

# Interference Free Integration of Pervasive Applications

Christophe Soares<sup>\*†</sup>, Rui S. Moreira<sup>\*†</sup>, Ricardo Morla<sup>\*</sup>, José Torres<sup>†</sup> and Pedro Sobral<sup>†</sup>

<sup>\*</sup>INESC TEC, FEUP. University of Porto, Porto, Portugal.

Email: {cpsoares, rjm, rmorla}@inescporto.pt

<sup>†</sup>ISUS Unit, FCT. University Fernando Pessoa, Porto, Portugal.

Email: {csoares, rmoreira, jtorres, pmsobral}@ufp.edu.pt

**Abstract**—Off-the-shelf smart devices and applications are expected to be pivotal in the coming need for massive home care. Deployment and integration of these systems in the same household may result in unplanned interactions involving users and entertainment, communication, and health-related devices. These unplanned interactions are a major concern when, for example, communication or entertainment applications interfere with the behavior of health-related devices. This paper presents a novel graph-based approach for representing the expected behavior of off-the-shelf smart devices and applications, their interactions, and for detecting interference in home care settings. A set of home care scenarios is used to assess the applicability of our approach. Our graph-based interference detection approach is integrated in the Safe Home Care reflective platform, which allows reifying the state of off-the-shelf systems and simulating home care scenarios.

**Keywords**-Feature Interaction, Interference-free, Graph-based Interference Pruning, Safe Home Care, Reflective Middleware, Ubiquitous Computing.

## I. INTRODUCTION

The deployment of pervasive computing applications into a large number of households raises challenges such as security, privacy, economic viability, and interferences or feature interactions (FI) between the pervasive computing applications of the household. In this paper, we present a view on FI in pervasive computing applications and a set of approaches to address this issue. In this paper, we complement our previous FI detection approach with mechanisms for identifying and solving the causes of FI (cf. inter-system filter). More precisely, we extend our former graph-based representation to explicitly integrate information about intra-systems interactions, i.e., we represent the set of media that each OTS-system state might affect or be affected by other OTS-system states (e.g., temperature, user awareness or interaction, etc.).

This paper is organized as follows. The rest of the introduction presents a view on FI in pervasive computing and the specific problems that we address. Section II explains our graph-based approach for tackling FI. Section III presents increasingly complex use cases applied to evaluate and compare FI detection with different approaches. Section IV reviews related work in FI. Finally, in Section V, we present final remarks and discuss future work.

### A. A View on the Feature Interaction Problem in Ubicomp

In ubiquitous computing environments [1] computing becomes intrinsic in the daily lives of users, with OTS systems like smart televisions, drug dispensers, home automation systems, and entertainment devices supporting users at home. We expect that off-the-shelf (OTS) systems will enable ubiquitous computing to be deployed into a large number of households: these systems will be purchased spontaneously by users in shops and will not be designed for a specific household. Also they are not prepared to interwork in the same-space without previous managed integration.

### B. State-based Model

We take the OTS vision of ubiquitous computing further by assuming each system to contain states and behaviors. A State corresponds to a particular condition an OTS system is in at a specific time, e.g., a phone is ringing or a drug dispenser notifies a missed medicine intake. States provide top-level information of what is happening/occurring on the system. Behavior corresponds to a time sequence of States of a running OTS system. For example, when the phone receives a call ( $state_1$ ) it starts ringing ( $state_2$ ), when the user decides to take the call ( $state_3$ ), the phone stops ringing ( $state_4$ ). When the user hangs-off the phone the call ends ( $state_5$ ).  $\langle state_1, state_2, state_3, state_4, state_5 \rangle$  is a sequence of states that depicts an expected behavior of the OTS system *Phone*.

### C. Interaction Media

When an OTS system is in use, it may interact with other systems through several media (e.g., concrete environmental variables such as brightness, temperature and sound, or more abstract variables such as the user perception). Each interaction has two effects: i) some OTS system states will affect specific media and ii) other OTS system states will be affected by these media. For example, when the phone starts ringing it will affect the *sound* medium, but the phone call will only occur if the user is aware he has to take the call. We consider that the natural or expected behavior of an OTS system is the behavior the system has when it is deployed in a household without any other OTS system. Thus when combining several OTS systems in

the same place, the expected behavior of an OTS system may interact with the natural behavior of others systems through a shared media, resulting in possible unexpected behavior. There are two types of interaction: i) some could be safe, when two OTS systems affects the same media and it results in desirable behavior (e.g., two OTS systems are attempting to increase the living room’s temperature, useful interactions) or ii) others are considered unsafe when they produce an undesirable behavior (e.g., two OTS systems interact constantly with the living room’s lights, one switch it on and the other turned it off). This problem is known as FI [2], it was firstly introduced in the telecommunications field and afterwards extended to pervasive computing [3]. Calder and Magill considered FI as interference between services or features (e.g., call forwarding in telecommunications) [4]. Morla and Davies proposed a framework for describing and reason about the problem of interference in ubiquitous places [5]. Here, we also consider that each OTS system provides a service for the smart-space which might interfere with each other.

#### D. Problems and approaches

In this paper, we present an approach for: 1) detecting misbehavior of individual OTS systems in the household, 2) detecting a priori potential interactions between several OTS systems, occurring through common shared media (by exploring the facilities of a 3D simulation framework [6]), and 3) detecting the causes of misbehavior actually occurring in the household (identified in 1) and also online potential FI (subset from 2). This is based on a graph-traversal algorithm that uses observed and expected OTS system states. The observed OTS system states are introspected through physical sensors (e.g., sound, presence, temperature and brightness sensors) and concrete SNMP-based agents (e.g., ongoing VoIP call, missed medicine intake, etc.). We assume that: we previously know the expected behavior of all the present OTS systems; the introspection is not subject to noise, hence we are able to determine all state occurrences of each OTS system; and there is only a unique user in the household. Later, Section III provides an evaluation of our interference handling algorithms in different home smart space scenarios where several OTS systems interact together. We describe our approach in Safe Home Care (SHC) reflective framework, which is being prototyped to simulate, analyze, manage and deploy interference-free home settings.

## II. FEATURE INTERACTION DETECTION

Graph representations are well understood and provide a flexible representation for state sequence transitions. We use a graph representation to model both the behavior and the interactions of the OTS systems in the smart space. We propose three approaches based on graphs to detect and identify FI: i) intra-system approach: identify per-system correct state sequences behavior; ii) inter-systems approach:

identify possible interactions between OTS systems through shared media (e.g., same window controlled by two OTS systems, etc.) and iii) combined approach: joints previous approaches both to detect and identify FI causes; and also to detect online possible interactions between OTS systems through a shared media.

This paper focuses on FI detection, evaluated throughout several scenarios. The inter-system FI approach will allow us to identify interactions; by explicitly modeling shared medium links between state transitions of different OTS systems. With combined approach we expect to be able to identify the cause of FI and detect possible unwanted interactions.

#### A. Behavior and Interaction Model for Pervasive Applications using Graphs

Directed graphs can be used to capture both the expected behavior of isolated systems and the observed behavior of combined systems. Figure 1a shows the expected state sequences and the affected/dependent media for a toy example; Figure 1b depicts the state transitions perceived/observed through sensors and available system APIs.

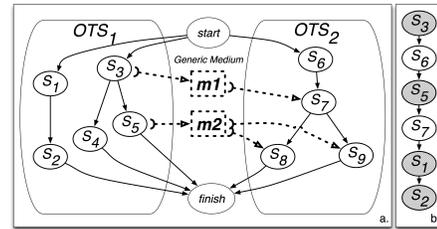


Figure 1. Generic example of FI detection approach: a) Expected States with media dependencies and b) Observed States - with pruned marks

Assuming that each system has a set of well-known state sequences, it would be possible to represent this sequence through a directed graph. The state of an element is characterized by its feature values. Each graph node represents a unique element state (cf. system state), i.e., a value change on any feature. The expected behavior of each deployed system can be captured into a state transition graph, that we name Graph of Expected States (GoES) (see Figure 1a). This approach is extensible to several OTS systems and facilitates the addition of new element graphs or state sequences, e.g., merging different already existent graph model representations. We then, for manipulation purposes only, assemble a single graph with common start and finish nodes from all given OTS system graphs. Hence, the GoES represents all existing OTS systems on the household. During simulation, we capture the states of all OTS systems, i.e., their Graphs of Observed States (GoOS) (see Figure 1b).

The GoES represents how systems should behave without interferences and the GoOS represents the current/observed system behavior. In Figure 1a, all possible state path sequences are:  $\langle S_1, S_2 \rangle$ ,  $\langle S_3, S_4 \rangle$ ,  $\langle S_3, S_5 \rangle$ ,  $\langle S_6, S_7, S_8 \rangle$  and

$\langle S_6, S_7, S_9 \rangle$ . We consider also that states  $\langle S_3 \rangle$  and  $\langle S_5 \rangle$  will affect, respectively, medium  $m_1$  and  $m_2$ . States  $\langle S_7 \rangle$  and  $\langle S_8 \rangle / \langle S_9 \rangle$  will be affected, respectively, by medium  $m_1$  and  $m_2$ . GoES could be provided by the OTS system manufacturer or installers, or even built from statistical in-site observations using Machine Learning techniques. The next three subsections provide a description of our three approaches to detect and tackle FI (see Figure 2) applied to the same toy example described in Figure 1.

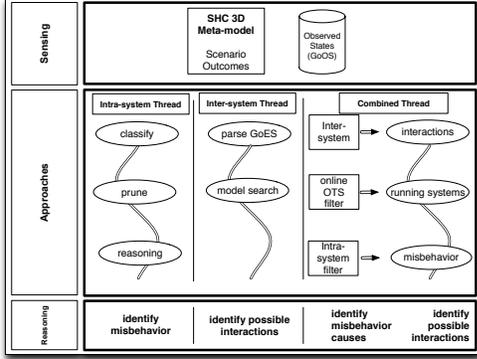


Figure 2. Workflow of the SHC Feature Interaction Engine

### B. Misbehavior Detection: Intra-system Approach

This approach tries to identify/discover systems that their behavior has been changed (cf. FI Problem). The intra-system approach has three steps (see Figure 2) [7]: i) classify – creates a graph with observed OTS system states; ii) pruning – based on the knowledge of expected state, identifies unexpected states; and iii) reasoning – identifies unexpected behavior or malfunctions using the pruned results. For example, assume that the current GoOS sequence is  $\langle S_3, S_6, S_5, S_7, S_1, S_2 \rangle$  (see Figure 1b). Based on these two graphs (Figure 1a and 1b) the pruning algorithm removes the complete GoES path sequences from the current GoOS:  $\langle S_3, S_5 \rangle$  and  $\langle S_1, S_2 \rangle$ . The result set is:  $\langle S_6, S_7 \rangle$  because as expected the  $\langle S_8 \rangle$  or  $\langle S_9 \rangle$  were not observed. The State Pruning Algorithm (SPA) extracts all expected state sequences from the observed behavior; when the result of the SPA algorithm is not an empty sequence of states, then it assumes: i) FI occurs and later would be solved; ii) state sequences or malfunctions were not captured in the existing GoES, which should be re-drawn. The main goal of the SPA algorithm consists of identify and eliminate sequences of GoES sub-paths from the GoOS, until there are no more possible sub-paths to prune. Hence, the algorithm filters all expected actions/states that have been properly executed by deployed OTS systems. The SPA acts as a pre-processing tool and allows to detect the existence of per system FI (cf. knowledge discovery process phases) [8]. We have currently extend this approach to support media-based inter-system FI,

because this first approach identifies an unexpected behavior but the cause remains unknown.

### C. Interactions Detection: Inter-system Approach

To identify the cause of the FI, we use the inter-system analysis; it consists of identifying all possible/offline interactions through a shared medium between OTS systems (see Figure 2). In this particular case (Figure 1a), we identify a priori two possible interactions. First, when the state  $\langle S_3 \rangle$  on OTS<sub>1</sub> occurs, it will affect the medium  $m_1$ ; and respectively could affect/interact with the state  $\langle S_3 \rangle$  on OTS<sub>2</sub>. The second interaction happens when the state  $\langle S_5 \rangle$  on OTS<sub>1</sub> occurs, it will affect the medium  $m_2$ ; and respectively could affect/interact with the state  $\langle S_8 \rangle$  or  $\langle S_9 \rangle$  on OTS<sub>2</sub>. Through the graph model previously presented (see Section II-A), we have two steps: i) model parsing - identify the entire set of media affected by all OTS system states; ii) model search - isolate all OTS system states that are affected by the previously identified media; thus, transitively identifying potential OTS system state FI (i.e., which states might be affected other states).

Media are a set of specific variables, defined a priori, constituted by i) environmental-related data shared by the OTS systems deployed on the smart-space (SS), i.e., physical ubicomp's context, and ii) user-related interaction data, i.e., user ubicomp's context. For example, our specific scenarios provided in Section III-B use the media: i) temperature, sound, brightness, window, physical presence (environmental-related media) and ii) user awareness, user interaction (user-related media).

### D. Misbehavior and Interaction Identification: Combined Approach

The combined approach results of the union of previous presented approaches, i.e., intra-system and inter-system. Hence, we will identify misbehavior, interactions and these dependencies. Best parts of both approaches will allow us to provide better accuracy on our FI detection mechanism. We use as input all the possible interactions that could occur between the OTS systems. We set some filters: online interactions and misbehavior identification. Other filters could also be applied later to enhance our approach as: spatial and temporal. The first filter will remove all the systems that are not running during the observation period. Since they were not executed, we consider they cannot interact with other OTS system. The second filter will identify all the possible causes of an FI using the intra-system output. The intra-system will provide us which systems have not been executed as expected. Based on this, we could know, which medium could have been affected and identify, which system has affected it. Hence, we identify in real-time the possible OTS system sources of FI and we will try to prevent them to interact with other systems. This information would be useful to identify possible causes of intra-system unexpected

behavior. Returning to the example previously described, since  $\langle S_8 \rangle$  or  $\langle S_9 \rangle$  did not occur, we may follow their common dependencies on  $m_2$ . A reverse-path analysis, based on GoES, allow us to identify that both  $\langle S_8 \rangle$  or  $\langle S_9 \rangle$  are affected by  $\langle S_5 \rangle$ . Hence, we are able to identify that  $OTS_1$  was affecting  $OTS_2$  through the medium  $m_2$ .

The other subset from this second filter returns all the possible interactions that occur through a shared media. They do not affect the expected behavior of an OTS system. Since  $\langle S_3 \rangle$  occurs, it has affect the medium  $m_1$ . Based on GoES, we identify that  $\langle S_7 \rangle$  is dependent of  $m_1$  and this state happen on the GoOS. Hence, we are able to identify that  $OTS_1$  could be interacting with  $OTS_2$  through the medium  $m_1$ . These interactions could be safe, and may in this case, not affect the expected behavior of the  $OTS_2$ . Usually, this does not mean that they were interacting in a good way. For example, consider that a system is switching on the light and the other is turning off the light. Both are working as expected, but at the same time they are manipulating the light on an improper way, causing some inconvenience to the user (see Scenario E on Table III). This method tries to take advantage of the strengths of both techniques to be able to identify in real time FI/malfunction problems from OTS systems and possible nuisance interactions. We are working on other filters (e.g., temporal relation, localization) to provide better detection/understanding on FI causes. Spatial filters could also be used to provide better results on FI identification over different SS places, e.g., two system that share the same media but in several place, maybe not interacting.

### III. EVALUATION OF FI DETECTION APPROACHES

To illustrate the proposed FI detection approaches, we first have applied them to a simple home care use case deployed in a 3D virtual world [6]. Later, we will describe the performance of these approaches on other scenarios.

#### A. Example Scenario

The particular scenario where we applied our approaches is the following. Like every morning, Mary watches her favorite TV show and at 10 AM the Drug-Dispenser (DD) triggers the alarm buzzer, reminding her it is time to take her medicines. Unlike every morning however, the phone rings just after the dispenser alarm is triggered. She answers the phone and spends some time talking to her friend. In this scenario, we explore two possible outcomes based on the duration of the call and on what happens after Mary hangs up: one where Mary hears the DD and takes her pill (without FI - out. 1); another where Mary does not hear the DD and misses her pill (with FI - out. 2), e.g., the DD buzzer timeout period has expired during the prolonged phone call. In this assisted-living setting, equipped with a drug dispenser and a VoIP Phone (PH), we use our simulation framework [6] to collect information from sensors (e.g., capturing environment raw

Table I  
OBSERVED STATE - OUT. 1

Element:Feature	Timestamp
DD:alarm	10:30 AM
DD:buzzer	10:30 AM
U:needs_pill	10:30 AM
PH:call_in	10:31 AM
PH:ringing	10:31 AM
U:receives_call	10:31 AM
PH:call	10:32 AM
U:take_call	10:32 AM
PH:~ringing	10:32 AM
PH:~call	10:45 AM
DD:take_pill	10:46 AM
U:medicated	10:46 AM
DD:~alarm	10:46 AM
DD:~buzzer	10:46 AM

Table II  
OBSERVED STATE - OUT. 2

Element:Feature	Timestamp
DD:alarm	10:30 AM
DD:buzzer	10:30 AM
U:needs_pill	10:30 AM
PH:call_in	10:31 AM
PH:ringing	10:31 AM
U:receives_call	10:31 AM
PH:call	10:32 AM
U:take_call	10:32 AM
PH:~ringing	10:32 AM
DD:~take_pill	11:00 AM
DD:notify	11:00 AM
DD:~alarm	11:00 AM
DD:~buzzer	11:00 AM
PH:~call	11:10 AM

data like presence, audible sound, etc.) and the installed OTS systems (see Table I and II for collected information). The SHC reflective system allows us to monitor (cf. online analysis) or simulate (cf. offline analysis) the home setting and reify the state of OTS systems. Here, we run the offline analysis over information gathered through simulation on 3D OpenSIM Meta-level representation. Figure 3a presents the combined GoES of the three OTS systems in this scenario. Figure 3b and 3c show respectively, the GoOS for outcome 1 and 2. These states may be introspected through different kinds of inputs, i.e., environment sensors and specific SNMP. We consider that a FI occurs each time one or more OTS systems prevent another from executing as it has been expected by the manufacturer (i.e., as described by its GoES); For example, Mary will not take her medicine because she has been distracted and forgot it. The User's GoES depicts only relevant and expected activities (e.g., take medication, answer phone, etc.) that can be perceived through the user agenda or activity monitoring systems.

1) *Intra-System FI Detection walk-through:* Table I and II present a subset of the observed states for the previously described outcome scenarios (Figure 3b without interference and Figure 3c with interference); these outcomes were generated by a 3D simulation of the home setting between 10:30 AM and 11:10 AM. The SPA uses the GoES (Figure 3a to prune correct state sequences from the GoOS. For *Outcome 1*, the SPA performs four iterations, identifying and removing all possible GoES paths from existing GoOS:  $\langle DD:alarm, DD:buzzer, DD:take_pill, DD:~alarm, DD:~buzzer \rangle$ ,  $\langle U:needs_pill, U:medicated \rangle$ ,  $\langle PH:call_in, PH:ringing, PH:call, PH:~ringing, PH:~call \rangle$  and  $\langle U:receive_call, U:take_call \rangle$ . The SPA leaves empty GoOS, which means that no FI were detected using the intra-system approach. For *Outcome 2*: the first three SPA iterations identify and remove three subpaths:  $\langle DD:alarm, DD:buzzer, DD:~take_pill, DD:notify, DD:~alarm, DD:~buzzer \rangle$ ,  $\langle PH:call_in, PH:ringing, PH:call, PH:~ringing, PH:~call \rangle$  and  $\langle U:receive_call,$

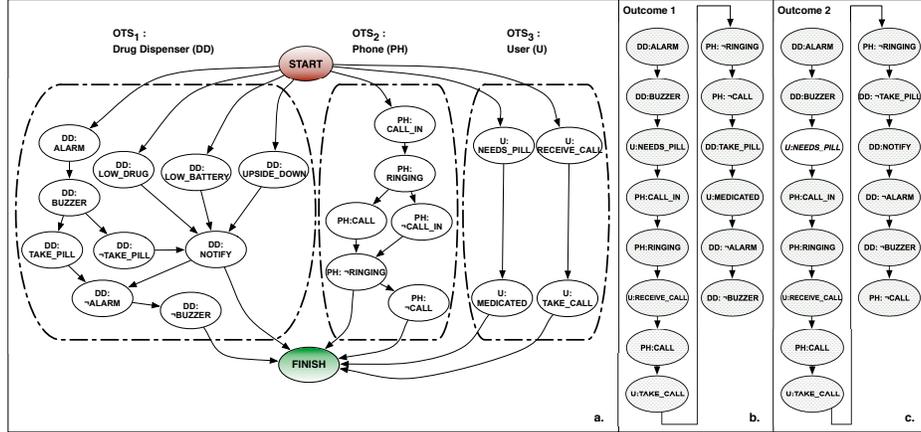


Figure 3. Graph-based representation of OTS Systems Behavior: a) GoES for Drug-Dispenser, Phone and User; b) GoOS for Out. 1; c) GoOS for Out. 2

$U:take\_call$ >. The fourth iteration of SPA returns the  $\langle U:needs\_pill \rangle$  state since state  $\langle U:medicated \rangle$  was not observed. Our approach successfully identifies a misbehavior/interference, i.e., Mary (the affected system) does not take her medication as expected. This approach however is unable to determine the possible FI causes.

2) *Inter-System FI identification walk-through:* Each OTS system graph incorporates information about, which states affect or are affected through which media; a combined parsing of all deployed OTS system graphs enable us to detect, a priori, all possible inter-system interactions. More concretely, Figure 4 shows that both the  $\langle DD:buzzer \rangle$  state and the  $\langle PH:ringing \rangle$  state affect require the user awareness, which in turn, will affect the  $\langle U:medicated \rangle$  state. The inter-system approach works like an offline searching tool; it uses this information to find that the Drug Dispenser and the Phone compete for the user awareness (shared medium), which may affect the User behavior (miss medication). This approach identifies all possible interactions between deployed OTS systems, which may or may not actually be happening/observed during a given period.

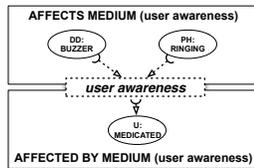


Figure 4. FI between DD and Phone via User Awareness medium

The inter-system identify that the state  $\langle DD:buzzer \rangle$  of  $OTS_{drugdispenser}$  and state  $\langle PH:ringing \rangle$  of  $OTS_{phone}$  would affect the same medium  $user\ awareness$ ; that consequently could interact with the state  $\langle U:medicated \rangle$  of  $OTS_{user}$ .

3) *Combined FI detection and identification walk-through:* In the simulated scenario, we observed (GoOS)

that the user interacted with the Phone, possibly distracting him, but did not take the medication. By combining both approaches we may detect a FI and also identify its possible cause. The intra-system approach detects that the  $\langle U:medicated \rangle$  state was not observed; then, the inter-system approach will firstly find which medium affects this state; secondly, it finds which other states could have affect the same medium during the observation period. In this case, it allows us to identify that both the Phone and the Drug-Dispenser, concurrently affected the user awareness (i.e., trigger alerts requiring the user attention) producing the FI.

### B. Application to other scenarios

We have applied our FI detection approaches (see Section II) to different scenarios, equipped with diverse systems (see Table III). For each scenario we identify possible FI outcomes (see [7] for more description of the scenarios).

#### 1) Scenarios Descriptions and Outcomes:

- *Scenario A:* This is the example scenario presented above. We identify one FI outcome:  $A1$  - Mary answers the Phone, talks to her friend over the DD alarm timeout and she forgets to take her pills (i.e., user awareness).
- *Scenario B:* Mary is watching the news on TV and, at the same time, the DD triggers its alarm. She needs to take her pills and still watch the news. Here, we identify three different FI outcomes:  $B1$  - she does not hear the DD because the TV sound overlaps the DD Alarm (i.e., user awareness);  $B2$  - she hears the DD but does not take her medicines because she does not want to move away from the TV (i.e., user will); and  $B3$  - she hears the DD ringing and moves toward it to take her pills thus missing the news (i.e., user will). All these outcomes focus on user senses and behavior: hearing, willingness and awareness.
- *Scenario C:* Mary is using her kinect application to practice the prescribed morning exercises. Meanwhile, the phone starts ringing but to answer the phone she

Table III  
COMPARISON EVALUATION OF FI DETECTION APPROACHES THROUGH SEVERAL SCENARIOS

SCENARIO		IDENTIFIED PROBLEM		DETECTION APPROACHES			
Use-Case	Outcome	Description	Category	Kolberg [3]	Intra-System	Inter-System	Combined
SCENARIO A	<i>A1</i>	user does not take her medicine	user awareness	✗	✓	✓	✓
SCENARIO B	<i>B1</i>	user does not hear	user awareness	✗	✓	✓	✓
	<i>B2</i>	user does not want to lose TV Show / move to another place	user will	✗	✓	✗ *	✓
	<i>B3</i>	user loses his TV show	user will	✗	✗	✗ *	✗ *
SCENARIO C	<i>C1</i>	user does not hear, sound overlapping	user awareness	✗	✓	✓	✓
	<i>C2</i>	user needs to move to another place, interrupt physical practice	user will	✗	✓	✗ *	✓
	<i>C3</i>	user forgets to resume physical practice	user memory failure, cognition	✗	✓	✗ *	✓
SCENARIO D	<i>D1</i>	alarm behavior influenced by HVAC	shared environment between two systems (i.e., window controller)	✓	✗	✓	✓
	<i>D2</i>	alarm is turned (causing insecurity)	user memory failure	✓	✓	✗	✓
SCENARIO E	<i>E1</i>	light intensity change (systems conflict)	shared environment between two systems (i.e., light controller)	✓	✗	✓	✓
	<i>E2</i>	user turns lights on but the media-center dims them down	shared environment between two systems (i.e., light controller)	✓	✗	✓	✓
SCENARIO F	<i>F1</i>	false alarm generated by fire control system	shared environment between two systems (i.e., window controller)	✓	✗	✓	✓

\* Note: the considered means of introspection may not be sufficient to detect some FI.

needs to move toward the phone and stop her training. We identify three different FI outcomes: *C1* - Mary does not answer the phone because she does not hear it (i.e., sound overlapping and user perception problems) and she keeps doing her physical practice; *C2* - Mary does not answer the call because she does not want to pause and resume her physical practice (i.e., user will); *C3* - Mary stops her physical practice and answers the call, but after hang-up she forgets to resume her exercises (i.e., memory failures/cognitive memory problem). These outcomes focus on the environment sound and user senses and behavior.

- *Scenario D*: Mary's home is equipped with an intrusion system and also an HVAC (cf. heating, ventilation and air conditioning) to maintain home temperature. The day is sunny and Mary goes to the elderly services center. Before leaving home she turns on the alarm. We identify two possible FI outcomes: *D1* the inner temperature is getting higher, hence the HVAC system opens the windows instead of turning air conditioning (to reduce energy consumption); the intrusion system detects the window is opening and triggers an alarm (i.e., window state/control is shared by two systems); and *D2* the inner temperature is getting higher and the HVAC system keeps opening the windows causing an opportunity for intrusion since Mary forgets to turn on the alarm before leaving (i.e., intrusion problem caused by memory/cognitive problems and HVAC system).

Systems can lead to unwanted situations as shown in the second outcome.

- *Scenario E*: Mary sits on her couch to watch a movie while the lights are reduced by the entertainment system to create a better ambiance. The home automation system (cf. domotic) uses presence detection sensors to turn lights on/off. We identify two possible FI outcomes: *E1* Mary stands still in front of her movie but the home automation system detects any small movements and turns lights on (i.e., two systems counter-sharing the lights control); and *E2* Mary deactivates lights automation and manually turns lights to be able to read a magazine while seeing her movie. However, the entertainment system reduces lights on the living room when starting the movie (both the user and the entertainment system control the lights).
- *Scenario F*: Mary leaves the house and turns on the security system. The house possesses also an active fire control system. We identify one possible FI outcome: *F1* the fire control system detects the presence of gas in the kitchen and opens the window for ventilation. At the same time, the security system triggers the intrusion alarm associated with that window opening.

We use the 3D simulation features of SHC reflective framework for generating all the presented scenarios and possible FI outcomes [6]. The proposed graph-based algorithms were then applied to each of the scenarios to analyze all particular outcomes. In the next Section, we present and discuss the

results of our approaches.

2) *Analysis of Intra-system approach:* Table III allows us to conclude, that we successfully manage to identify FI in Scenarios <A,C>, partially detect FI in Scenarios <B,D> and fail to detect FI in Scenarios <E,F>. In scenarios <A,C>, we are able to detect FI because the observed graph (cf. GoOS) was not completely pruned by the graph of expected behavior, i.e., the observed behavior does not match the expected behavior. For example in the C3 outcome, we identify that Mary stops her physical practice to take a call and after forgets to keep on doing her exercise. We detect the FI because the graphs lack one or more sensed states in the user or on the Kinect System. So we assume something unexpected happen. For Scenarios <E,F>, the observed behavior matches what is specified by the graph of expected behavior. In this particular case, even when the system is affected by a particular FI, it continues to behave as expected, preventing us from using the intra-system approach to detect this type of problem. The inter-systems might enable us, in the future, to detect and solve FI problems due to unexpected interactions between different OTS systems. Finally, in Scenarios <B,D> we did not have enough information to detect the FI, since the introspection level used was insufficient. The more information we collect from the real world, the richer will be the state graph representations, hence, better will be the FI detection. In these scenarios, we assume a realistic/feasible level of information reification. For example, it allows us to access the user agenda to know his medicine prescriptions, but does not capture the user will for watching a TV show (see B3 outcomes). But if we could increase our introspection level, we may infer the user will (e.g., based on a user profile) then we will also be able to detect this particular FI case in these scenarios. Our graph-based solution considers also the user, which might be the cause or effect of FI. In contrast, Kolberg does not consider the user in the FI analysis and does not model the media of interference [3]. Some outcomes, however, deal with issues that do not affect the expected behavior of the systems; rather, these FI outcomes result from system interactions that occur through a shared media, i.e., indirect intra-system relationships. Our inter-system approach will tackle these particular cases.

3) *Analysis of Inter-system approach:* provides us a set of all possible interactions between systems, and particularly for these scenarios provides us some good results. The only con of this approach is the subset size result that will depend on the number of present OTS systems and the media size set. When we increase both parameters, we will also increase the possible interactions between OTS systems. For some scenarios, this approach does not successfully identify the possible interactions (e.g., B2,B3,C2,C3 & D2), because we assume a feasible/realistic level of introspection, i.e., we assume we can not understand or sense the user as we do for any system using sensors or API. So, for our purposes

we consider here we cannot sense/understand the user will. Also we consider that if we can observe these parameters, our approach could detect them since these were previously represented in our graph-model.

4) *Analysis of Combined approach:* by joining both previous approaches we may not only identify FI but also understand the causes of FI. This process works like a filtering chain where the intra-system filter enables us to detect the individual OTS problems and the inter-system filter allows us to identify the possible causes of the problems. The higher the level of introspection used, the higher will be the number of interferences that we might identify (and be able to solve). We are currently improving our graph-model representation to better characterize the media interactions, so that we be able to distinguish safe/unsafe FI. By enhancing the description of interactions we may distinguish additive and subtractive FI. For example, instead of declaring that several OTS systems affect the brightness we might declare they increase or decrease it (e.g., one by opening/closing the blinds, another by turning the lights on/off). Hence, when two OTS systems are affecting the same media (e.g., light) we might predict the outcome and take corrective measures if necessary. Table IV synthesizes a classification of all FI types. We are interested not only on identifying all *possible FI* cases but also exploring the mechanisms to solve these FI occurrences.

Table IV  
CLASSIFICATION OF THE PROPOSED FI DETECTION APPROACHES

	Safe	Interaction	Unsafe
<b>Behavior</b>			
As Expected	<i>no FI</i>		<i>possible FI</i>
Not as Expected		<i>malfunction</i>	<i>FI</i>

#### IV. RELATED WORK

In [9], the authors propose a methodology to avoid interference in smart environments. They record normal user interaction patterns with the SS. Those patterns are then compared with the observed user behavior. The interference detection is based on a probability model that matches the expected and observed user activity. For example, if the user does not open the refrigerator throughout the day, as usual, this is considered an abnormal behavior and should be reported. In [3], the authors propose a solution to ease the integration of independent systems in an intelligent space. They detect resources (e.g., environmental sensors or actuators) that can be shared between systems. The intelligent space manages concurrent access to resources using system priorities and protocol interworking techniques to avoid interference. This approach focuses on interactions between different systems, but the user is not considered as a possible source of interference. In [10], the authors present a bi-directional interface between a physical Sensor-Based Sys-

tem and a 3D virtual space (e.g., Second Life or OpenSim). This middleware layer (cf. Twin-World Mediator) is used to reflect and reify changes and is also able to identify problems and interferences related with spatial requirements of devices in the SS. Similarly to [3] and [9], we manage to capture the state of the system and its applications. However, contrary to these systems, we use a graph representation that implicitly represents state sequences. Moreover, we additionally gather information about the environment and the user perception, also implicitly represented in the graphs, i.e., the possible interactions between OTS systems. Furthermore, akin to [10], our architecture also proposes the reification of base-level SS; yet, our meta-level combines the use of a 3D simulation framework with graph-based state representations [6]; this combination allows the analysis and detection of FI between independent built OTS systems. In [11], the authors propose a framework for managing FI between multiple pervasive applications that share the same physical environment. They use a coordinator not only for monitoring a shared medium, applying locks and timeouts, but also for managing the concurrent access to that medium. Therefore the coordinator must interact with existing OTS systems, introducing synchronization delays and possibly becoming a source of interference.

#### V. CONCLUSIONS AND FUTURE WORK

Current health systems will be unable to cope with the significant raise of life expectancy. Hence, new ways must be found to answer the care needs of the population and of elderly people in particular. We argue that home assisted-living settings may relieve the pressure on health systems, e.g., providing non intrusive remote monitoring capabilities, assisting on daily activities, allowing automatic collection of health parameters, endorsing the use of health promoting systems, etc. However, creating these futures SS, involves equipping homes with autonomous OTS systems that may not be easily integrated and managed. The proposed SHC reflective middleware was conceived with two goals in mind: i) manage the safe integration of OTS systems (cf. interference-free) by exploiting reflection and 3D virtual world simulation [6]; ii) provide non-intrusive pervasive interface mechanisms for home assisted-living actors. In this paper we focus specifically the first goal and explore state-graph representations to perceive FI. We demonstrate the pertinence and efficacy of the proposed approaches on different home care use cases. We explore the representation and simulation facilities of 3D virtual worlds for generating OTS systems state-graphs used to study more complex FI scenarios (cf. Prognostics). We are extending our approach to improve FI detection and resolution using new kinds of filters: spatial and temporal. Moreover, our future work will also focuses on distinguish from safe and unsafe FI and finally to perform usability tests with elderly and also caregivers.

#### ACKNOWLEDGMENT

Christophe Soares thanks the FCT - Foundation for Science and Technology, Portugal, for PhD Grant SFRH/BD/64210/2009.

This work is financed by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project Safe Home Care Project (Interference-free Home Health-Care Smart Spaces using Search Algorithms and Meta-Reality Reflection)/EIA-EIA/108352/2008.

#### REFERENCES

- [1] M. Weiser, “The computer for the 21st century,” *Scientific American*, vol. 3, no. 3, pp. 3–11, 1991.
- [2] L. Blair, G. Blair, J. Pang, and C. Efstratiou, “Feature’ interactions outside a telecom domain,” in *Workshop on Feature Interaction in Composed Systems*, 2001.
- [3] M. Kolberg, E. Magill, and M. Wilson, “Compatibility issues between services supporting networked appliances,” *Communications Magazine, IEEE*, vol. 41, no. 11, pp. 136 – 147, nov. 2003.
- [4] M. Calder and E. H. Magill, Eds., *Feature Interactions in Telecommunications and Software Systems VI, May 17-19, 2000, Glasgow, Scotland, UK*. IOS Press, 2000.
- [5] R. Morla and N. Davies, “A framework for describing interference in ubiquitous computing environments,” in *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on*, march 2006, pp. 4 pp. –635.
- [6] C. L. Velasquez, C. Soares, R. Morla, R. S. Moreira, J. Torres, and P. Sobral, “A 3D Simulation Framework for Safe Ambient-Assisted Home Care,” in *UBICOMM 2011. International Conference on*, Lisboa, Portugal, 2011, pp. 61 – 66.
- [7] C. Soares, R. S. Moreira, R. Morla, J. Torres, and P. Sobral, “Prognostic of Feature Interactions between Independently Developed Pervasive Systems,” in *PHM 2012. International Conference on*, Denver, CO, USA, jun. 2012.
- [8] K. J. Cios, W. Pedrycz, R. W. Swiniarski, and L. A. Kurgan, *Data Mining: A Knowledge Discovery Approach*, 1st ed. Springer, Feb. 2007.
- [9] V. Jakkula, D. Cook, and A. Crandall, “Temporal pattern discovery for anomaly detection in a smart home,” *IET Conference Publications*, vol. 2007, no. CP531, pp. 339–345, 2007.
- [10] H. Prendinger, B. Brandherm, and S. Ullrich, “A simulation framework for sensor-based systems in second life,” *Presence: Teleoper. Virtual Environ.*, vol. 18, pp. 468–477, December 2009.
- [11] V. Majuntke, G. Schiele, K. Spohrer, M. Handte, and C. Becker, “A coordination framework for pervasive applications in multi-user environments,” in *Intelligent Environments, 2010. International Conference on*, july 2010, pp. 178 –184.