Human in the Loop: A Model to Integrate Interaction Issues in Complex Simulations

Stefano Filippi[1], Daniela Barattin[1], Francesco Ferrise[2], Monica Bordegoni[2], and Umberto Cugini[2]

[1] University of Udine - DIEGM Dept. - PIRG Group
Via delle Scienze 206, 33100 Udine, Italy
{filippi,daniela.barattin}@uniud.it
[2] Politecnico di Milano
Via La Masa 34, 20156 Milano
{francesco.ferrise,monica.bordegoni,umberto.cugini}@polimi.it

Abstract. Several activities of the product development process as for example ergonomic analyses, usability testing, and what is defined as User Experience - UX- design in general require humans to be involved as testers. In order to achieve a good effectiveness degree, these tests must be performed on prototypes as much as possible similar to the final product, and this is costly and sometimes difficult to obtain during the development process. This is especially true at the earliest stages of the process. Functional mock-up - FMU - methods and tools can be of great help, because they allow technological aspects of the products, as electronics, hydraulics, mechanics, etc. to be represented and managed in a simple and effective way. Mathematical equations allow product behavior to be determined, due to input values representing the application environment of the product. At the moment, an FMU model is great in simulating product behavior from the technological point of view, but concerns about user interaction issues are left apart. The research described in this paper aims at widening the coverage of FMU to user-product interaction issues. The goal aims at evaluating the possibility of substituting real users with a characterization of them, and to model and simulate interaction in a homogeneous way together with all the other product aspects. All of this makes the research activities very challenging, and the result is a sort of FMU-assisted interaction modeling. As an evolution of what is generally recognized as hardware and software-in-the-loop, this methodology will be referred as human-in-the-loop.

Keywords: Functional Mock-Up, Interaction, User Experience.

1 Introduction

Several activities during the product development process [1] require the involvement of real humans as testers. This is the case for example of ergonomic analyses that despite the recent interest toward the use of virtual humans in the industrial practice [2] still require some physical testing involving real users. It is also the case of any

A. Marcus (Ed.): DUXU/HCII 2013, Part I, LNCS 8012, pp. 242–251, 2013.

kind of usability testing [3], and what is defined as User Experience - UX - in general [4] especially concerning the testing of the emotional response of products. In order to achieve a good effectiveness degree, these tests must be performed on physical prototypes as much as possible similar to the final product, and this is costly and time-consuming [5]. This is especially true at the earlier stages of the process when only an idea of the product to-be is available.

Recent advances in virtual reality technologies and simulation algorithms suggest moving these testing activities from physical to virtual prototypes. Virtual prototypes can be used by humans through several sensory modalities, so as to reproduce the real interaction as best as possible. All these issues make real-time responses of simulation tools mandatory. Consequently, it comes out that simulation algorithms must be simplifications of the physical behavior of products, in order to be executed in real-time.

Functional mock-up - FMU - methods and tools can be of great help in this sense, because they allow technological aspects of the products, as electronics, hydraulics, mechanics, etc. to be represented and managed in a simple and effective way. Mathematical equations allow product behavior to be determined, due to input values representing the application environment of the product [6].

To date, FMU models have been used mainly to simulate the product behavior from the technological point of view. Mathematical descriptions of any kind of product components, from mechanical to control, are usually available in the FMU tools. However, less has been done for human behavior simulations. This implies that still the user must experience personally and physically the prototype when necessary, and this means a loss of time and higher costs by the companies. The challenge in simulating users is to be able to correctly predict all their possible behaviors, as a function of a certain phenomenon they must deal with. The user is not a machine that according to a given input always behaves in the same way. People have different needs and expectations concerning a product and they implement different ways to interact with it, according to their characteristics and abilities.

The research described in this paper aims at widening the coverage of FMU to user-product interaction issues. The goal is evaluating the possibility to substitute real users with a characterization of them, and to model and simulate interaction in a homogeneous way together with all the other product aspects. This aims at extending the concept of technological FMU to include interaction, in order to be able to simulate the characteristics and behavior of the user who turns out to be highly unpredictable and showing very different characteristics compared to the components of the current traditional FMU.

In this way real users are no longer required in the tests in which the FMU are involved, but their presence would be considered through the new component of the FMU dedicated to interaction.

All of this makes the research activities very challenging, and the result is a sort of FMU-assisted interaction modeling. As an evolution of what is generally recognized as hardware and software-in-the-loop, this methodology will be referred as human-in-the-loop.

The paper is structured as follows: Section 2 summarizes the main achievements in the field of virtual prototyping, focusing on the FMU tools; Section 3 describes the core activities of the research, the development of the FMU components devoted to interaction; Section 4 discusses the results and, finally, Section 5 draws the conclusions and plots some hints for future work.

2 Background

A mock-up, or prototype, is a product simulation used for a specific testing activity [5]. Several kinds of mock-ups or prototypes exist which are specifically built for different testing activities as for example product resistance, engineering performances, usability and UX, and they can be physical or virtual [7]. If the aim of the prototypes is to allow designers to test aspects concerning the product functionality, these are called Functional Mock-Up - FMU. Different tools exist that support the creation and use of Functional Mock-Ups. Modelica is a well-established language that allows the creation of technological FMU with particular attention to the models and simulation exchanges [8]. Modelica allows designers to simulate multi-physics and multi-domain problems in a unique environment. Many tools have been developed to simulate single, specific aspects of the product and they are currently used in the industrial practice (multi-body, FEM, etc.). More than often, product complexity requires many aspects to be simulated simultaneously; thus, recent researches aim at creating a standard FMU interface to allow different tools to communicate to each other [9]. Modelica-based tools do not still include simulations of the users in order to test the functioning of the products. In general, few human simulations are available; one example of them is used in testing the driving performance of cars. Another human simulation usually available is exploited in evaluating the thermal comfort in buildings. Anyway, these models are specifically built for companies and not generally available in open and commercial libraries. The research described in the paper aims at enlarging the domain of FMU to include the simulation of the human-machine interaction, applicable in a wide collection of situations and environments.

3 Activities

3.1 Definition of the FMU about Interaction

Real situations where user-product interactions happen are considered to highlight similarities with technological FMU. After that, the highlighted elements are analyzed and formally described, in order to make them suitable to be considered as part of future FMU.

Technological FMU simulate physical products. FMU about interaction - named hereafter FMUi - simulate the synergy between a user and a product during problem solving activities. This synergy is named situation here. Fifteen situations have been considered. Among them, there is the interaction with a photocell-driven door to access a security room, the management of the temperature in a room through a heater and the loudness control of radio equipment.

Technological FMU is made of blocks, corresponding to product components like valves, pumps, gears, etc. The predictable behavior of these components allows the corresponding mathematical equations to be defined and used in the simulation. Similarly, the FMUi should represent all the simple user-product interactions due to the user (unpredictable) and the product (predictable) behavior in every situation. Simple interaction is when only one action is involved (performed by the user or the product indifferently). For example, leveling the radio volume happens thanks to two simple interactions because the first action is to rotate the control knob (user) and the second is to change the volume level (product). Simple interactions are considered the blocks of the FMUi corresponding to the FMU ones.

FMUi blocks are characterized using two orthogonal dimensions. One sets the goal of the FMUi, the simulation of ergonomic vs. cognitive aspects. The second refers to the structure of the FMUi; it can be based on functions as for the technological ones but in some cases other formalisms, tools, etc. will be used for data processing. Once the nature of the FMUi blocks has been defined, it is the time to deal with their input, output, and structure, in order to perform a simulation of interaction as controlled and predictable as possible.

Input. The input of technological FMU is constituted by physical quantities, processed by flowing through the blocks simulating the product. These quantities are univocally measurable and this measurement is repeatable. All of this must be worth also for the FMUi. In order to simulate simple interactions, FMUi blocks need to know the characteristics that determine the user behavior, as well as the environmental conditions that could affect interaction.

User characteristics belong to three categories: ergonomics, skill and needs/expectations. Regarding the first one, height and sight are two examples, useful in dealing with the interaction with a photo-cell or with a display respectively. Memorability and dexterity are two examples belonging to the second category. They could be exploited in simulating the interaction with a teller machine (code to memorize and recall) or with a vending machine (accessibility of the delivered product). The last category could include the desired volume of a TV program or the type of linen and the ending time using a washing machine.

Thanks to this three-category characterization, users are described in the FMUi quantitatively and univocally, in order to be able to consider their description as repeatable in different situations.

Environmental conditions allow describing the context where the interaction takes place. They regard both the product itself (it is already described by the FMU but only from the technological point of view) and all the external elements it interacts with. Some examples are the height of a piece of furniture where a radio is placed on top, the color of a warning label and the darkness level in a room.

Since FMUi input is mainly related to user behavior, needs and expectations, its characterization is expressed using the user language. In this way, the generation of the FMUi can be seen as a co-design activity, where sample users are constructively involved. Consider for example the radio loudness control. Input will be the current level, the desired one and the maximum level allowed by the neighbors. Moreover, current noise level in the room must be considered. Measuring these parameters in decibel, especially the desired volume level and the room noise, would be scarcely useful because users do not get it. So, ranges are introduced. They are value intervals labeled in a user friendly way. For example, desired volume is represented by a four-value parameter: silent, corresponding to the interval 0 to 10 dB, whispers - 11 to 35 dB, talks - 36 to 65 dB and shouts - 66+ dB. In this way the user can directly associate the desired volume level to his/her need: hearing the actors' whispering in a movies, rather than having the TV set switched on just to hear some noise sometimes in the background.

Output. The output of the technological FMU is constituted by product performances evaluated thanks to the prototype. The FMUi output has a double-face nature.

From one hand, it has the same role and meaning as for the technological FMU. FMUi performances are the translation of user needs and expectations in order to make them measurable and comparable against target values. Considering again the example of the radio loudness, the output is the yes/no success of the volume leveling action performed by the user/product system given the surrounding conditions (users' and environmental).

From the other hand, FMUi output presents further aspects that mainly characterize the interaction in terms of evolution in time, and definitely distinguishes the FMUi from the technological FMU. FMUi blocks can generate numerical values, percentages, Boolean values, etc. that become known and available only at precise moments and thanks to precise interaction paths. These values can be used as input both for FMU and FMUi blocks and their timing and triggering conditions make them the personification of the simulation of interaction. For example, a FMUi block controlling the brakes of a car will generate as output both the yes/no flag of the pedal pressed and the current pressure in the braking system, available only if and when the pedal is pressed by a driver with specific characteristics and perceptions of the environment.

Structure. Technological FMU generate results starting from input data thanks to mathematical equations. Again, FMUi show the same characteristics from one hand, but they need more sophisticated tools in order to manage unpredictability and qualitative issues. If the output is a numeric value, mainly related to quantifications of ergonomic issues, etc., mathematical equations are involved. For example, if a FMUi block simulate the approaching of the user's hand/finger to the stop button riding a bus, the value indicating the distance between the hand/finger and the button is computed thanks to a mathematical equations based on anthropometric parameters (user's height, arm length, etc.). In case of Boolean or even more complex results, more articulated elaborations are required. They must be generic, repeatable and comparable as well. Results are obtained thanks to procedures exploiting logical expressions, conditional statements, etc. For example, consider a magnetic card lock system. Its interaction with the user can be simulated through an FMUi block that reads the card and opens the lock in case of success. The input is composed by the signal coming from the swipe sensor and the orientation of the card during the swiping action. The output consists of a success flag about the correct reading of the card and another value indicating the status of the lock (open/locked). The input and the statements used to compute the output are shown in table 1.

Table 1. Input and output of an FMUi block simulating a magnetic card lock system

INPUT	(Bool)swiping_in_progress, (Bool)card_orientation, (int)lock_number, (int)card_number

OUTPUT	(Bool)success(swiping_in_progress, card_orientation, lock_number, card_number)=
	IF (swiping_in_progress AND card_orientation)
	THEN
	IF (lock_number=card_number) THEN success=1 ELSE success=0
	ELSE
	success=0

	(Bool)lock_status=NOT success

3.2 Experience in the Field

A first experience in the field has been conducted to test and validate the FMUi. The test case regards the hand washing using a faucet releasing water at a pleasant temperature thanks to a photo-cell, and with the temperature controlled by a notched knob. This situation counts on four simple interactions corresponding to the following FMUi blocks: B1 - hand positioning, B2 - automatic water release, B3 - water temperature evaluation and B4 - water temperature setting.

B1 - Hand Positioning. The user approaches the faucet with his/her hands. This action depends from user's height and washbasin dimensions. This block clearly refers to ergonomic issues and a simple function is enough to implement it. In this case the output is numeric but it does not represent a performance; on the contrary, it belongs to the category of output that characterizes interaction. In fact, the hand distance becomes known and available only if and when other interaction conditions happen (the approaching of a user showing specific characteristics). Table 2 summarizes the data processing inside block B1.

Table 2. Data processing in the FMUi block B1

INPUT	(cm)user_height, (cm)washbasin_height, (cm)washbasin_depth
OUTPUT	(cm)hand_distance(user_height, washbasin_height, washbasin_depth)= user_height-washbasin_height+wasbasin_depth

For example, a user 160cm tall, interacting with a washbasin placed one meter high from the floor and with a depth of 50cm (distance between the foremost surface and the wall), will have his hands 10cm far from the faucet.

B2 - Automatic Water Release. Given the distance of the user's hands from the faucet, the faucet can release the water or not. Here the only input is this distance. There is not any other variable, human or environmental, that could influence this action (except for hardware/system failure like water shortage, photo-cell breaks, etc., not managed here). Regarding the output, the water will flow only if the hands are less than 10cm far from the faucet. If yes, the temperature of the water spilling from the faucet is given. This value is known in the system because it corresponds to the aqueduct temperature at the beginning, and to the last use of the basing afterwards. So it is used both as input and output of this block. Otherwise, a N/A value is given. Table 3 shows the data processing in the FMUi block B2.

Table 3. Data processing in the FMUi block B2

INPUT	(cm)hand_distance, (°C)water_temperature
OUTPUT	(°C)water_temperature(hand_distance, water_temperature)= IF hand_distance<10 THEN water_temperature=water_temperature ELSE water_temperature=N/A

B3 - Water Temperature Evaluation. This block simulates the user evaluation of the temperature of the water spilling from the faucet. The input consists of the current temperature and the user's desired one. It can have four values: scalding, hot, warm and cold. These are the four values used by the user to qualify the temperature using his/her language. They refer implicitly to the following intervals: equal or more than 50°C, 40°C to 49°C, 30°C to 39°C and less than 30°C.

Table 4 shows data processing in this block. Output consists of the success of the user evaluation and it also informs about the current water temperature in order to manage further iterations in case of success equal to 0.

Table 4. Data processing in the FMUi block B3

INPUT	(custom)water_temperature, (custom)user's_expectation
OUTPUT	(Bool)success(water_temperature, user's_expectation)= IF (water_temperature=user's_expectation) THEN success=1 ELSE success=0
	(°C) water_temperature(water_temperature)=water_temperature

B4 - Water Temperature Setting. The last FMUi block focuses on the interaction between the user and the faucet, aimed at adjusting the water temperature. This block is involved in any case; if there is the need to adjust the temperature because the current one does not match the user's expectation some elaboration happens, otherwise no actions are taken. If the user feels uncomfortable with current temperature, he/she operates the knob to raise or lower it. In this case the input of the block corresponds to the knob change (number of notches), the success flag coming from the previous block and the water temperature. The output is the new water temperature, looped-back to the previous block for a new evaluation. The number of notches quantifying the user action determines the required variation in the water temperature. Each notch corresponds to two degrees Celsius. Table 5 shows the data processing inside the block B4.

Table 5. Data processing in the FMUi block B4

INPUT	(notches)knob_change, (°C)water_temperature, (Bool)success
OUTPUT	(°C)new_water_temperature(water_temperature, knob_change)= IF (success=0) THEN new_water_temperature=water_temperature+knob_change*2 ELSE new_water_temperature=water_temperature

There is no success flag this time, because the action will be performed for sure and the outcome will be checked again in the block B3, as shown by the block layout in figure 1.

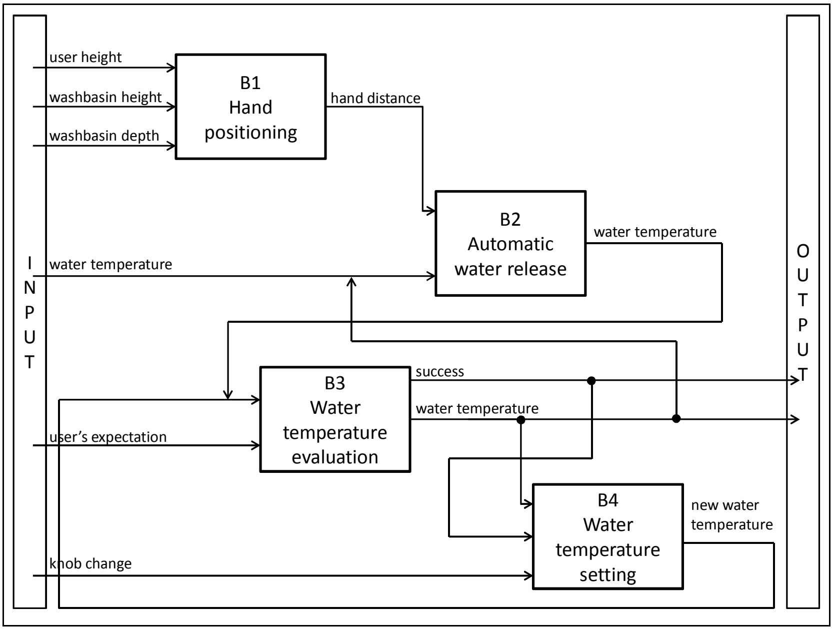

Fig. 1. FMUi simulating the interaction between a user and a faucet

4 Results and Discussion

The FMUi shown in Figure 1 simulates the interaction between a user and a washbasin currently installed in almost any public restroom. The input values can be changed as required by the prototyping situations. This model allows verifying different what-if situations and this satisfies the main reason for the FMU adoption. Regarding the pros, these early results show that user-product interaction can be simulated quite easily; moreover, the components used for this simulation appear to be compatible with technological FMU mainly because they have been generated just starting from these.

There are also some cons. FMUi blocks are scarce at the moment, regarding both the number and the definition refinement. They are too specific and the data processing suffers from two major drawbacks. The first refers to the discretization of the input values introduced by the ranges. These can determine lack of precision. Second, it is quite impossible to foresee - and consequently manage - all the combinations of input values and this could generate steady situations.

5 Conclusions

The research described in this paper aimed at extending the coverage of current functional mock-up by including the simulation of the human-machine interaction. The analysis of the traditional FMU blocks allowed input, output and structure of the new components to be defined. Some of them have been already tested and validated through an early experience in the field.

Regarding future work, the major effort will be in making the FMUi components as general as possible, in order to simulate a wide collection of situations. After this, the focus will move towards a full integration in the FMU, maybe in the form of a specific library of components devoted to the simulation of interaction.

References

1. Ulrich, K., Eppinger, S.: Product Design and Development. McGraw-Hill (2008)
2. Magnenat-Thalmann, N., Thalmann, D. (eds.): Handbook of virtual humans. Wiley (2004)
3. Filippi, S., Barattin, D.: Generation, adoption and tuning of usability evaluation multi-methods. International Journal of Human-Computer Interaction 28(6), 406–422 (2012)
4. ISO DIS 9241-210:2008. Ergonomics of human system interaction - Part 210: Human-centred design for interactive systems. International Organization for Standardization (ISO). Switzerland (2008)
5. Zorriassatine, F., Wykes, C., Parkin, R., Gindy, N.: A survey of virtual prototyping techniques for mechanical product development. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 217(4), 513–530 (2003)

6. Ferrise, F., Bordegoni, M., Cugini, U.: Interactive Virtual Prototypes for testing the interaction with new products. Computer-Aided Design and Applications 10(3), 515–525 (2013)
7. Wang, G.G.: Definition and review of virtual prototyping. Journal of Computing and Information Science in Engineering(Transactions of the ASME) 2(3), 232–236 (2002)
8. Mattsson, S.E., Elmqvist, H., Otter, M.: Physical system modeling with Modelica. Control Engineering Practice 6(4), 501–510 (1998)
9. Blochwitz, T., Otter, M., Arnold, M., Bausch, C., Clauß, C., Elmqvist, H., Wolf, S.: The Functional Mockup Interface for Tool independent Exchange of Simulation Models. In: Modelica'2011 Conference, pp. 20–22 (2011)