenga. *Spice: A Guide to Circuit Simulation and Analysis Using PSpice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1991.

[19] H. Veld and M. Ordelman. Context aware algorithm for epileptic seizure detection. In *Awareness deliverables*, 2005.

[20] K. Venkatasubramanian, G. Deng, T. Mukherjee, J. Quintero, V. Annamalai, and S. K. S. Gupta. Ayushman: A Wireless Sensor Network Based Health Monitoring Infrastructure and Testbed. In *Distributed Computing in Sensor Systems*, pages 406–407, July 2005.

[21] K. K. Venkatasubramanian, A. Banerjee, and S. K. S. Gupta. Green and sustainable cyber-physical security solutions for body area networks. In *BSN '09: Proceedings of the 2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks*, pages 240–245. IEEE Computer Society, 2009.

[22] P. Vibha, T. Yan, P. Jayachandran, Z. Li, S. H. Son, J. A. Stankovic, J. Hansson, and T. Abdelzaher. Andes: An analysis-based design tool for wireless sensor networks. In *Real-Time Systems Symposium, RTSS 2007. 28th IEEE International*, pages 203–213.

[23] D. Wada and D. Ward. The hybrid model: a new pharmacokinetic model for computer-controlled infusion pumps. *Biomedical Engineering, IEEE Transactions on*, 41(2):134 –142, feb. 1994.

[24] B. Wang, J. Bodily, and S. K. S. Gupta. Supporting persistent social groups in ubiquitous computing environments using context-aware ephemeral group service. In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on*, pages 287 – 296, march 2004.