Safety Assurance of Medical Cyber-Physical Systems using Hybrid Automata: A Case Study on Analgesic Infusion Pump*

Priyanka Bagade, Ayan Banerjee, and Sandeep K.S. Gupta
IMPACT Lab
Arizona State University, Tempe, Az
{pbagade,abanerj3,sandeep.gupta}@asu.edu

ABSTRACT

Interactions between the medical devices and the human body in Medical Cyber Physical Systems (MCPSes) are considered for verifying patient's safety. The discrete and continuous dynamics of an MCPS require a hybrid approach towards modeling and analysis. In this regard, hybrid automata is used to model analgesic infusion pumps, an exemplary MCPS application. Excursions of unsafe states in this model such as respiratory distress due to drug overdose, are analyzed by hybrid automata reachability analysis. However, given the time delayed dynamics of traditional reachability analysis using Zonotope approximations of states is not feasible. Thus, we propose a zero order hold approximation on the time delayed state variables and perform the reachability analysis on the resulting approximate model. We also provide a bound on the maximum error of the reachability analysis methodology.

Categories and Subject Descriptors

D.2.4 [**Software Verification**]: Formal methods

Keywords

Hybrid Automata, Reachability Analysis, Infusion Pump

1. INTRODUCTION

In recent years, technological advancement has enabled embedding computing units in a human body which monitor and control physiology as well as actuate critical life saving medical operations such as drug infusion. Such Medical Cyber-Physical Systems (MCPSes) are by definition safety critical [26] and guarantees the safety of the human body e.g. drug concentration should not go over a threshold for normal operation before marketing. This paper deals with the modeling, designing, and verification of such MCPSes for their greater social acceptability.

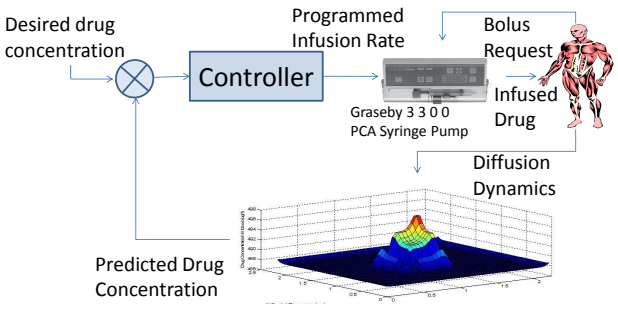

Figure 1: Network controlled infusion pump.

An important aspect of MCPSes is the seamless and complex interactions among the computing units and the human body, referred to as cyber-physical interactions. Such interactions can be often un-intentional and hazardous to the environment. For example, heat generated from a pulse oximeter can burn the human skin [13] or the chemotherapy drugs may kill normal cells apart from the cancer cells [17]. It is therefore imperative that an MCPS maintains such interactions within the desired limits to ensure its safe operations.

Experimental evaluation of MCPSes can be intrusive and hazardous when safety requirements are violated. As such, automated verification through model checking, similar to the verification of safety-critical systems [22], is required. To this effect, development of proper models to characterize the interactions in MCPSes is imperative for analyzing their safety. A Federal Networking and Information Technology Research & Development (NITRD) report has indeed identified the importance of new design abstractions for MCPSes [26]. This paper is a step towards that.

Research advances in the formal methods world has proposed hybrid automata (HA) [15] as a mathematical representation to describe the operation of complex control systems. Such a representation, allows one to characterize both continuous and discrete dynamics in a unified framework. This feature is particularly useful in formalizing and analyzing MCPSes. Figure 1 shows an analgesic infusion pump which is the example MCPS used in this paper. The infusion pump software includes a discrete control algorithm that samples the human physiology as feedback and decides on

*This work is partly funded by NSF projects CNS #0831544 and IIS #1116385. The authors are thankful to Madhurima Pore for helping in the error analysis simulations.

.

the future infusion rates to control a steady level of analgesic drug. Thus, on one hand, MCPSes such as analgesic infusion pumps are software controlled, where the software operation essentially implements certain discrete control logic. On the other hand, continuous dynamics govern their mechanical/electrical actuation and their (continuous) interaction with the patients and the environment. Intuitively HA are most capable of expressing the dual nature of MCPSes. The HA uses discrete states to model control operation while it uses continuous variables to model the dynamics. The values that the continuous variables assume at any given time is called the *state* of the HA. The set of all possible values of the state is called the continuous state space.

The principal advantage of modeling MCPSes using HA is that patient safety properties can be theoretically proven. However, the safety violations in the systems might occur due to random events which can not be captured by HA. Finite state machine model checking is needed to address these issues, but it does not represent continuous and discrete dynamics of MCPS. For the analgesic infusion pump, if the drug concentration increases beyond a certain prescribed value then it causes respiratory distress in the patient. Safety properties such as drug concentration should be below certain threshold, can be specified using simple relational operators in an HA model e.g., greater than or less than the threshold. Such relational specification of safety properties implies that certain subset of the continuous state space is unsafe. Given such an unsafe region, the reachability analysis of the HA can be used to check whether the safety properties are violated at any time. Reachability analysis outputs the possible values that the continuous states of the HA model of the MCPS can assume at any given time, also called the *reach set*. Intersection of the set of possible values with the unsafe set entails that there exists a time when the physical parameters of the MCPS assumed unsafe values. In such cases, it is clear that the MCPS design is unsafe.

Formalizing MCPSes as mathematical representations like HA models can thus: 1) allow manufacturers to reduce ambiguity and understanding in the safe design of these devices; 2) enable one to consider the safety of MCPSes under the context of device-human interaction; and 3) provides greater confidence for approving agencies such as Food and Drugs Administration on the marketability of the device.

The goal of this paper is to *model the cyber-physical interactions among the computing and physical entities in the MCPSes and prove patient safety properties.* The operating modes of the software are represented using discrete states while the human physiology is represented using differential equations. The transitions between states represents the evolution of device-human interaction over time. Specifically, we consider an analgesic infusion pump and model its discrete control algorithm using discrete states in hybrid automata and the drug diffusion dynamics using linear state space equations.

Reachability analysis of HA is an undecidable problem [15]. Researchers have considered an over-approximation of states to tractably compute the reachable states of HA. However, such approximation is only applicable to linear dynamics. Interestingly, the linear dynamics of the drug diffusion is time delayed. For time delayed systems there has been limited research on reachability analysis. Several authors have successfully attempted to solve the stability problem of hybrid systems with time delayed dynamics [21,28]. The stability problem only proves that the hybrid automata will reach a stage when the change in state variables will be within a given upper and lower bound. As long as the upper bound on error is an over-approximation of actual value (does not have to be accurate), it is guaranteed that the actual model is safe.

In this paper, we make a *zero order hold approximation* for time delayed variables. It enables us to use existing linear HA algorithm. We introduce extra *delay state variables* in the HA to model the time delayed version of a given state variable. For a state variable with delay T_i, we enforce a discrete transition every T_i time interval and reset the values of the delay state variables to the most recent value of the state variable. Such an approximation leads to an error in state estimation. In this paper, we also provide an upper bound on the error of state estimation for the infusion pump example. We use SpaceEx [10] reachability analysis tool to perform the reachability analysis using zonotopes on the analgesic infusion pump example and derive the safe and unsafe states. Reachability Analysis of pharmacokinetic model has been proposed [25], however it only considers fixed time step and linear dynamics of the system without considering transport delay. Our technique allows usage of more accurate SpaceEX reachability analysis algorithm which uses variable time steps and also enables consideration of transport delay.

The rest of the paper is organized as follows. Section 2 discusses the analgesic infusion pump example in detail. Section 3 discusses the hybrid modeling of the analgesic infusion pump example, the safety properties that can be analyzed and the reachability analysis technique. Section 4 provides the error analysis of the reachability study. Section 5 demonstrates the results of the reachability analysis and errors associated with it. Section 6 gives an overview of the related works in the area of medical device safety analysis. Section 7 discusses the applicability of the hybrid system approach and fundamental challenges faced in applying this technique to other MCPSes and, finally section 8 concludes the paper.

2. ANALGESIC INFUSION PUMP

Consider the case of a wearable infusion pump on an user delivering anesthetic as shown in the Figure 1. The operator issues a constant anesthetic concentration in the blood as an input, which is to be maintained by the infusion pump for a given time. The infusion pump control algorithm takes this as the *reference input* and uses feedback from the pharmacokinetic model of the diffusion process to control the infusion rate so that the desired drug concentration is reached. In this process, the infusion pump first infuses drug with an initial rate. Then based on the estimation of the drug concentration level in the blood for this initial infusion, through the pharmacokinetic model, the pump modifies its future infusions until the drug concentration is stable and the desired drug level is reached.

2.1 Discrete Control Algorithm

The control algorithm [18] of the pump is given in Algorithm 1. The initial infusion rate is x_0. Table 1 gives a list of notations used in this paper. 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

Table 1: Notations for reachable states and over approximations.

Notation	Definition
y_1	state space variable
y_2	state space variable
z_1	drug concentration in blood
z_2	arterial drug concentration
u	infusion rate at time t
A_p	a state matrix
A_s	a state matrix
B_p	a state matrix
$\dot{Q}$	cardiac output
C_s	a state matrix
C_p	a state matrix
T_i	infusion input delay
T_p	cardio-pulmonary transport delay
T_r	arterial, capillary and venous transport delays
Q	discrete states in Hybrid Automata
V	variables in Hybrid Automata
Init	initial values of variables in Hybrid Automata
F	set of differential equations for each state in Hybrid Automata
E	set of discrete transition relations in Hybrid Automata
G	guard conditions to state transitions in Hybrid Automata
y_1^e	error vector of y_1 due to time delay
y_2^e	error vector of y_2 due to time delay
u^e	error vector of u due to time delay
Y_e	error vector for state variables
U_e	error vector for the input drug concentration
J	Jordanian matrix
M	Modal matrix
Y_e^{max}	maximum error vector for state variables due to time delay

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. The magnitude of the bolus is also a variable of the system.

Algorithm 1 Discrete Time Infusion Control(Desired Drug Level Cp_d,Initial Infusion(x_0),Increment Step(δx),Time Step(δt)).

1: NoOfSteps = Total Time / δt // *time discretization*
2: **for** i = 1 to NoOfSteps **do**
3: Infusion Rate x_i = x_{i-1}
4: Predicted Drug Level Cp_p = Pharmacokinetic Simulation with infusion rate x_i
5: Increment Infusion Rate x_{incr} = x_i + δx
6: Predicted Future Drug Level Cp_f = Pharmacokinetic Simulation with infusion rate x_{incr}
7: slope = $\frac{Cp_f - Cp_p}{x_{incr} - x_i}$ // *assume linear variation of drug concentration with infusion rate*
8: Cp_0 = Cp_p - slope $\times$ x_i
9: **if** $Cp_0 \geq Cp_d$ **then**
10: New Infusion Rate x_{i+1} = $\frac{Cp_0 - Cp_d}{slope}$
11: **else**
12: New Infusion Rate x_{i+1} = x_{incr}
13: **end if**
14: **end for**

2.2 Analgesic Interactions

The pharmacokinetic model expresses the drug diffusion process as a set of multi-variable linear differential equations in state space form (Equation 1).

$$\begin{aligned} \dot{y_1} &= A_p y_1 + B_p \dot{Q} z_2 + B_p u(t - T_i), \\ z_1 &= C_p y_1(t - T_p), \\ \dot{y_2} &= A_s y_2 + B_s \dot{Q} z_1, \\ z_2 &= C_s y_2(t - T_r). \end{aligned} \tag{1}$$

Here y_1 and y_2 are the state space variables of the equation. y_1 consists of vectors of left heart, lung blood, lung tissue and right heart compartments through which infused drug passes. Newly infused drug merges with recirculated drug from Vessel Rich Group, Muscle, Fat and Residual drug which is represented by y_2 state space variable vectors. A_p, A_s, B_p, $\dot{Q}$, C_s, and C_p are constants. z_1 is the drug concentration in the blood while z_2 is the arterial drug concentration. 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-delayed. 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-delayed nature of the physical process comes from the consideration of the transport delays.

3. MODELING INFUSION PUMPS USING HYBRID AUTOMATA

In general a hybrid dynamical system is defined as:

DEFINITION 1. *Hybrid Automata (HA): The HA [15] is a tuple* $M = \{Q, V, F, Init, E, G\}$, *where:*
- $\mathbf{Q} = (\{q_1, q_2 .... q_n\}) \cup b$ *is a set of* $n + 1$ *discrete states.*
- $\mathbf{V} = \{v_1, v_2, \ldots, v_m\}$ *is a set of* m *real numbered variables in the real set* $\mathbb{R}$.
- $Init : Q \times V \rightarrow \mathbb{R}$ *is a set of functions, which specifies the initial values of the variables in* V *for each discrete state, which are also the boundary conditions for solving the differential equation.*
- $\mathbf{F} : Q \times V \rightarrow Exp(\Sigma)$ *denotes a mapping that maps each variable for a given discrete state to a function of the form* $F(v, \dot{v}, Init(q, v)) = 0$. F *is typically a set of differential equations for each state in* Q, *which governs the temporal variation of the continuous variables in* V. *The set of all functions* F *is denoted by* $Exp(\Sigma)$.
- $\mathbf{E} \subseteq Q \times Q$ *is a set of discrete transition relations in the model. Presence of the ordered pair* (q_i, q_j) *indicates a transition from state* q_i *to* q_j.
- $\mathbf{G}$: $E \times 2^V \rightarrow \mathbb{R}\ OP\ \mathbb{R}$ *is a set of relations to specify guard conditions to state transitions, where* OP *is the set of relational operations* $\{<, >, \leq, \geq\}$. *A member in* G *can be the equation* $v_1.v_2 > \alpha$ *where* $\alpha \in \mathbb{R}$ *is a constant.*$\diamond$

We apply this definition to the infusion pump example and define a hybrid automata representation. The control algorithm of the infusion pump is represented using finite state automata (FSA) as shown in Figure 2. Each state of the FSA representation is associated with the continuous dynamics of analgesic diffusion. For the infusion, Definition 1 can be instantiated as follows -

- $Q = \{Normal, Off, PCA\}$ is a set of discrete states. In *Normal* state the infusion pump inserts drug with specified infusion rate. When drug concentration at set point goes above threshold value d_{high}, infusion pump stops working until drug concentration comes back to normal. Patient might ask for bolus, in PCA state, the extra drug gets infused depending on the request.
- $V = \{\ x_{LH}, x_{LB}, x_{LT}, x_{RH}, x_{VRG}, x_M, x_F, x_R, u, t\ \}$ is a set of variables, where x represents drug mass at different components of the organ. LH, LB, LT and RH are

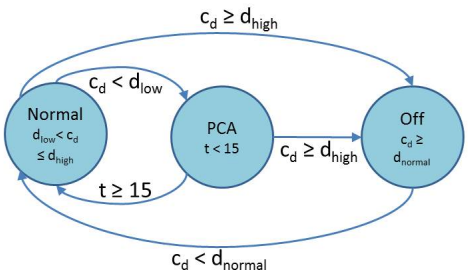

Figure 2: Hybrid Automata model for Infusion Pump.

left heart, lung blood, lung tissue and right heart compartments of the cardiopulmonary subsystem. VRG, M, F and R represents Vessel Rich Group, Muscle, Fat, Residual from systematic subsystem. u represents effective infused drug concentration.

- At each discrete state the $Init$ mapping enables us to specify the control decisions. The control decision of the infusion pump is the future infusion rate u. Thus, in the normal state the value of u is computed according to Algorithm 1. In the PCA state, the value of u is incremented by a constant bolus value. In the off state the value of u is set to zero. In addition in each set the value of variable t is set to zero in order to keep track of the time spent in the state.

- The differential equations used in each state are similar to the Equation 1. In addition to keep track of time the equation $\dot{t} = 1$ is also included.

- E represents the transitions in the FSA. In infusion pump FSA, total five transitions are possible, from *Normal* to *Off* and reverse, *Normal* to *PCA* and reverse, and *PCA* to *Off*. These transitions are governed by guard conditions defined in G.

- G are the guard conditions to decide transitions between the states of FSA. The OP defines the relational condition of guard in terms of less than, greater than or equal to. When drug concentration goes beyond threshold d_{high}, infusion pump hybrid automata model goes in *Off* state. It stays in *off* state until drug concentration becomes normal. If the user or patient asks for bolus and if drug concentration goes below d_{low}, *PCA* state comes into picture to infuse extra drug.

3.1 Safety properties in MCPS

Safety is a property of an MCPS by virtue of which it can be guaranteed that there will be no harm to the infrastructure and to the human body during normal or faulty operation of the system. The most generic definition of safety for a MCPS can be found in the ISO 60601 standard for safety of medical electrical equipment. ISO 60601 defines safety as the avoidance of hazards due to the operation of a medical device under normal or single fault condition [1]. One of the unique features of an MCPS is the interaction of the computing unit with the physical environment. Hence, for MCPSes the safety concerns are related to the interaction between the computing device and the physical environment, *interaction safety.*

Interaction safety: Traditionally, researchers have focused on bypassing this interaction characterization and transforming the safety assurance problem into a well understood problem in computer science such as formal model reachability analysis. In this regard, several static assumptions on the physical environment has been considered, which abstract out the dynamic nature of the physical environment. For example, in works such as [4, 19], infusion pump software has been modeled using a timed automata. The diffusion process is simplified so that the drug concentration in the blood is incremented by the infusion rate instantaneously. The problem of safety assurance is consequently reduced to developing bug free software or a control system analysis problem. Such simplified notion of safety, however, may not entirely capture the hazards resulting from the dynamic cyber-physical interactions. In essence more focus is needed on the interaction safety.

Addressing interactions safety is a challenging task. Principally, it requires exact understanding of the physical processes of the environment and the properties of the computing unit that affect the physical processes. In this paper, special focus is given on interaction safety of MCPSes. For the specific case of analgesic infusion pump, the drug overdose is a result of faulty interaction of the discrete infusion control algorithm with the physical dynamics of drug diffusion.

3.2 Reachability analysis of HA

The reachability analysis of HA considers the computation of the states that the continuous variables can reach at any point in time. The reachability analysis depends on infinite precision computation of the solution of the state space equations. For a given linear differential equation: $\dot{x} = ax(t) + bu(t)$, where a and b are constants and $u(t)$ is the input, the reachability analysis assumes that we can compute $x(t)$ for a time t using the Equation 2.

$$x(t) = e^{at}x(t_0) + \int_{t_0}^{t} e^{a(t-\tau)}bu(\tau)\mathrm{d}\tau. \tag{2}$$

Such a computation of $x(t)$ from $x(0)$ is referred to as the image computation from an initial state $x(0)$ for a time t. Given a set of continuous state V, as an initial state, the reachability analysis starts by computing the image of the initial state V and appending it to the reach set. An initial state is generally specified using some form of initial conditions, which may be satisfied by infinite number of states. It is an intractable proposition to go through each state and compute its image. Instead suitable approximation of the states using convex polygonal cells are used. Such approximations over estimate the reach set but enable tractable execution of the reachability analysis algorithm. Thus, at any point in its execution the reachability analysis algorithm keeps track of a *symbolic state* S, which comprises of: a) a reach set R, a collection of convex polygons and b) a discrete state of the controller q_i. The reachability analysis algorithm uses the following steps: a) from this symbolic state S the images of the vertices of the convex hull of R, $Imag(R_v)$, is computed using Equation 2, b) if no member of $Imag(R_v)$ satisfies a guard condition then $S = \{q_i, Imag(R_v)\}$ is the new symbolic state and $R \bigcup Imag(R_v)$ is the new reach set, c) otherwise if a guard condition is satisfied then the corresponding transition is processed and the new discrete state q_j is obtained, and d) for this state q_j the $Init$ function

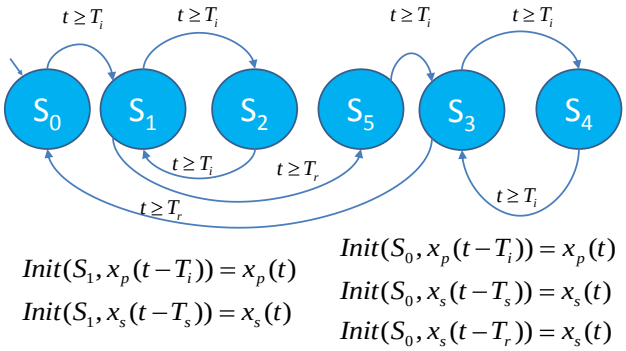

Figure 3: Each state of Figure 2 is further represented using six states and transitions between them in order to represent transport delays.

is applied on the $Imag(R_v)$ to obtain the new symbolic state $S = \{q_j, Init(q_j, Imag(R_v))\}$ and the new reach set is $R \bigcup Init(q_j, Imag(R_v))$.

3.3 Implementation in SpaceEx

Proving safety properties of hybrid systems is a well established research topic. Reachability analysis of hybrid automata is an undecidable problem for any general linear dynamics. Nevertheless several tools are being proposed for reachability analysis of hybrid automata with a restricted set of continuous dynamics. Tools such as HyTech [16] focused on relatively simple continuous dynamics in each discrete state, where the derivative of the continuous variables does not depend on any external inputs. For such hybrid automata, the computation of reachability to different states can be realized using linear algebra. However, in such methods one can easily find cases which were undecidable and could not be solved.

Hybrid verification tools proposed in [2,5,8] focus on computing approximations of the reachable states for systems having linear continuous dynamics with external inputs. However, these tools cannot handle systems with more than tens of continuous variables. Only recently there has been efforts to over-approximate reach sets using Zonotopes [12], which has led to a dramatical increase in the number of variables that can be handled. Tools that support such analysis are SpaceEx [10], HybridSAL [27], and PHaver [9]. Among the three, SpaceEx uses variable time steps in simulating time and provides reach sets within a lower error bound. It uses an improved approximation model by combining polyhedra and support function representations of the continuous state space to provide better accuracy. Thus, in this paper we use SpaceEx to specify and analyze our hybrid automata model.

SpaceEx however does not support time delays in the differential equations. To model time delays in SpaceEx we consider an approximation strategy. We see that there are two sets of time delays: a) $T_i = T_p = T_s = 5$s and b) $T_r = 30$s. We represent each state of the infusion pump hybrid automata as a collection of six states as shown in Figure 3. We consider two types of transitions one of which occurs every T_i seconds and the other occurs every T_r seconds. Now in each state we keep separate variables $y_1'(t)$ and $y_2'(t)$ for storing the value of $y_1(t - T_i)$ and $y_2(t - T_r)$, respectively. During each transition the parameters y_1' and y_2' are reset to the current values of y_1 and y_2 using the $Init$ functions of the HA definition as shown in Figure 3. This means that whenever the HA is in state S_1 it resets the value of y_1' to the current value of $y_1(t)$ and does not change it for T_i seconds after which it transits to S_2. On transition to S_2 the value of y_1' is again set to the current value of $y_1(t)$. Thus, every T_i seconds the parameter y_1' gets the correct value of $y_1(t - T_i)$. Same thing happens with y_2' every T_r seconds. Such an approximation allows us to use SpaceEx for time delayed systems albeit with a cost of inaccuracy. In Section 4, we analyze the error of such an approach.

4. ERROR ANALYSIS OF THE TIME DELAY APPROXIMATION

In our hybrid system specification we assume a zero order hold approximation for the delayed state variables i.e., $y_1(t - T_i) = y_1(t)$, $y_2(t - T_r) = y_2(t)$, and $u(t - T_i) = u(t)$. This will introduce error in valuation of the state variables. Let us consider that the error in y_1, y_2, and u is denoted by y_1^e, y_2^e, and u^e, respectively. Due to these errors there will be errors in the differentials as well and the errors will be governed by the state space equations as shown in Equation 3.

$$\dot{y_1^e} = A_p y_1^e + B_p \dot{Q} C_s y_2^e + B_p u^e, and \qquad \dot{y_2^e} = A_s y_2^e + B_s \dot{Q} C_p y_1^e. \tag{3}$$

Note that the errors in y_1 and y_2 themselves form a linear time invariant system. The equation set 3 can be converted to a single equation of the form: $\dot{Y_e} = AY_e + BU_e$, where A $= \begin{pmatrix} A_p & B_p\dot{Q}C_s \\ B_s\dot{Q}C_p & A_s \end{pmatrix}$ and B $= \begin{pmatrix} B_p \\ 0 \end{pmatrix}$, Y_e is the error vector for state variables, and U_e is the error vector for the input. Such a state space equation can be solved using the modal decomposition method [14]. The solution is of the form given in Equation 4.

$$Y_e = e^{AT_i} Y_e(0) + \int_0^{T_i} e^{A(T_i - \tau)} BU_e \mathrm{d}\tau. \tag{4}$$

We find the Jordanian J of the matrix AT_i and the modal matrix M and compute e^{AT_i} as $e^{AT_i} = MJM^{-1}$. The maximum value of the error in state estimation Y_e is then given by Equation 5.

$$Y_e^{max} = max(MJM^{-1})Y_e(0) + max(MJM^{-1})T_i BU_e. \tag{5}$$

Since the error is an over approximation of state variables' rate of change systems deemed unsafe are truly unsafe. However, some safe configurations can also be considered unsafe. The values of the error for the analgesic infusion pump is given in Section 5.3.

5. REACHABILITY ANALYSIS RESULTS

Reachability analysis of hybrid automata is to calculate the reach set of variable values staring from initial set over the time. We have applied this method to calculate the reach set of drug concentration at different points on the human body using hybrid automata of analgesic infusion pump described in above section. If the drug concentration goes above threshold, we have considered that as unsafe state.

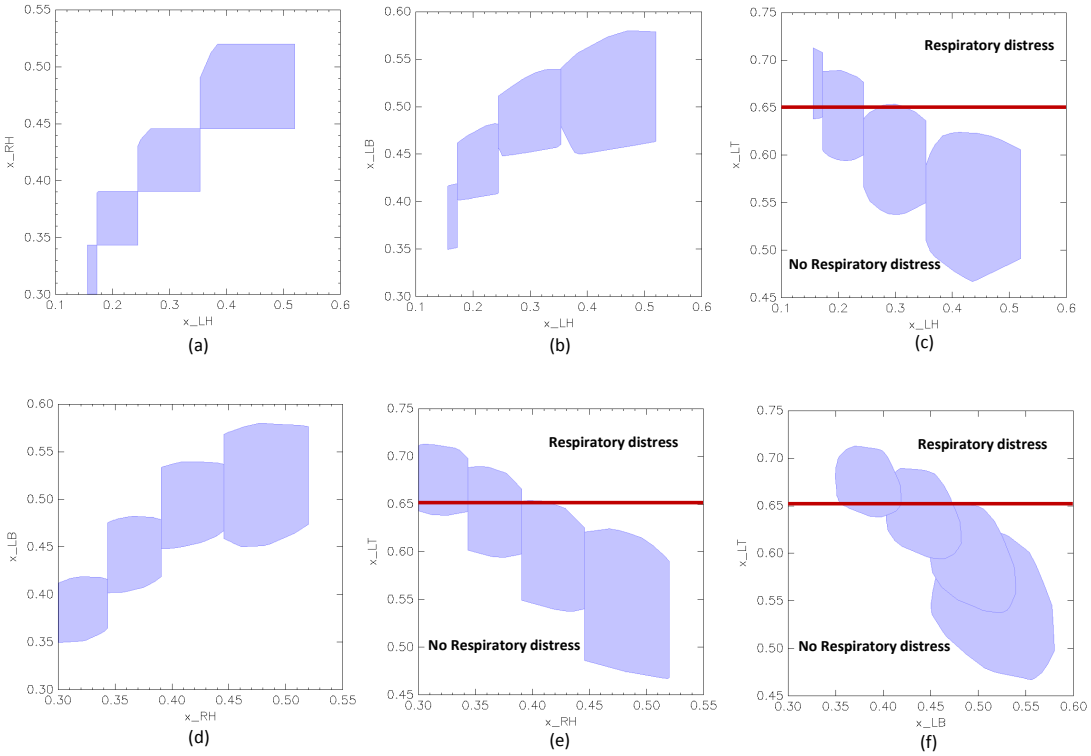

Figure 4: Reachable states for analgesic infusion pump configuration discussed in Section 5.2

5.1 SpaceEx setup

SpaceEx [10] tool is used to do the reachability analysis of analgesic infusion pump hybrid automata. The parameter values for the system equations are obtained from the case study on analgesic infusion conducted by [30]. Initial values of variables are set to 0.52 *liters* and the initial drug concentration is considered as 0.3 *liters*. We are monitoring the drug concentration at left heart(x_{LH}), lung blood(x_{LB}), lung tissue(x_{LT}) and right heart(x_{RH}) parts of the human body. For the reachability analysis, we have assumed that drug concentration at monitored points should not go above $d_{high} = 2.5$ *liters* and below $d_{low} = 0.6$ *liters*. If it goes beyond these threshold values, we say that the system is in unsafe state.

5.2 Reachability results

Figure 4 illustrates the reachable set for monitored points. The shaded region indicates the possible values that state variable of monitored points can take. The state variable values are obtained by using system Equation 1. It can be expressed as, $x_{LH} = c_d \times V_{LH}$, where x_{LH} is the state variable for left heart, c_d is the infused drug concentration and V_{LH} is the volume of the blood into which drug is infused. From this formula, the highest drug concentration becomes 0.65 *liters*, using $c_d = d_{high} = 2.5$ *liters* and $V_{LH} = 0.26$ *liters*. The bold line is marked at threshold value, 0.65 *liters* which is the maximum allowable drug concentration. These reach sets show state variable of lung tissue reaches above this threshold and system reaches in unsafe state for lung tissue point over the time. Left heart, lung blood and right heart monitored points always remain in safe state. We also logged the runtime of the reachability analysis using SpaceEx. The time taken to perform the reachability analysis with 13 state variables was 4.9 seconds on a regular Intel dual core desktop.

5.3 Error analysis results

Equation 5 is used for computing the maximum error of the zero order hold approximation for time delayed systems. We consider a transition every T_i seconds as discussed in Section 3.3 and an initial maximum error of 0.26 * 2.5 per min for the state variable y_1 and y_2, which corresponds to a maximum error of 2.5 *liters* in the actual drug concentration. Further, we assume a maximum error of 0.6 *liters* in U which corresponds to a bolus input. Then from Equation 5 we obtain the maximum error to be 0.034 per min in the state vector. From the reachability analysis results in Figure 4, we see that the maximum value of the state variables is around 0.75. It gives a maximum percentage error of 4.5% which is acceptable based on value of discretization step. To get better accuracy, discretization step size should be decreased which will increase computations required for reachability analysis.

6. RELATED WORK

The past research in MCPS safety analysis can be classified according to the type of safety addressed.

Scenario safety: It considers the safety of the MCPS and its environment from a the high level decision making perspective. It considers how the MCPS handles random hazardous events occurring in the environment potentially causing harm to life and infrastructure if not mitigated called criticalities. Example research in this regard includes the criticality response planning, evaluation, and actuation [23,

29] framework developed at the IMPACT Lab, Arizona State University.

Network safety: MCPS can involve a network of computing units communicating with each other through wireless or wired channels to achieve mission critical and smart operations. The wireless channel is prone to errors such as bit errors, burst errors, and multi-path fading errors. Under this circumstances the network safety ensures that information transferred from one device to the other is not corrupted, reaches within a given amount of time, and is not lost due to errors in the channel. Evaluation of network safety in medical device networks has been performed by Gehlot et al. in Villanova University [11].

Software safety: This is a broad area of research and is related to the operation of the software of the MCPS computing devices. It includes:

1. *Code safety*, which considers safety from coding errors such as infinite while loops, unreachable conditions etc. as performed in the project at the University of Pennsylvania [24].
2. *Control system safety*, which considers safety from undershoot, overshoot, instability and long settling times (investigated as part of the design of infusion pumps [7]).

Interaction safety: Interaction safety considers the cooperation of the software of a MCPS with the dynamic physical environment. Formal methods has been used extensively to verify interaction safety of MCPSes. However, most of these works try to characterize the computational aspects of the medical device or make simplifications to the physiology of the human body. Formal models have been used [4] to analyze infusion pump software hazards. The use of formal models in medical device regulation has been proposed [19]. However, none of these works consider formal representation of the interactions of the medical device with the human body. The physiology of heart has been modeled using a timed automata [20]. The timed automata model of the heart is a statistical model based on previously collected heart rate data and can only prove properties related to timing of events. Timed automata is also used to verify the control actions in a closed loop infusion system with pulse oximeter signals as feedback [3]. However, according to the authors the safety evaluation is often not accurate for more complex interactions of the device with the human body. Further, the authors have not considered the dynamics of the drug diffusion process in the human body. In this paper, we model the interaction of medical device with the drug diffusion dynamics and perform a comprehensive reachability analysis on it. The result of the reachability analysis is an assessment on the safety of the patient. Further, we also provide an error bound on the reachability study.

7. DISCUSSION

MCPSes are inherently complex systems and application of existing hybrid automata theory necessitates a number of simplifying assumptions. In the example, shown in this paper we had to employ a zero order hold assumption on the time delayed system in order to apply HA theory to the analgesic infusion pump. The model that we used was simple and not very accurate and still we had to make simplifying assumptions to fit the existing theory of reachability analysis. Several accurate models of drug diffusion are proposed in recent literature which impose many more hurdles on using the existing HA theory. A few of the challenges are listed below:

Non-linear interaction: Non-linearities are inherent in the human. The infused drug concentration variation over time inside human body follows an error function curve. Hence, MCPS models can be non-linear in nature. Non-linearities can arise in many different forms in a model including delays and multiplicative terms. Theoretical analysis techniques for linear hybrid systems thus may not apply to a large class of MCPSes.

Dynamic context changes: The human body in an MCPS is constantly undergoing change in context, e.g. weather changes, change in temperature, change in physiological condition. Mobility, for example, is an primary cause of such context changes in the system. These changes affect the operation of the computing units in the MCPS, for example, movement from indoor to an outdoor environment changes the packet delivery rate (PDR) of the wireless medium, which can lead to loss of control information between the controller and the actuation device in an MCPS. Safety analysis of MCPSes should consider such dynamic changes in the environment as a part of the model. Reachability analysis of hybrid automata does not consider random events.

Spatio-temporal interactions: The interactions in an MCPS, can often be spatio-temporal in nature. For example, more accurate models of drug diffusion express the effects of drug over space and time. Capturing spatio-temporal effects requires multiple independent variables that determine the evolution of the model. Researchers have extensively studied models that evolve over a single independent variable, most commonly time. However, there exist limited efforts to capture model evolution over both space and time.

Aggregate effects: Many MCPSes, such as network of sensors on body, comprise of more than one computing entities distributed across the environment. They often perform concurrent operations; thus causing aggregation of the detrimental impact on the environment from multiple nodes. For example, in chemotherapy multiple drugs are infused simultaneously to achieve a desired cancer cell death ratio which cannot be achieved if the same dosage of any one drug is administered. Such effects have been very recently considered in the hybrid system analysis domain [6]. It requires development of new theories far beyond the realms of linear time dependent hybrid system analysis.

8. CONCLUSIONS

In this paper, we have used hybrid automata for formal patient safety verification of analgesic infusion pumps. In doing so we have outlined several challenges of hybrid modeling and analysis of MCPSes in general. MCPSes often have time delays which makes the reachability analysis of hybrid automata intractable. We have proposed a zero order hold approximation technique that allows reachability analysis of time delayed hybrid models. The methodology is prone to errors and we have also provided a bound on the error of the reachability analysis. We have applied the technique to analgesic infusion pumps and have obtained the unsafe configurations that can cause respiratory distress.

9. REFERENCES

[1] ISO 60601 safety standard. `http://www.iso.org/iso/iso\_catalogue/catalogue\_tc/catalogue\_detail.htm\\?csnumber=45605`.

[2] M. Althoff, O. Stursberg, and M. Buss. Reachability analysis of linear systems with uncertain parameters and inputs. In *Decision and Control, 2007 46th IEEE Conference on*, pages 726 –732, dec. 2007.

[3] D. Arney, M. Pajic, J. M. Goldman, I. Lee, R. Mangharam, and O. Sokolsky. Toward patient safety in closed-loop medical device systems. In *ICCPS '10: Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, pages 139–148, New York, NY, USA, 2010. ACM.

[4] D. E. Arney, R. Jetley, P. Jones, I. Lee, A. Ray, O. Sokolsky, and Y. Zhang. Generic infusion pump hazard analysis and safety requirements version 1.0. 2009.

[5] E. Asarin, O. Bournez, T. Dang, and O. Maler. Approximate reachability analysis of piecewise-linear dynamical systems. pages 21–31. Springer, 2000.

[6] A. Banerjee and S. K. S. Gupta. Spatio-temporal hybrid automata for safe cyber-physical systems: A medical case study. *Intl' Conf' on Cyber-Physical Systems (To Appear)*, 2013.

[7] L. G. Bleris and M. V. Kothare. Implementation of model predictive control for glucose regulation on a general purpose microprocessor. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pages 5162 – 5167, dec. 2005.

[8] A. Chutinan and B. H. Krogh. Verification of polyhedral-invariant hybrid automata using polygonal flow pipe approximations. pages 76–90. Springer, 1999.

[9] G. Frehse. Phaver: Algorithmic verification of hybrid systems past hytech. In *HSCC*, pages 258–273, 2005.

[10] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler. Spaceex: Scalable verification of hybrid systems. In *Proc. 23rd International Conference on Computer Aided Verification (CAV)*, LNCS. Springer, 2011.

[11] V. Gehlot and E. B. Sloane. Ensuring patient safety in wireless medical device networks. *Computer*, 39:54–60, April 2006.

[12] A. Girard and C. Guernic. Zonotope/hyperplane intersection for hybrid systems reachability analysis. In *Proceedings of the 11th international workshop on Hybrid Systems: Computation and Control*, HSCC '08, pages 215–228, Berlin, Heidelberg, 2008. Springer-Verlag.

[13] D. G. M. Greenhalgh et al. Temperature threshold for burn injury: An oximeter safety study. *Journal of Burn Care and Rehabilitation*, 25(5):411–415, 2004.

[14] E. Hendricks, O. Jannerup, and P. Sørenson. *Linear Systems Control: Deterministic and Stochastic Methods*. Springer-Verlag, 2008. The book is used in the Elektro-Automation Course Linear Control Design 2 (Reguleringsteknik 2).

[15] T. A. Henzinger. The theory of hybrid automata. *Logic in Computer Science, Symposium on*, 0:278, 1996.

[16] T. A. Henzinger, P.-H. Ho, and H. Wong-Toi. HYTECH: A model checker for hybrid systems. In *Proceedings of the 9th International Conference on Computer Aided Verification*, volume 1254 of *LNCS*, pages 460–463. Springer, 1997.

[17] T. L. Jackson and H. M. Byrne. A mathematical model to study the effects of drug resistance and vasculature on the response of solid tumors to chemotherapy. *Mathematical Biosciences*, 164(1):17 – 38, 2000.

[18] J. R. Jacobs. Algorithm for optimal linear model-based control with application to pharmacokinetic model-driven drug delivery. *Biomedical Engineering, IEEE Transactions on*, 37(1):107 –109, 1990.

[19] R. Jetley, S. P. Iyer, and P. L. Jones. A formal methods approach to medical device review. *Computer*, 39(4):61–67, 2006.

[20] Z. Jiang, M. Pajic, A. Connolly, S. Dixit, and R. Mangharam. Real-time heart model for implantable cardiac device validation and verification. *Real-Time Systems, Euromicro Conference on*, 0:239–248, 2010.

[21] X. Liu and J. Shen. Stability theory of hybrid dynamical systems with time delay. *Automatic Control, IEEE Transactions on*, 51(4):620 – 625, april 2006.

[22] L. E. Moser et al. Formal verification of safety-critical systems. *Softw. Pract. Exper.*, 20(9):799–811, 1990.

[23] T. Mukherjee, K. Venkatasubramanian, and S. K. S. Gupta. Performance modeling of critical event management for ubiquitous computing applications. In *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, pages 12–19, New York, NY, USA, 2006. ACM.

[24] U. of Pensylvannia. Generic infusion pump project. `http://rtg.cis.upenn.edu/gip.php3`.

[25] S. Sankaranarayanan, H. Homaei, and C. Lewis. Model-based dependability analysis of programmable drug infusion pumps. *Formal Modeling and Analysis of Timed Systems*, pages 317–334, 2011.

[26] The Networking Information Research and Development. `http://www.nitrd.gov/about/blog/white_papers/16-Importance_of_Cyber-Physical_Systems.pdf`.

[27] A. Tiwari. Hybridsal relational abstracter. In P. Madhusudan and S. Seshia, editors, *Computer Aided Verification*, volume 7358 of *Lecture Notes in Computer Science*, pages 725–731. Springer Berlin Heidelberg, 2012.

[28] Q.-Y. Tong, G.-f. Yan, and G.-Z. Zhao. Stability analysis of hybrid systems with time-varying delayed perturbations via single lyapunov function. In *Machine Learning and Cybernetics, 2003 International Conference on*, volume 2, pages 919 – 922 Vol.2, nov. 2003.

[29] K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta. CAAC - an adaptive and proactive access control approach for emergencies for smart infrastructures. *ACM Transactions on Autonomous and Adaptive Systems Special Issue on Adaptive Security*, (To Appear).

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