In-Depth Analysis of Non-deterministic Aspects of Human-Machine Interaction and Update of Dedicated Functional Mock-Ups

Stefano Filippi and Daniela Barattin

University of Udine, DIEGM, Department of Electrical, Management and Mechanical Engineering, Udine, Italy
{filippi,daniela.barattin}@uniud.it

Abstract. Increasing product complexity makes usability matters more and more important to account for in product development processes. For this reason, tools to design and evaluate interaction are studied and developed day by day. Unfortunately, user non-determinism is difficult to manage. When problems occur during interaction, users can react in several, different ways, depending from their behavioral characteristics. The research described in this paper analyzes non-determinism in depth, characterize situations where it can raise and exploits an existing tool to model and manage it as best as possible.

Keywords: Human-machine interaction, Simulation of non-determinism, Functional mock-up.

1 Introduction

Human-machine interaction is the dialogue between users and products. It describes both user behavior and product functioning, based on executed actions and feedbacks [1]. Its importance has kept increasing in the last thirty years, since products have become more and more complex and this complexity usually leads to poor usability [2]. As the ISO 9241 standard says, usability is "the effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments" [3]. Therefore, interaction design - ID - has become one of the disciplines involved in the product development process [4]. It focuses on the correct interpretation and implementation of the user-product dialogue [5] and allows generating products ready to be used easily and intuitively by the most of the users, accepted since the beginning, free of usability problems [6]. These problems are hard to foresee and manage because of the inner non-determinism of users' behavior. The Norman's model is used to highlights where, when and the reasons why interaction problems could happen [7]. The large variety of users' behavior, depending from many factors like patience, hardness to please, etc., suggests adopting simulation tools for representing interaction to speed up design activities and anticipate reviews in the product development process. The FMUi - functional mock-up for interaction - is a tool for simulating interaction studied and developed by the authors' research group in

A. Marcus (Ed.): DUXU 2014, Part I, LNCS 8517, pp. 185–196, 2014.

the last years [8]. It is involved here in trying to simulate at best non-deterministic interaction issues. All this said, the research described in this paper investigates non-determinism in user-product interaction, aiming at highlighting generic representations of user and product reactions to interaction problems and at finding the best ways to manage them. An updated release of the FMUi is proposed and used to test the results of the research.

Paper structure is as follows. Background section describes current release of the FMUi and some fundamentals of the Norman's model are given. Next section describes the first research activity, consisting in an in-depth analysis of non-determinism during interaction. The limits of the current FMUi and a proposal for an update are described afterwards. Then, an example using the updated FMUi is described. Sections discussing the outcomes of the research and summarizing conclusions and future work close the paper.

2 Background

The research described in this paper exploits a specific tool for the simulation of interaction named Functional Mock Up focused on interaction - FMUi. It has been developed by the authors' research group in the last years and described in [8]. Together with it, the Norman model is used here to describe and validate non-deterministic aspects of interaction [7]. Both of them are introduced in the following.

2.1 The FMUi

The FMUi allows simulating interaction between users and products. It has been derived from the original FMU [9, 10, 11], exclusively focused on technological issues. The FMUi can be used to test and validate interaction design solutions before the concept generation phase. FMUi models are flexible enough to allow easy reconfigurations, so many different design alternatives can be evaluated in short. An FMUi model represents all simple user-product interactions due to the user and the product behavior in every situation. There is a simple interaction when only one action is involved (performed by the user or the product, indifferently). Each simple interaction corresponds to an FMUi block. FMUi blocks contain algebraic equations, Boolean expressions, conditional statements, etc., to elaborate the input data to produce the output. Input data describe user characteristics, but also environmental conditions. User characteristics are ergonomics aspects as height, skill, and memorability, as well as needs/expectations, as desired temperature level, etc. The output allows evaluating the quality of interaction. Output values can be measurements of performances - translation of user needs and expectations in order to make them measurable and comparable against target values - and they are represented as success flag (Yes/No). On the other hand, output can consist of 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 for further FMUi blocks.

Fig. 1 shows an example of FMUi model. Each block is a black box. Designers do not need to know their content. It is enough to understand how input is transformed into output from a conceptual point of view; this allows designers to build models of interaction by combining blocks together.

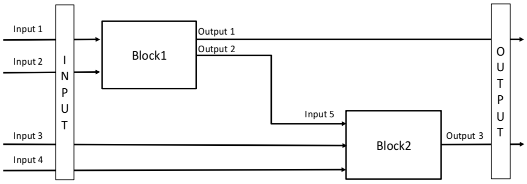

Fig. 1. Example of FMUi model

As an example, the interaction between a user and a magnetic card lock system 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

Current release of FMUi shows some limits, mainly regarding its inner determinism. All aspects of a specific interaction must be foreseen in order to build the corresponding FMUi model. Clearly, all of this goes against the simulation of a scenario as dynamic as human-machine interaction is. This is why limits of current release of FMUi are studied in details and updates are proposed, in order to represent and simulate non-determinism at best.

2.2 The Norman's Model

Norman's model describes users' activities in interacting with a system using seven stages [7, 12, 13]. In the first stage, the goals of the interaction are set. The second stage establishes the actions needed to get them. Here the execution gulf takes place. It represents possible misalignment between the actions the user would like to perform and the ones the system seems to make available. Third stage selects the actions to perform among the available ones and fourth stage performs these actions. In the last three stages, the user perceives and interprets the system state after the execution of the actions. Thanks to this, he/she should be able to claim if the goals have been obtained or not. A second gulf is present here, the evaluation gulf. This represents possible problems occurring in interpreting the system state. The two gulfs are used in this research to highlight non-deterministic aspects of interaction.

3 Activities

First activity analyzes user behavior during interaction. It mainly focuses on situations where problems arise, since non-determinism could likely emerge there. This establishes the starting point for highlighting representations of non-determinism to simulate real behavior. Then, these representations are developed and tested thanks to the FMUi. Limits of the current FMUi are highlighted first, in order to highlight current impossibility to implement the representations; after that, the needed modifications are presented. Last activity develops an example to show the exploitation of the new release of the FMUi.

3.1 Analyzing User/Product Behavior

During interaction, a problem arises when what happens does not match the user's problem solving process. In other words, there are discrepancies between the expected results and the real ones. The aim here is to analyze how users react in these cases. Norman's model is used as helping tool. The execution and evaluation gulfs represent the two moments where these discrepancies could arise.

Two different kinds of behavior could happen: the user reacts vs. he/she abandons the task and renounces to get the result he/she aims at. The second case is not considered here, because a renounce does not generate any interaction model. Anyway, the reasons why the user abandons could be interesting and they will be kept into consideration as future work.

The first case is analyzed in detail. In order to solve a problem, user can act in several, different ways. For example, consider a user who wants to close a window. Unfortunately, the window is broken; it allows only being fully opened or tilted in at the top. A user could settle down and set the tilt in position; the window is not close but less air than before flows in. Another user could search for a heavy object to place against the window frame to keep it closed. Another user could move to another room waiting for the windows to be repaired by the maintenance. In the light of the number and heterogeneity of possibilities, the research aims at highlighting interaction models able to summarize the actions the users could undertake.

As a starting point, ten situations where users run into problems are considered. Three interaction experts try to highlight all possible user behavior. The outcomes are analyzed, searching for recurring behavior.

Consider a user interacting with an oven to heat up food. If food remains too cold or becomes too hot after several attempts of temperature setting, the user settles down and eats even if the food temperature is slightly different from the liked one. This is an example of a first recurring behavior. It consists in a voluntary change of the initial user requirements. Hereafter, recurring behaviors will be named representation of non-determinism so the voluntary change of requirements is the first representation of non-determinism. Furthermore, the number of user's attempts to get the result before to change the requirements is an important indicator. This depends from behavioral user's characteristics. If he/she is demanding, this number will be high. On the contrary, a compliant user could limit his/her attempts to one or two before to settle down. Of course, more characteristics determine user's behavior other than the hardness to please; e.g., patience is another one. In the example, a demanding, but impatient, user could make one or two attempts to get the food at the right temperature, as well as the compliant user would do.

Let us consider another example now, consisting in opening a mason jar. After some unsuccessful tries, the user could act differently. He/she could warm it up, in order to exploit thermal strain, or use a knife to punch the cap to let the air flow in the jar and open it thanks to the vacuum disappearance. In both cases, there are heavy changes in the problem solving process. To see the problem solved, user prefers to change strategy instead of modifying the initial requirements. This example shows the second representation of non-determinism considered here. New actions - with related interactions - are added to the user-product dialogue, representing the changes in the problem solving process. Again, user's characteristics determine the nature and number of these actions. In the example, a patient user would likely adopt the warming up solution, while an eager would punch the cap.

Consider now a third example, where the lights in a room are controlled by a motion sensor. The interaction between the user and the product (the lights) could fail because of the morphological characteristics of the user, e.g. he/she is too short. There is no way for the user to understand why interaction fails, so he/she cannot change his/her mind about the constraints or undertake corrective actions (change strategy). The only solution applicable is to change product behavior, instead of user's one. This is the third representation of non-determinism considered in this research. The photocell could be moved or re-calibrated to detect a wider collection of user's morphologies. To avoid false-positive situations (light switched on by the presence of insects, etc.), a two-second interval of continuous presence before to switch on the lights could be required. This amount of time comes from a compromise between the sensor precision (and related cost) and the user's patience, waiting for the lights to be switched on after the entrance in the room and before starting to think about a failure. This shows that user's behavior influences product changes as well.

Finally, of course, sometimes interaction fails and there is no way to get the result, by exploiting any of the previous cases. This must be considered as well, in the definition of the representations.

3.2 Exploiting the FMUi

This section describes the exploitation of the FMUi in modeling the three representations of non-determinism just highlighted. An example of interaction is used to show if they can be already managed using current release of the FMUi or if there is any limitation.

Consider hand washing using a faucet releasing water thanks to a photocell. The FMUi model to simulate interaction is composed by six blocks and its structure is shown in figure 2.

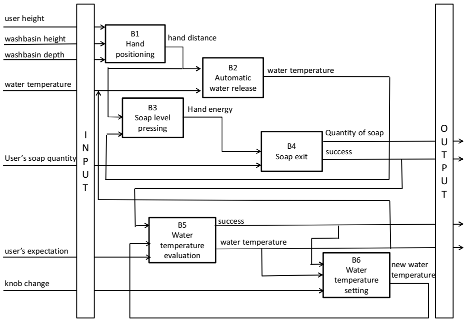

Fig. 2. FMUi simulating the interaction for hand washing

First two blocks model hand positioning and automatic water release. In the hand positioning block, named B1, the user approaches the faucet with his/her hands. This action depends from user's height and washbasin dimensions. In this case, the output is numeric and it becomes known and available only if and when precise conditions happen (the approaching of a user with specific characteristics). Block B2 models the automatic release of water. Given the distance of the user's hands from the faucet, it can release the water or not. Here the only input is this distance. 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 flowing 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 basin afterwards. Otherwise, a N/A value is given. Blocks B3 and B4 model the interaction with a soap dispenser. B3 represents the hand approaching the dispenser handle. Its input is the hand distance and the output is the force used. In B4, this value determines the quantity of dispensed soap. Another input for B4 is the quantity of soap expected by the user. Thanks to this, B4 can set a yes/no flag representing the success of interaction. If its value is equal to no, in other words if the quantity of dispensed soap does not match the user's expectation, the B3-B4 loop restarts.

Last two blocks manage the water temperature setting. Block B5 simulates user evaluation of temperature of the water flowing from the faucet. The input consists of current temperature, derived from B2, and user's desired one. This can have four values: scalding (more than 50°C), hot (40°C to 49°C), warm (30°C to 39°C) and cold (less than 30°C). The output consists of the success of user evaluation and it informs about current water temperature in order to manage further iterations in case of success equal to no. Block B6 models interaction between the user and the faucet aimed at adjusting 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 user's expectation, some elaboration happens; otherwise, no actions are taken. If the user feels uncomfortable with current temperature, he/she operates the notched knob to raise or lower it. In this case the input 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.

First representation of non-determinism, regarding slight users' changes of mind during interaction due to unsuccessful events, cannot be modeled with current release of FMUi. This because input values describing users' requirements are set at the beginning and cannot be modified on the way. They are somehow external to the model, considered as constants instead of variables inside it. This is considered here as the first limit against modeling non-determinism. It is clearly present in block B5 and B6 of the example. Once the pleasant temperature is set at the beginning, it cannot be changed and this could lead to an infinite number of user's attempts trying to reach it. Of course, this does not correspond to what happens in real life, because soon or later the attempts finish, for several reasons. The impossibility to manage infinite loops represents the second limit of current FMUi. Moreover, main reasons for loops to be finite in real life are closely related to user's characteristics. It is not easy to characterize users from the behavioral point of view. Many aspects must be considered, non-necessarily independent to each other and sometimes varying from situation to situation. Once highlighted, these aspects must be elaborated, and this requires the introduction of interval discretization, etc. Carelessness about users' characteristics is the third limit of current FMUi release to modeling non-determinism.

Second representation of non-determinism, dealing with heavy user's changes in the problem solving process, cannot be modeled using current FMUi as well. Now the FMUi model is static; once defined before starting interaction, it cannot change. All possible alternatives in getting the result of interaction must be planned before to run it. This can be found in the example considering blocks B1 and B2. If the user cannot reach the required distance to let the water flowing from the faucet, hands cannot be washed. Simulation does not allow considering alternatives, as for example the use of a footstool. This static nature of the model is tagged as fourth limit of current FMUi.

Third representation of non-determinism, focused on changes in product behavior aimed at supporting the user at best, would require modifications of the contents of FMUi blocks describing product behavior during the simulation of interaction. Once again, this is not allowed by current FMUi because of the static nature of the model. The impossibility to model product changes on the way is considered as the fifth limit.

The example and the following considerations make clear that in the current release of FMUi everything is strictly deterministic. There is no care to what could happen if an unexpected problem arises. If an action fails - in other words, if the expected result seems to be missing to the user, the FMUi can highlight the unsuccessful state of the dialogue (the success flag is equal to no) but there is no clue about user's interpretation of the error state (evaluation gulf of Norman model) and about his/her possible recovery actions (execution gulf) to proceed towards the final result, the solution of the problem.

3.3 FMUi Update

Once highlighted the limits of current release of the FMUi that prevent modeling non-determinism, this section describes the modifications introduced in order to allow it.

Modeling non-determinism requires the introduction of a supervisor, external to the FMUi model, able to manage any interaction problem. Its introduction comes by exploiting the three representations and eliminates the limits highlighted before. When an interaction problem arises, the supervisor exploits the information present in the model to propose the best solution based on specific behavioral user's characteristics. These characteristics allows the user to be described in order to understand if he/she is disposed to change his/her requirements and the amount of this change, or if he/she could change his/her mind heavier by performing alternative actions to get the goal. This way, the third limit is overcome.

The supervisor contains a counter to manage the number of iterations for loops. This, together with behavioral user's characteristics, allows deciding if one more iteration can be executed instead of exit the loop searching for alternative solutions. Thanks to this, the second limit is gone because infinite loops are not allowed anymore.

The supervisor contains a decision tree. The answers to the questions in the nodes are generated using the information present in the model as the interaction problem arises. The algorithm exploiting this tree works as follows. As soon as any of the success flag in the FMUi model becomes equal to no, the supervisor takes the helm. Based on behavioral user's characteristics and number of interactions of current loop (if any), the first section of the supervisor decides if the user likes to try interaction once again without changing his/her requirements and problem solving process or not. If yes, the counter is incremented and the control comes back to the model; otherwise the FMUi model needs to be changed. The way the changes will happen is decided by the second section of the supervisor, by exploiting the decision tree containing the three representations of non-determinism.

First, the supervisor decides if the user is disposed to change the initial requirements. If yes, the first representation of non-determinism is implemented. The structure of the FMUi has been changed in order to let input parameters act as internal variables. Their values can be changed if required and this eliminates the first limit of current FMUi.

If the user does not like to change requirements, the decision tree goes to the next node, asking for his/her willing to change the problem solving process. If yes, the second representation of non-determinism is implemented. This requires an architectural modification of the FMUi model because new blocks could be added, as well as existing blocks eliminated. A database containing implementations of functions is searched using keywords. These implementations can be considered as sorts of interaction design patterns [14]. The possibility to modify the architecture of the FMUi model once the simulation has started overcomes the fourth limit.

If the user cannot change his/her problem solving process, the only thing to do before to declare the unsuccessfulness of the interaction is to suggest changes in the product the user interacts with. This corresponds to the implementation of the third representation of non-determinism. It comes by modifying the internal structure of the blocks representing the interaction of the product. This way, product behavior changes to support the user problem solving process as best as possible. The supervisor exploits a database of technological design guidelines suggesting how to modify the product. These suggestions allow the blocks of the FMUi model to be changed, so the fifth limit of current release of FMUi goes away as well.

Finally, if behavioral user's characteristics do not allow any other iteration of the interaction and the success flag is still equal to no even after modification to the product behavior, interaction is declared as unsuccessful and simulation ends.

3.4 Exploitation of the Updated FMUi

The hand-washing example described before is involved again here to validate the updated release of the FMUi. The simulation considers a very demanding user regarding the goal to achieve (washed hands) because he is a health fanatic. This makes him disposed to perform many tries and follow different strategies as long as he gets the result. Moreover, recently he burned one hand while cooking, so he requires a precise water temperature. Finally, he is quite compliant on secondary matters not directly related to the final goal or the burn.

Simulation starts with the user approaching the faucet (block B1) and stopping at a certain distance far from it. In B2, this distance is evaluated as too high for the water to flow so interaction fails and success flag is set to no. This activates the supervisor. Based on behavioral user's characteristics, it determines that the maximum number of attempts to make the water flowing before to do something else will be two. Therefore, the counter is activated and a new iteration, representing a second user's attempt to let the water flow is fired. This attempt fails as well, because the user is still too far. Now the supervisor exploits the decision tree. The option for the user to change the requirements is not available, because the only thing he could consciously do at this point is to leave with his/her hands still dirty but abandons are not considered at the moment, as stated before. Therefore, possible user's changes of mind are taken into account. User's characteristics allow this, because he wants to wash his hands in any case. In other words, he is disposed to change his problem solving process in order to obtain the expected result. Therefore, the FMUi block database is searched for an interaction design solution. The problem is the limited user's height and this is used as keyword. A solution involving a footstool is suggested. The interaction is implemented thanks to two new blocks. One simulated the placement of the footstool close to the faucet and the second the user climbing it. This produces a new user's height, used as input for block B1. A new iteration of blocks B1 and B2 (allowed because when the decision tree is involved the value of the iteration counter is reset) now results in a successful interaction; the water flows from the faucet. Now attention moves to the interaction between the user and the soap dispenser (blocks B3 and B4). One pull of the handle gives a scarce quantity of soap, in user's opinion, so success flag is set to no. The supervisor activates and lets the user try again. No success. Another try, no success again, the quantity is still not enough. The decision tree is involved again. The user seems disposed to accept the soap quantity - he is demanding regarding the result of interaction, the hands washed, and he accepts this tradeoff to be able to get the result, also because his hands are not damaged about the scarce quantity of soap - so the implementation of the first representation of non-determinism comes to the stage. A handful (value expressed in the user's language) of soap will be changed into some drops of it and this is allowed because now the input of the FMUi model acts as internal variables and their values can be changed if required. Last part of the simulation regards the setting of the water temperature. In B5, the user evaluates it; if it does not match his expectations, the success flag is set to no and the supervisor is involved. Once again, the user is disposed to repeat the setting twice. First time water seems too cold. Then the user move the knob of just one notch but this time the temperature is too hot. Then, the decision tree is exploited again. The user cannot wash his hands with water too cold or too hot because of his burn; at the same time, he cannot change something in the problem solving process because he has no idea about how it could change, since the product offers only this way to change temperature. Therefore, the supervisor discards the implementation of both the first and second representations of non-determinism. What remains is the third one, the change of product behavior. The database of technological guidelines is searched, using setting variables as keyword. Proposed solutions focused on automatic setting and setting values closer to each other in order to simplify fine-tuning. An example is the automatic rolling shutter. A two-way button allows moving them in any position, (ideally) without the need to discretize the space. By mapping this example in the case of the faucet, the same two-way button is placed on it. One makes the water warmer, the other colder. Heaters placed in the faucet allow any water temperature to be obtained. Hence, block B4 is modified to reflect this. The number of notches as input is substituted with the button pressed (up or down) and internal data elaboration is changed. Now it manages the pressing of the button up to reach the desired temperature (or not). The success flag indicates if the right way of the button is pressed. Then simulation proceeds. In the first iteration, the success flag is equal to no (the user pressed the wrong way of the button); next iteration has the flag equal to yes and the simulation of the interaction ends with a success.

4 Results and Discussion

The new release of the FMUi avoids the five limits to non-determinism simulation highlighted in paragraph 3.2. The previous release noticed interaction problems but reactions were static. Now, on the contrary, when an interaction problem arises, behavioral user's characteristics are exploited to determine what happens. Initial input values representing user requirements can change, different interaction paths implementing alternative problem solving activities can succeed, or blocks describing product behavior can be modified, reflecting the adaptation of the product to support the user at best. The supervisor performs all of this automatically.

The simulation of interaction generates many interesting pieces of information. Looking at the FMUi model, especially by comparing it before and after the simulation, it contains suggestions about both alternative problem solving processes and product modifications; moreover, some variables as water temperature and user's height are tagged as considerable, so they will be kept into particular account. The same goes for variables describing behavioral user's characteristics. They can be weighted based on the impact/importance they had during the simulation. This stated, the new release of the FMUi is not only a tool to simulate/evaluate human-machine interaction but it is evolving toward being a design aid because it suggests solutions to solve interaction problems.

Some drawbacks must be highlighted as well. Now the block database is quite poor and not structured enough to be easily searched using keywords. Interaction designers and/or evaluators are asked to select the blocks to modify the FMUi model following the suggestions offered by the supervisor. This happens as well for the guidelines aimed at modifying the product. The database of guidelines is enough populated at the moment and the search by keywords works fine. Anyway, human intervention is required to translate the guidelines into modifications of existing blocks.

Another important drawback is the exclusive selection of one representation of non-determinism at a time. For example, a technological modification of the product in order to support better the problem solving process could imply a different strategy in solving the problem by the user. This contemporaneity cannot be managed by the new release of the FMUi.

Last negative aspect regards the use and management of the variables collecting behavioral user's characteristics. Now they are considered quite orthogonal and separated, while in real life they very often influence each other. Moreover, values of these variables are considered as constants during the simulation of interaction and once again, this does not find correspondence in the real life. A user can start interacting with something quite patiently, but suddenly he/she can become impatient because of inner or outer causes.

5 Conclusions

The research described in this paper has dealt with non-deterministic issues of human-machine interaction and their applications into a dedicated simulation tool named functional mock-up for interaction - FMUi. Norman's model helped in highlighting where, when and why users could change their mind while solving problems in interacting with products. The research has studied what could happen if interaction problems arise and generated three ways to explicate non-determinism, named representations. These allowed highlighting some limits in the current release of the FMUi that prevent it to be used for simulating non-determinism. A new release of FMUi has been proposed, where all limits seem overcome. Its characteristics and functioning have been described using an example.

Future work will focus on the new FMUi. Specifically, structure and functioning of the supervisor need to be further validated; the database of the FMUi blocks must be structured and populated while the database of the guidelines needs modifications in order to apply structural changes to existing blocks in an automatic way. Moreover, the reasons why users could abandon interaction must be kept into consideration as source of information to improve the human-machine dialogue. Variables describing the user from the behavioral point of view need to be further investigated, by taking into account any relationship/dependence among them and associating weights to define mutual importance. Weights could derive from the resources needed to accomplish them; e.g., fewer resources, more importance. The same attention should be placed to the variables representing the output of the FMUi model. Interaction goodness is not addressed now; the model is only able to say if interaction drives to a success or not.

References

1. Dix, A.: Human-Computer Interaction. Pearson Education (2004)
2. Hertzum, M., Clemmensen, T.: How do usability professionals construe usability? Int. J. Hum.-Comput. Stud. 70, 26–42 (2012)
3. ISO 9241-11. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)-—Part 11: Guidance on Usability (1994)
4. Lee, G., Eastman, C.M., Taunk, T., Ho, C.H.: Usability principles and best practices for the user interface design of complex 3D architectural design and engineering tools. Int. J. Hum.-Comput. Stud. 68, 90–104 (2010)
5. Hertzum, M.: Images of usability. Int. J. Hum.-Comput. Interact. 26, 567–600 (2010)
6. Koca, A., Funk, M., Karapanos, E., Rozinat, A., van der Aalst, W.M.P., Corporaal, H., Martens, J.B.O.S., van der Putten, P.H.A., Weijters, A.J.M.M., Brombacher, A.C.: Soft Reliability: an Interdisciplinary Approach with a User-System Focus. Qual. Reliab. Eng. Int. 25, 3–20 (2009)
7. Norman, D.A., Draper, S.W.: User Centered System Design; New Perspectives on Human-Computer Interaction. L. Erlbaum Associates Inc., Hillsdale (1986)
8. Filippi, S., Barattin, D., Ferrise, F., Bordegoni, M., Cugini, U.: Human in the loop: a model to integrate interaction issues in complex simulations. In: Marcus, A. (ed.) DUXU/HCII 2013, Part I. LNCS, vol. 8012, pp. 242–251. Springer, Heidelberg (2013)
9. 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, 513–530 (2003)
10. Ferrise, F., Bordegoni, M., Cugini, U.: Interactive Virtual Prototypes for testing the interaction with new products. Comput. -Aided Des. Appl. 10, 515–525 (2013)
11. Enge-Rosenblatt, O., Clauß, C., Schneider, A., Schneider, P.: Functional Digital Mock-up and the Functional Mock-up Interface – Two Complementary Approaches for a Comprehensive Investigation of Heterogeneous Systems. In: Proceedings 8th Modelica Conference, Dresden, Germany, March 20-22, pp. 748–755 (2011)
12. Norman, D.: The Design of Everyday Things: Revised and Expanded Edition. Basic Books (2013)
13. Ag. Ibrahim, A. A., Hunt, A.: An HCI Model for Usability of Sonification Applications. In: Coninx, K., Luyten, K., Schneider, K.A. (eds.) TAMODIA 2006. LNCS, vol. 4385, pp. 245–258. Springer, Heidelberg (2007)
14. Gangemi, A.: Ontology design patterns for semantic web content. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 262–276. Springer, Heidelberg (2005)