Improved Multi-user Interaction in a Smart Environment through a Preference-Based Conflict Resolution Virtual Assistant

Kenzhegali Nurgaliyev[1], Dario Di Mauro[2], Nawaz Khan[3] and Juan Carlos Augusto[3]
[1]Department of Information Technology, Kazakh University of Technology and Business, Astana, Kazakhstan,
[2.] Department of Information Technology and Electrical Engineering, University of Naples "Federico II" Naples, Italy,
[3.] Department of Computer Science, Middlesex University, London, United Kingdom.
[1]knurgaliev@nu.edu.kz, [2]dario.dimauro@unina.it, [3]{n.x.khan, j.augusto}@mdx.ac.uk

***Abstract*— In this work we will examine and develop a system that can assist people in Activities of Daily Life (ADL). This study focuses on resolving conflicts for the requests from different users' profiles, for instance - elderly, adult and young. The objective of the system is to present a dialogue manager which is able to detect multi-user semantic conflict and to resolve the conflict for improved dialogue informing about its decisions using a system interface Avatar. The system is also able to prioritize requests that occurred among the services of multiple home appliances, as well as to deal with conflicting entities involving a single device. We investigated whether the multi-user context awareness by a Virtual Assistant adds value to the Smart Home concept in recognizing multi-user conflicts dynamically. This work has proposed a preference based method for resolving conflict and evaluated the developed system in a smart home environment.**

Keywords*— self-care; virtual assistant; sensors; intelligent environment; smart home; context-awareness; conflict resolution; Avatar

I. INTRODUCTION

Managing a proper life is a challenge, especially for people that may need an assistance to perform a task. Self-care is a very specific domain where Virtual Assistant (VA) can provide ambient assisted livings. The Smart Home concept implies a system that controls the electrical load, lighting, security, climate control, media devices. The obvious advantage of the Smart Home system is the fact that it simplifies everyday life, increases the comfort of living [1]. For example, the light in the entire house can be switched off by pressing one button or sending a request to an Avatar. In general, homes should also be warm places where people can find a comfort, but managing them may be difficult, especially for people who tend to forget to do something or for people who physically impaired [2]. A similar problem can be observed amongst teenagers that have to manage the house when their parents are at work. Technology can assist in this process; a plenty of new approaches has been published in literature [3] and they report solutions for smart homes, that should be automatic, multifunctional, adaptive and interactive [4]. Further to these characteristics, efficiency in self-care should be taken into account [5]. With these aspects, a smart home can provide support for each type of users, but it should be focused on people that particularly need guidance in order to perform a task, due to different physical or cognitive impairments.

On the contrary, other researchers [6] agree that a smart home should provide a support for each type of users. Nevertheless, this IT industry, which can be described as a mediator between human users and computer, is starting to be cost effective in promoting positive healthy attitudes [3]. Nowadays, automatic home management associates with lack of adequate systems that supports perception of certain events. From a technical point of view, this problem solution depends on multi-objective addressing. A technology like a rule-based system can be applied in order to solve multi-criteria problems in an Intelligent Environment (IE) [7]. As Augusto et al. pointed out [8], IE should incorporate principles of a correct rules delivery to a user, a continuous recognition of a situation, where system's context-awareness allows providing assistance in emergency cases as well as a privacy balance to ensure situations when a user does not need an interaction with a system.

In this paper, we propose a system that monitors actions in the smart house and provide users with customized reminders and responses. It relies on a central system to gather data from sensors placed in a smart home and to provide recommendations to inhabitants. The interaction with an Avatar has to be conducted in an intuitive way. The personalized context awareness and the resolving conflicting interaction are the main parts of this work. This project objective is to create a system that, firstly, analyzes the data and, then produce a recommendation for an action as a result of context-aware computing. Actions of planning, controlling, organizing and directing are quite important functions of overall well-being. Therefore, it can be beneficial if users are empowered to manage their lifestyle and provide their preferences to the VA. So, it is quite worthy to have a user-oriented VA that would be designed to become an independent helper and would suggest what to do in various emergency circumstances [9]. At the same time, while a Smart Home technology is about to become a widespread enhancement, this adoption might have problems with distinct intentions of users operating with such systems. Moreover, we should consider the fact that there are a variety of coexisting devices and sensors that Smart Home inhabitants want to deal with. It all creates different semantic conflicts between shared devices and users. Furthermore, complete decisions made by the system in these cases affect not only the states of devices, but multiple users' perception in future [10]. The rest of this work is organized as follows.

VA based approaches for conflict resolution is described in Section II. Section III explains the architecture of the proposed structure with the conflict resolution system. Section IV reports about a strategy we used to develop such kind of the system amongst three basic profiles of users dealing with various real house appliances. Section V demonstrates the validation and results. Finally, the Section VI draws the conclusions and lists the predictable future work.

II. CURRENT APPROACHES

A Smart Home is basically a network of actuators and sensors which should be non-disruptive regardless of request conflicts. It should follow a non-intrusive framework considering user queries. Many researchers in recent studies reported that complexities and limitations of the technology adaptation occupants of smart house are not served considerably due to semantic conflicts of requests and inability of the house to adapt the habits dynamically [11, 12 and 13].

The challenge here is to acquire the contexts of users, i.e. location, movement and then to match it with the users profile, i.e. habits, safety etc. [14, 15]. Those constraints are studied by [16, 17 and 18]. Munoz et al. [19] describes an approach for multi-user occupants where changing contexts are adapted based on different context values, i.e. temperature values, security (fire alarm), safety requirements and energy saving issues in the house.

For instance, MavHome project enables to create a context-aware framework. It automatically changes the state of devices, such as lights or air-conditioning, by setting a neutral entity for each device based on the expected users' location [20]. It is beneficial in automatic switching devices the Smart Home without human command, but, it is useless because of dealing with user location tracking factor based mainly on probability. Besides, user can have a command irrespective to changed context. Human's order depends on his mood, motivations and needs [21]. Considering this, we assumed that behavioural system should react explicitly to user's requests.

Another approach to solve various multi-user conflicts is Reactive behavioural system (ReBA). This is a context-aware application which operates with devices by assigning priority to users [22]. However, since it separates appliances for future effective operation, it means that other users with lower importance cannot influence the decision making process later on. Also, we should consider a method that required user intervention in conflict resolution. Any user-centric application can make recommendation for possible solutions, especially, for different media devices such TV, radio or smart table [23]. In this case, users are able to choose specific media service from recommendations based on their preferences. In a nutshell, the process of context finding concerns data gathering from actuators and sensors. This implies any conflict solution detection from this information using current scenario or probability on future actions based on user's feedback [19, 24]. Regarding context changes, system can act according to previously defined rules, preset policies and instructions. At any instant, these characteristics may be different, but there is a shared space – a smart house [25]. We assumed that each user ought to have opposite features in terms of the same context (a device). These features cannot be activated simultaneously; therefore there is a room for conflict. Our motivation is to create a framework that would be able to resolve predefined conflicts and to deal with all possible semantic problems between context notions in an Ambient Assisted Environment.

III. ARCHITECTURE

Technical details of the proposed agenda are given in this section. It considers a system supporting ADL interacting with users through an avatar, predefined scenarios and preferences. An Avatar has been largely studied in literatures, especially for the situations when an improved interaction should be implemented [26].

A. Users Profile and Preferences

We outlined different types of users by defining a profile for each type. The system should provide a support for each of them in an explicit way. In the following list, we summarize these profiles into:

- Adult people (26-69 ages).
- Elderly people (older 70).
- Young people (up to 25 ages).

Furthermore, we declared a list of preferences for each Profile (see Table I).

TABLE I. LIST OF ACTIVE USERS WITH PREFERENCES

ID	PROFILE	PREFERENCE (positions 0,1,2,3)
1	adult	Security, Health, Energy, Entertainment
2	young	Security, Entertainment, Health, Energy
3	elderly	Health, Security, Energy, Entertainment

We assumed that for an Adult person "Security" is in the highest priority (position 0). At the same time, "Health" issues have the most significant spot for Elderly. For Young people "Energy" savings problems has position 3 and it will be of least importance in compare to other profiles.

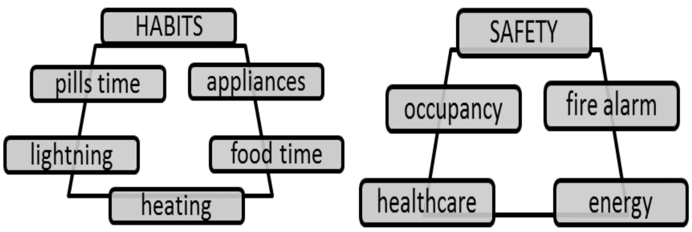
See Section III(b) for detailed explanation of the proposed preferences concept. Consequently, we explain an Avatar based support for N-users working with them simultaneously.

B. Fields of Assistance

Our system needs to differentiate user profile categories in terms of situations and required actions [27]. They are required to plan a type of the provided assistance and what kind of feedback the system expects from the user. Using various predefined scenarios and behaviours of smart home system according to a certain event - system can receive a command from the Avatar. Avatar sets a timer and continues work by schedule. Most of the functions and events would be controlled manually or via voice control, a small part is in the automatic mode. In this proposed system we have dealt with two types of scenarios: namely, Habits and Safety (Table II).

TABLE II. ACTION SCENARIOS FOR A SMART HOME ENVIRONMENT

SCENARIOS	TYPE	DESCRIPTION
scenario 1	habit	Eat food
scenario 2	habit	Take pills
scenario 3	safety	Open doors & windows
scenario 4	safety	Activity in the night
scenario 5	safety	Working devices w/o supervision
scenario 6	habit / safety	Food + Pills
scenario 7	habit / safety	Leaving the house + Pills + Set new reminder



(a) Domains involved in user habits (b) Safety domain for inhabitants

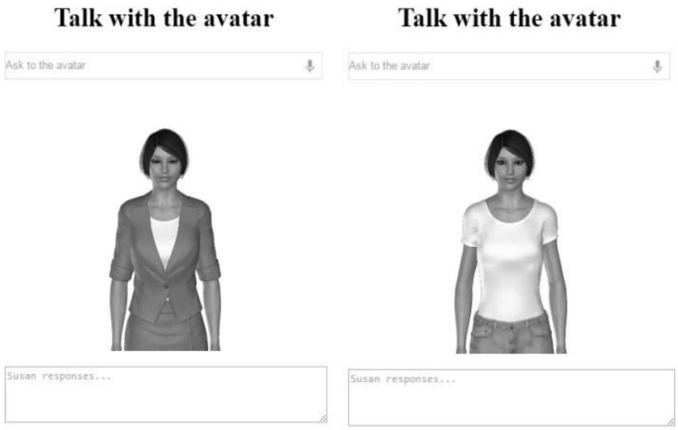
Fig. 1. Scenario types for a smart house environment.

- Habits: actions that are in a timetable and each user should follow them, e.i. time to have a lunch, etc. The timetable has actions that should be followed by each user (regardless of the profile). In other cases, it can be an event for specific users such as operating TV or radio. The system must monitor the actions in the environment and give a prompt just in case of an event is missed out (see Fig. 1a).
- Safety: the system should recognize undesirable situations that can undermine safety for both users and house (see Fig. 1b). In this category we have fire-alarm (if present), doors open (main entrance or garden door) for a long time and with activities detected in other rooms and working device in the kitchen for a long period. The system sends an alert such as (i) reminder with feedback: the system provides reminders and should check if the user received it; (ii) reminders without feedback: the system provides reminders for activities regardless gathering a feedback from the user. In terms of energy savings, this category refers to collecting data for an event such as time period for working of a floor lamp or sockets activity in a room.

As Table II shows, we fixed seven possible scenarios and assumed that a general user can be at the kitchen having his lunch and be reminded to take pills (scenario 6). Moreover, device activity and active doors in specific rooms during the night time can be assigned to safety scenarios (scenarios 3 and 4). Additionally, we considered a case when a user leaves a house, the system sends a reminder to him about time for a medicine or about a planned event. Feedback requires due to fact that user can set a reminder on later time, or cancel it all (scenario 7). Regarding scenario 5, we can form a list of devices where user cannot leave without attention, i.e. a cooker or an iron.

C. Avatar

User interaction with the framework is one of the main aspects of the study. Starting from different profiles of users, it is essential for an effective interface [28] studying the best way to deliver messages for each of them. The interaction will be managed by an Avatar [26] and the system does not require users to wear a device making it mobile and portable. VA system monitors the activity recurring to passive sensors, such as infrared motion sensors, switches (for doors and cupboards), pressure sensors for bed and sofas. The aspect of an Avatar and the related GUI (Graphical User Interface) should be adapted on the basis of profile of the users [29, 30]. Also, a general user perception of an Avatar depends on non-verbal characteristics during the dialogue [3].



(a) Assistant for Elderly and Adult (b) Assistant for a Young person

Fig. 2. Avatars retrieved from MediaSemanics.com.

Using MediaSemantics Character Builder, we came up with two basic Avatar layouts; each of them refers to three different profiles of users. As Fig. 2 depicts, the first Avatar is designed for Adult and Elderly profiles. Other one is for Young users. There is a speech recognition button at the top that receives user utterances and sends it to OpenDial (see Section III(d)). All system responses, namely, reminders and semantic results of conflict requests are displayed on the bottom box of the Avatar web-page. To start working with Avatar, user needs to push a button of this Web Speech API. At the beginning, the system would require input information about the type of user initiated operation. It sends additional questions about new user's name and age, and following appearance of the Avatar changes accordingly to age, i.e. elderly or young, as in Fig. 2a and Fig. 2b.

D. *Dialogue Manager (OpenDial Framework)*

To build our dialog system, we used the OpenDial toolkit. It provided us with a basic dialog system to start, which we extended and tested in the validation stage. OpenDial software is based on probabilistic rules (a Bayesian Network) with a trigger variable and if-then-else dialogue construction. Also, it enables to monitor current dialog state and update it when relevant changes are detected. In general, OpenDial defines our dialogue in terms of utterances (inputs) and dialogue actions (outputs): u_u - user utterance; u_m – machine utterance; a_u – user action; a_m – machine dialog action (see Fig. 3).

Fig. 3. Dialogue Manager Scheme. Users send input and receive output of the process.

In comparison with Program-O, OpenDial is more integrated with Java. Moreover, it is more robust and adaptive framework which enables to insert or remove modules without affecting the dialogue state. After User profile has defined, OpenDial stores dialog systems as domains. In addition, we can specify keywords for each request such as:

```
<condition>
<if var="A" relation="in"
value="[switch,turn,put]"/>
<if var="Dir" relation="in" value="[on,off]"/>
<if var="u_u" relation="contains" value="({A})?it
{Dir} "/>
<if var="Switchable" relation="contains" value=
"{last Device}"/>
</condition>
```

In this case, the message from users might have one out of three synonymic words (*switch, turn, put*) for device operations with different request entities (*on, off*). Later, it stores last activated device ID and user just needs to say "*Turn it on*" to have an access again (see Fig. 4).

[user]	turn the kettle on
[system]	Ok. The kettle is on (0.5) Ok. I switch the kettle on (0.5)
[user]	switch it off
[system]	Ok. I switch the kettle off (0.5) Ok. The kettle is off (0.5)

Fig. 4. Example of system responses with probabilities for action

Conditions for each command can be arbitrarily complex and effect is defined by probability of 0.5, where a cumulative effect of a condition is 1 (see Fig. 4). The same logic applied during any device activation:

```
<condition>
<if var="Act" relation="in" value="[on,off]"/>
<if var="say" relation="="value="switch{Act}({X})"
/>
</condition>

<effect prob="0.5">
<set var="a_m" value="{say}"/>
<set var="u_m" value="Ok. I switch the {X} {Act}"
/>
</effect>
```

In general, dialogue manager is capable of integrating multiple decision variables where the system can execute various actions in parallel. Our dialogue system must be robust to errors and uncertainties whilst dealing with the user input (transcriptions). In OpenDial, we declared a list of potential transcriptions with the probability of execution. To represent a set of possible answers, OpenDial uses a probability in parentheses and presents a list of alternative answers.

E. *Available Sensors & Devices*

MDX Farmside House is equipped with Router shell (VeraPlus Home Controller) which can be connected through SSH (Secure Shell protocol) using its IP address and password. It means that system is able to collect instantly all log data from each type of remote sensors in the Smart House:

- Door / Window Sensor
- Dimmable Led Bulb
- Motion Detector (Multisensor, PIR sensor)
- Plug-In Switch Sensor

To avoid unnecessary information, we worked only with following device logs: date, time, deviceVariable; deviceId; service; variable; oldValue; newValue

For the sake of simplicity, we separated all available devices on three classes in our system:

- Switchable (appliances refer to ON/OFF entities in the log)
- Absorbing (power consumable devices)
- "Can't Be Alone" (devices that should be monitored at any time in terms of safety)

As we have three user profiles, we assume that there is no necessity to detect user position in a specific room.

IV. CONFLICTS RESOLUTION SYSTEM

There are some cases when general commands or requirements stated by a user may be irrelevant to the system. User domain (Elderly, Adult or Young) is used to generate automatic profiles switch in order to reorganize overlapping dialogues. Once one of the users asks the system with similar utterance ("light on" and "light off"), system needs a clarification ("which light?" or "in which room?"). It benefits to recognize which part of the predicted dialogue node system may use.

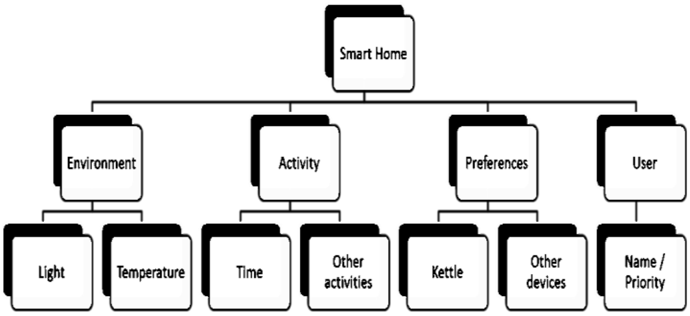


Fig. 5. Smart Home context model.

Also, the system automatically collects information about states of the devices in the area. In the case of similar user requests, it can stop the route, sending the prompt back, i.e. "your light is already on". It means that an user should utter precisely which device mentioning the location should he/she wants to operate in order to overcome conflicts described in Context Model domains (see Fig. 5). We can assess a circumstance when the user asks "to turn a kettle off" (Preferences), but it concurs with other user's time schedule for a breakfast (Activity). Another conflict example of conflicts between our context model notations can be a case when a user would like to decrease a heating level inside the house in terms of energy saving purposes. At the same time, the system prompts that it is too cold in rooms. It may have a conflict between Environment and Activity context domains. System can offer to keep the heating on at the same level, but the final decision should be made by a user.

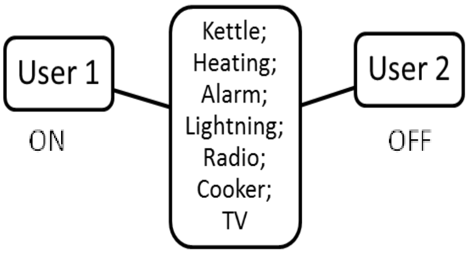


Fig. 6. Conflict case occurred with different request entities.

In addition, there might be a case when specific Agent requests an override a parallel utterance from different User. For example, Young person asks the system to operate a device, while Adult or Elderly profile give a prompt to shut this device down in the same time. It refers to a case when system relies on Profile information to give priority to the request and start to perform Adult or Elderly dialogue tree, stopping Young conversation. We proposed following Conflict Resolution system, when two or more users attempted to have an access to the same device (see Figure 6). For example, we have two requests for a kettle (device ID = 19):

- R1: User 1 wants at T_1 DevID(19) to be ON
- R2: User 2 wants at T_2 DevID(19) to be OFF

System stores each entity of requests in the Activity list. Then, Conflict Resolution system starts to operate (see Fig. 7).

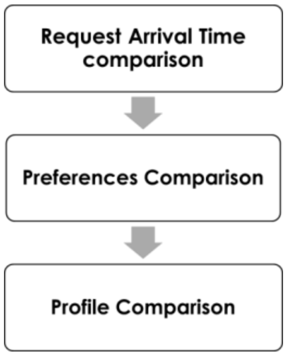


Fig. 7. Proposed Conflict Resolution System.

According to the proposed system, the Conflict Resolution method consists of three basic modules. Each of them starts to operate whenever previous stage didn't satisfy conditions to resolve a conflict. Considering all three filters we can state that all semantic clashes during the operation should be addressed to the 1^{st} or 2^{nd} stage (Request Time and Preferences Comparisons) without determining which profile has higher priority (Profile Comparison). On the contrary, if request conflict cannot be solved after checking the profile priorities, system determines randomly which user's command to perform. It can lead to significant problems which would arise a necessity of additional filter.

A. *Request Arrival Time Comparison Module*

As Fig. 8 displays, the system checks T_1 (request 1) and T_2 (request 2) whether the 2nd request come within predefined operated time for a kettle ($T_1 + \delta$).

We have created variables to describe time of usage for certain amount of devices (see Table III). Therefore, if T_2 arrives before the kettle operated time ends, thus there is still a conflict. If 2^{nd} request comes later, ie., after 150 seconds, system would prioritize this request and execute the task.

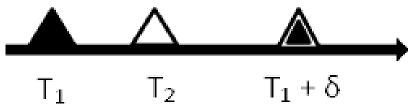


Fig. 8. Request Time Arrival timescale.

TABLE III. PREDEFINED OPERATED TIME FOR DEVICES (Δ VALUE)

Device	**δ value**
Kettle	150 sec
Microwave	30 sec
Living room light	30 sec
Bedroom light	30 sec
Radio	60 sec

We split possible Preferences on four different categories (see Fig. 9). They have been composed from [31] and they are suitable for the needs and requirements of our user profiles living in a smart house.

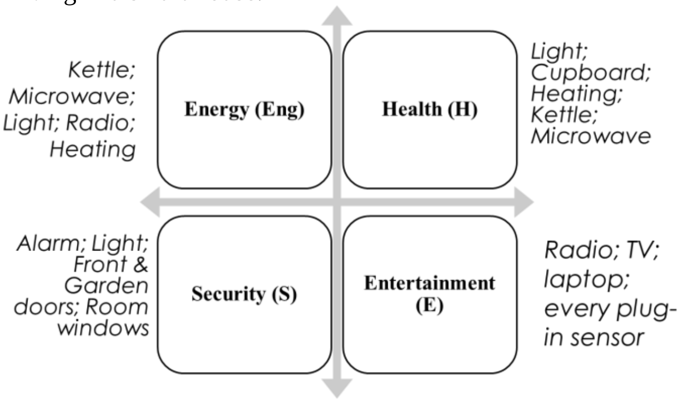


Fig. 9. Preference categories.

In addition, we assigned each preference to a list of devices, meaning that:

- Energy: *devices are supposed to be off*
- Health: *home appliances should be on*
- Security: *active doors and alarm sensors*
- Entertainment: *sensors for fun activity*

B. Preferences Comparison Module.

The system assigns a manually defined list of categories based on the chosen Profile (1).

$$Categories = \{Health, Security, Entertainment, Energy\} = \\ = Cat = \{C_i, C_j, C_h, C_b\} \quad (1)$$

Each category is different from other and has a unique entity as following (2):

$$Pref_x = (C_i \mid C_i \in Categories \wedge C_i \neq C_j \; \forall i, j \in [0, |categories|], \; i \neq j) \quad (2)$$

Let's assume that these two conflicting requests (R1 and R2) came from corresponded users (U_x and U_y). We can define unique position as index k of each n-th element of the preferences list (3):

$$pos_x(C_i, Pref_x) = k, \text{ where } Pref_x(k) = C_i \quad (3)$$

After that, system compares the position of categories in the certain list of preferences in order to prioritize the highest one. For instance, in total we have five different users' inputs with their predefined preferences (see Table IV). Therefore, two of them, U_1 and U_3, have conflicting interests. For U_1, turned on heating is about Health (position 2) preference, whereas, for U_3 switching off heating is about Energy (position 1) preference. Therefore, system will give a priority to U_3 as Energy position is more important in its set and executes it according to the following preference (4):

$$R_1 : pos_x(Cat(R1), Pref_x) > pos_y(Cat(R2), Pref_y) \quad (4)$$

TABLE IV. PREFERENCE CONFLICT EXAMPLE SOLUTION

User{preferences}	Device	Conflict	Preprocessing	Result
U1 {E, S, H, Eng}	Heating	User 1(ON) User 2(OFF)	posU2 (Health) > posU1 (Eng)	User 2
U2 {H, S, Eng, E}				
U3 {H, Eng, S, E}		User 1(ON) User 4(OFF)	posU4 (Health) = posU1 (Eng)	checks priority
U4 {S, H, Eng, E}				
U5 {S, H, E, Eng}				

In case of equal positions (the same device between U_1 and U_4), system compares whether the Health preference positioned as the highest order in the U_4 category list. In this case, U_4 request should be executed (see Section IV(c)).

C. Profile Comparison Module

To extend the resolution further, if there is no empathy between requests, system falls to a stage where it can make age comparison amongst users. Basically, system has a certain order to prioritize messages according to the Profile age (Fig. 10).

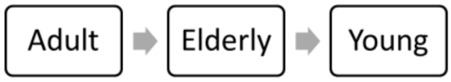


Fig. 10. Profile Priority (Adult – A, Elderly – E, Young - Y).

For our system, we gave a preference to an Adult profile in compare to others. This is because it was assumed that a Smart House relies more on Adult decisions due to wide range of expertise. This precise constraint can be changed under different context and circumstances. Relaxing a strict order of users' priority can be managed in order to give more administrative obligations to a specific profile in an emergency case or when the other profile user is alone at the Smart Home.

V. VALIDATION TEST FOR REGULAR AND CONFLICT SCENARIOS

In order to test our system in terms of quality and usability, we conducted a pilot study with four groups of three people (12 people in total). Each was assigned to a specific Profile with predefined Preferences and Habits. We assigned three possible scenarios in terms of two types of behaviour (see Table V).

TABLE V. VALIDATION REGULAR SCENARIOS

#	Type	Time	δ (delta)	Devices	Incl.
1	Habits	15:28	20 min	kettle, microwave, fridge	false
2	Habits	15:32	10 min	cupboard 1, cupboard 2	false
3	Safety	15:32	20 min	radio, light	false

System takes into account in what Time the 1st Habit should be met; in case of missing it will remind all users (empty place in User column) five more times each minute. Within this period, activity on devices (19 = kettle, 20 = microwave, 16 = fridge) is monitored. This activity should also be detected in 20 minutes (Delta column) before Time. Inclusively column with false value informs about an independence of all devices, it means that user can activate only one device out of three in order to satisfy Habits or Safety scenarios.

The same is for 2nd Habits, but we specified the User as it requires only an Elderly person. Fun activity is dedicated for an Adult user. This Adult operator might use a radio or TV, or literally to turn on plug-in sensor in the bedroom.

Moreover, during the 2nd stage of validation, we considered different conflicting cases, when all three groups attempt to get access to one device (see Table VI).

We distributed instructions, and all participants sat in front of their computers to start operating the Avatar to test different conflicting requests and have received different reminders. Their experiences about the interface (Avatar) responses in terms of likeability, usability, accuracy and speed have been surveyed.

TABLE VI. VALIDATION CONFLICT SCENARIOS

Device / Command	Profile	Preferences	Request	Result	Interpretation
light off	Y	S; E; H; Eng	OFF	no	same entity
	A	S; H; Eng; E	ON	no	Health pos. - 1
	E	H; S; Eng; E	ON	**executed**	Health pos. - 0
radio off	Y	S; E; H; Eng	ON	**executed**	Entertainment pos. - 1
	A	S; H; Eng; E	OFF	no	same entity
	E	H; S; Eng; E	ON	no	Entertainment pos. - 3
kettle off	Y	S; E; H; Eng	OFF	no	Energy pos. - 3
	A	S; H; Eng; E	OFF	**executed**	Energy pos. - 2
	E	H; S; Eng; E	ON	no	same entity

In most of responses, testers pointed out that the system is reliable enough because it supports N-users and provides fast and natural interaction with the House. Six users stated that our system can potentially combine multiple features in a single application. It tends to allow users to import data such as inserting new habits using an Avatar. All participants identified this factor as a “general advantage”, because the application can store various information regarding activities and device entities. Moreover, one of the users offers to reorganize the system as a sort of to-do list for users.

Four people wrote that it can help to plan a day for inhabitants, only three of all participants suggested that reminders about important tips in certain emergency situations can be added as a future application. One of the participants argues that our system conforms to elements of ubiquitous computing. In a nutshell, Avatar can work on desktop computer, mobile phone or tablet. Thus, all participants stated that it is great to have a user-friendly application to control all appliances in the house.

It was obvious for all participants that the system recognizes a voice for transmitting a request to the conflict resolution manager. However, there was a constraint during the test – it demonstrates a problem to process an accent and miscellaneous words in sentence construction. User should pronounce the orders clearly; hopefully they can type their commands to speech recognition box directly.

Two other users mentioned that it is still expensive technology in general, because it works only in electronically equipped environment. In addition, two participants stated importance about interaction languages. Since we used Web Speech API (English version), it can support other languages, nonetheless we need to adapt our dialogue manager for each language as well.

As a drawback, one validation participant pointed out that VA needs a permanent connection to router in order to give desired output each time when needed. Also, some of the participants (eight people) noticed that system is not able to switch a user when being operated with other one. In terms of more customized response, five people wanted to get more detailed description why their request cannot be executed, as a result of conflict resolution process.

One of the important point has been revealed during the study is - security. Of course, 10 people stated that adding face and gesture recognition feature would make this application more flexible. In case of the same preferences with no empathy and profile ages, the system goes to a protocol where it can make a choice in terms of other ambient sensors. For example, for Heating conflict the system is able to use the Temperature sensor getting information whether it is cold or warm in the house, for Light Conflict it enables the Illumination Sensor in the room to check brightness there, for Front/Garden Doors and Windows we can use Motion (PiR) sensors to detect activity, for Habits to use Pressure sensors under the bed during a night to watch time period of sleeping, for Face Recognition to use cameras.

In addition, we can equip a cooker with a distance sensor to measure presence of the user right in front of this device. This approach can be considered in our future work, where we can equip a House with these types of different sensors and actuators which will enable us to implement last reference to resolve a conflict. Moreover, during the validation we had several restrictions such as laptop-only version of the VA, and surely optimized scenarios for these cases. Thus, there is a space to enhance scenarios in terms of actions and commands.

VI. CONCLUSION AND FUTURE WORK

In this paper we introduced a dialogue conflict management framework for a smart home, in which multiple occupants can interact with various devices simultaneously through Avatar. In order to resolve conflicts among users, the proposed system maintained three-layered filtering for requests, namely - preferences of users, comparing each category of preferences for each user, matching ages of users and relating request arrival time to Avatar. This three-staged system can overcome problems with conflicts among context notions and can benefit in automatic resolution process. Overall, performing this algorithm can only be done though the context-aware system that takes information about the Smart House surroundings as input, checks for conflicting situations and then produces a new context notation there, in order to reconcile contradictory requirements or notify user about the conflicting situation.

In other words, facilitation of interaction between users and ambient environment consists of creating scenarios to define new requirements of new activities carried out by occupants of the house.

VA resolved the conflicts not in a similar fashion as normal users would do. Validation test subjects agree with fact that we need to consider different strategy for filtering the requests. At the same time, for our experimentation, users in front of their computers can simulate future conditions such as using VA through mobile phone or PC. On the contrary, in practice this system can be more integrated in Smart Home inhabitants' life to fulfill all safety requirements.

Here, in this work we emphasized on the preference based conflict resolution and the preference sets are configured by the users. However, the future direction of the work will be to develop a system which will learn from users' habit and adapt the preferences based on the users' interaction and selection and system will adapt to users' behaviour.

REFERENCES

[1] J. S. Jeedellaa, M. Al-Qutayrib, "Technological Solutions for Smart Homes. Handbook of Smart Homes", Health Care and Well-Being, Springer, 2014. pp 1-13.

[2] H. Richard, Inside the Smart Home. Springer Science & Business Media. 2006.

[3] S. K. S. E. A. James, I. Vales, "Personal healthcare assistant/companion in virtual world", AAAI Fall Symposium Series, 2009.

[4] Q. Lê, H. B. Nguyen, T. Barnett, "Smart homes for older people: Positive aging in a digital world", Future internet, vol. 4(2), pp. 607-617, 2012.

[5] G. Acampora, D. Cook, P. Rashidi, A. Vasilakos, "A survey on ambient intelligence in healthcare", Proceedings of the IEEE, vol. 101, pp. 2470-2494, 2013.

[6] M. Wich, T. Kramer, "Enrichment of Smart Home Services by Integrating Social Network Services and Big Data Analytics", 49th Hawaii International Conference on System Sciences (HICSS), 2016. pp. 425-434.

[7] Z. Meng, J. Lu, "A Rule-based Service Customization Strategy for Smart Home Context-aware Automation" IEEE Transactions on Mobile Computing, 2016, vol. 15(3), pp. 558-571.

[8] J. C. Augusto, V. Callaghan, D. Cook, A. Kameas, I. Satoh, "Intelligent environments: a manifesto". Human-Centric Computing and Information Sciences, vol. 3(1), pp. 1-5, 2013.

[9] W. Ourique de Morais, "Architecting Smart Home Environments for Healthcare: A Database-Centric Approach", Dissertation, Halmstad University Press, 2015.

[10] C. Shin, W. Woo, "Service recommendation for conflict resolution in context-aware media services", 7th Ubicomp Conference, 2005.

[11] J. C. Augusto, "Ambient intelligence: the confluence of ubiquitous/pervasive computing and artificial intelligence", Intelligent Computing Everywhere. Springer London, 2007. pp. 213–234.

[12] H. Nakashima, H. Aghajan, J.C. Augusto, "Handbook on Ambient Intelligence and Smart Environments", Springer Verlag, 2012.

[13] D. J. Cook, W. Song, "Ambient intelligence and wearable computing: Sensors on the body, in the home, and beyond" Journal of ambient intelligence and smart environments, vol. 1(2), pp. 83-86, 2009.

[14] I. Satoh, "Building and operating context-aware services for groups of users", Procedia Computer Science, vol. 5, pp. 304-311, 2011.

[15] M. McTear, T. Raman, "Spoken Dialogue Technology: Towards the Conversational User Interface", Springer Science and Business Media, 2004.

[16] H. Aghajan, R. Lopez-Cozar Delgado, J.C. Augusto, "Human-Centric Interfaces for Ambient Intelligence", Academic Press Elsevier; 2009.

[17] N. Streitz, A. Kameas, I. Mavrommati, "The Disappearing Computer, Interaction Design, System Infrastructures and Applications for Smart Environments", The Disappearing Computer, Interaction Design, System Infrastructures and Applications for Smart Environments. Springer, 2007.

[18] G. van den Broek, F. Cavallo, C. Wehrmann, "AALIANCE Ambient Assisted Living Roadmap", Ambient Intelligence and Smart Environments Series, vol 6, IOS Press, 2010.

[19] [30] A. Muñoz, J. Botía, J.C. Augusto, "Intelligent Decision-Making for a Smart Home Environment with Multiple Occupants". Computational Intelligence in Complex Decision Systems, Atlantis Press, pp. 325-371, 2010.

[20] S. K. Das, N. Roy, A. Roy, "Context-aware resource management in multi-inhabitant smart homes: A framework based on Nash H-learning", Pervasive and Mobile Computing, vol. 2(4), pp. 372-404, 2006.

[21] M. Danninger, R. Stiefelhagen, "A context-aware virtual secretary in a smart office environment", Proceedings of the 16th ACM international conference on Multimedia, pp. 529–538, 2008.

[22] A. Kulkarni, "A Reactive Behavioral System for the Intelligent Room" Doctoral dissertation, MIT, Massachusetts, USA, 2002.

[23] C. Shin, H. Yoon, W. Woo, "User-centric conflict management for media services using personal companions", ETRI journal, 2007. vol. 29(3), pp. 311-321.

[24] [31] V. Zamudio, V. Callaghan. "Understanding and avoiding interaction-based instability in pervasive computing environments", International Journal of Pervasive Computing and Communications, vol. 5(2), pp. 163-186, 2009.

[25] C. L. Oguego, J. C. Augusto, A.M. Ortega, M. Springett, "Survey on Managing Users' Preferences in Ambient Intelligence", Universal Access in the Information Society, pp. 1-18, 2017.

[26] A. Ortiz, M. del Puy Carretero, D. Oyarzun, J. J. Yanguas, C. Buiza, M. F. Gonzalez, I. Etxeberria, "Elderly users in ambient intelligence: Does an avatar improve the interaction?", Universal access in ambient intelligence environments, Springer Berlin Heidelberg , 2007. pp. 99-114.

[27] A. Crandall, "Behaviometrics for multiple residents in a smart environment", PhD Dissertation, Washington State University, 2011.

[28] R. Sharma, F. F. H. Nah, K. Sharma, T. S. S. S. Katta, N. Pang, A. Yong, "Smart Living for Elderly: Design and Human-Computer Interaction Considerations", International Conference on Human Aspects of IT for the Aged Population, Springer International Publishing, 2016. pp. 112-122.

[29] D. Hawthorn, "Possible implications of aging for interface designers" Interacting with computers, vol. 12(5), pp. 507-528, 2000.

[30] P. Jia, Y. Lu, B. Wajda, "Designing for Technology Acceptance in an Ageing Society through Multi-stakeholder Collaboration", Procedia Manufacturing, vol. 3, pp. 3535-3542, 2015.

[31] A. Aztiria, A. Izaguirre, R. Basagoiti, J. C. Augusto, "Learning about preferences and common behaviors of the user in an intelligent environment", BMI Book, pp. 289-315, 2009.