Experience with a Model-based Safety Analysis Process for Autonomous Service Robot

Damien Martin-Guillerez, Jérémie Guiochet, David Powell

Experience with a Model-based Safety Analysis Process for Autonomous Service Robot

Damien Martin-Guillerez*†, Jérémie Guiochet*†, and David Powell*†
* CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France
† Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France
Email: *firstname.lastname@laas.fr*

***Abstract*— Safety is a major concern for autonomous systems that physically interact with humans, such as service robots. However, modeling dynamics of such systems is hard so classical safety analysis methods need to be adapted. In this paper, we propose an approach based on a combination of well-known safety analysis techniques. We propose to describe scenarios of use with the common Unified Modeling Language. Risk analysis is then performed using a Preliminary Hazard Analysis, an adaptation of the HAZOP method and the classical Fault Tree Analysis. This paper explains the overal process and illustrates it through the exemple of the MIRAS projects which aims to develop a robotic strolling assistant that will help disabled persons to stand, sit and walk.**

Safety assessment process, risk assessment, autonomous systems

I. Introduction

Safety is now a major concern in many computer-based systems and more particularly for autonomous systems such as service robots in physical contact with human. The traditional approach to analyze safety of such systems is to use methods such as Fault Tree Analysis (FTA) or Failure Mode, Effects, and Criticality Analysis (FMECA). Those methods are usually based on models of the systems such as block diagrams or functional decomposition and automata for dynamics. For autonomous systems it is impossible to represent dynamics with automata as they evolve in an unstructured environment, including humans. A functional decomposition of the decision architecture is also impossible. Moreover, the fact that environment is not structured means that the number of operating conditions is essentially infinite.

We propose an approach to cope with these issues through the combination and adaptation of several well-known techniques. We consider that the earliest models of a system usually describe scenarios of use of the system. We claim that the analysis of deviations of such scenarios allows the identification of major risks. We propose to describe scenarios of use with the common Unified Modeling Language (UML [1]), and to analyze risks using with a Preliminary Hazard Analysis (PHA), the guideword-based collaborative method HAZOP (HAZard OPerability) [2] and Fault Tree Analysis (FTA) [3]. A major advantage of using UML as input model is that it is now a *de facto* standard for system description, and non-experts can easily understand diagrams such as sequence and use-case diagrams. HAZOP analysis is also well-adapted to the initial steps of the development as it is easily understandable, and through the use of guidewords, it enables a systematic analysis.

This process has been successfully applied to robotic projects and we illustrate each section with the MIRAS project [4]. The objective of this project is to develop an assistive robot for standing up, sitting down and walking, and also capable of health state monitoring. It is designed to be used in elderly care centers by people suffering from gait and orientation problems. It is composed of a mobile base and a moving handlebar (Figure 1).

Fig. 1. Robuwalker – First prototype

This paper is structured as follows. We present our general process in Section II. This approach is detailed step by step with its application to the MIRAS project in Section III. In Section IV, we discuss the validity of our approach. Section V gives an overview of a tool we developed to support this approach. We present related work in Section VI. Section VII concludes this paper.

II. Method overview

In safety critical systems, safety assessment is usually performed using a safety assessment process [6] where the objective is to reduce the risk to an acceptable level. This process and its terminology is now quite stable in industrial standards [7]. It is based on a decomposition of activities into risk analysis, risk evaluation and risk reduction [5]. Risk analysis aims to identify hazards and estimate the risk. Risk evaluation is a step for comparing the estimated risk against given risk criteria to determine the acceptability of the risk.

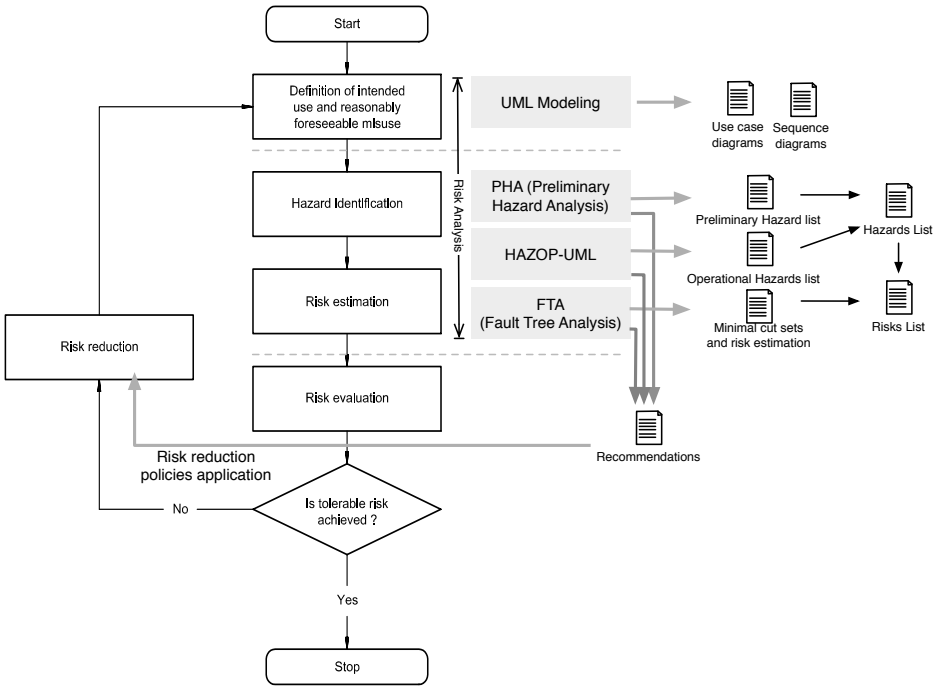


Fig. 2. Adaptation of the standard safety assessment process [5] and its generated artifacts

Risk reduction is a process in which decisions are made and measures implemented by which risks are reduced to, or maintained within, specified levels. In order to complete risk analysis many techniques have been developed (Fault Tree Analysis, Failure Mode Effects and Criticality Analysis, Event Tree, etc.) and applied in very different safety critical domains from nuclear power-plants to medical robots.

In this context, we based our approach on the classical safety assessment process as described in Figure 2. The same cycle is repeated until the designed system achieves tolerable risk. When applying this process, it is important to:

- describe the target of evaluation at the right level of abstraction;
- facilitate communication and interaction between different stakeholders involved in the safety assessment process (e.g., in our case, the stakeholders are the patients, the medical staff, and the robotics experts);
- manage the combinatorics of risk analysis, which often results in an excessive number of documents and models;
- document safety analysis results and the assumptions on which these results depend to support reuse and maintenance.

We chose a subset of UML to describe the system, communicate with the stakeholders and organize safety analysis documents. UML is a standard general-purpose modeling language that includes a graphical notation enabling the representation of an abstract model of a system [1]. The UML model of a system is composed of different UML diagrams, each of which is a partial graphical representation of the system that concentrates on a particular viewpoint. Two diagrams are commonly used for description of the system usage: use case and sequence diagrams. Use cases represent intended use of the system and are linked with the actors that can trigger scenarios of the use case. Each use case is further documented by fields such as pre and post conditions. Each sequence diagram represents one particular scenario of one use case. Those two diagrams, as presented in the process view of Figure 2, are also the input for hazard identification step.

Hazard identification is then performed using a Preliminary Hazard Analysis (PHA [8]) and an adaptation of HAZOP and UML [9]. Through the HAZOP method, a system is analyzed by holding a review of the systematic generation of deviations defined by the conjunction of parameters of the system (e.g., pressure, temperature...) and guidewords (e.g., no, more, less...). We apply the PHA in a standard way but adapt the HAZOP method to apply it to UML use cases and sequence diagrams.

Risk estimation is then carried out after the HAZOP analysis. Quantitative risk estimation is however impossible to obtain, mainly because of the impossibility to evaluate probabilities of identified hazards. However, it is important to propose risk reduction means even if probability is not calculated. We cope with this problem by defining different objectives regarding the iterations of the development process. For instance, during the first iteration, only severity (see Table I) is considered, and when the cost (in term of design changes impact, development efforts) is acceptable by robotic experts, risk reduction is carried out. As a result of the first iteration, the document *Hazards List* is produced (see Figure 2). The next step is based on the PHA and HAZOP-UML hazards, which are analyzed with the Fault Tree Analysis (FTA). This leads to an estimation of the final risks (document *Risks List* in Figure 2) and also to new recommendations.

III. Safety process steps

In this section, we detail the various steps of the safety process. These steps are illustrated by the application in the MIRAS project.

TABLE I

SEVERITY LIST USED IN THE MIRAS PROJECT, DERIVED FROM THE ABBREVIATED INJURY SCALE OF [10])

Num.	Severity	Type of injury	SIL
0	None	None	0
1	Minor	Superficial injury	0
2	Moderate	Recoverable	0
3	Serious	Possibly recoverable	1
4	Severe	Not fully recoverable without care	2
5	Critical	Not fully recoverable with care	3
6	Fatal	Not survivable	4

A. Definition of intended use

In the first step, the intended use of the system is modeled using UML use cases and scenarios with UML sequence diagrams. The use cases are completed by conditions of use: preconditions to be fulfilled before any action of the use case can be performed, postconditions to be satisfied at the end of the use case and, invariants that must hold during the use case. Interactions are represented by messages in sequence diagrams. Messages can be annotated (e.g., to specify whether the interaction is physical or cognitive).

These diagrams are often used for high-level representations of a system. They are easy to understand even by non experts so they are suitable as a means for presenting the scenarios to the different stakeholders. Furthermore, they can be included as part of the documentation for the certification process. Sequence diagrams are highly expressive yet can remain quite simple when used to describe use scenarios. This simplicity makes them an attractive support for hazard identification by deviation analysis since it helps to keep the combinatorial aspects of such analysis under control. Throughout the modelling process, preliminary safety remarks and recommendations can be issued.

In the MIRAS project, we identified 11 use cases: *Strolling* (UC01), *Standing up operation* (UC02), *Sitting down operation* (UC03), *Balance loss handling* (UC04), *Summoning and autonomous movement of the robot* (UC05), *End of use detection and movement to a waiting position* (UC06), *Positioning the robot by hand* (UC07), *Alarm handling* (UC08), *Patient profile programming* (UC09), *Patient profile learning* (UC10), and *Robot set-up* (UC11).

By way of an example, Table II list the conditions put on use case UC02 (*Standing up operation*) and Figure 3 describes the nominal scenario for this use case. This sequence diagram depicts the interaction between the users and the robot as the potential source of harms. Robot decisions are represented as self-messages.

B. Hazard identification

To identify hazards that can arise from the use of the robot, we used two complementary techniques: Preliminary Hazard Analysis (PHA) and UML-HAZOP (UML - HAZard OPerability).

A standard PHA [8] is applied at an early stage of the design. The various stakeholders of the project meet together for several workshops (two in the MIRAS project).

TABLE II

UC02 "STANDING UP OPERATION"

Use case name	UC02. Standing up operation
Abstract	The patient stands up with the help of the robot
Precondition	The patient is sitting down The robot is waiting for the standing up operation Battery charge is sufficient to do this task and to help the patient to sit down The robot is in front of the patient
Postcondition	The patient is standing up The robot is in admittance mode
Invariant	The patient holds both handles of the robot The robot is in standing up mode Physiological parameters are acceptable

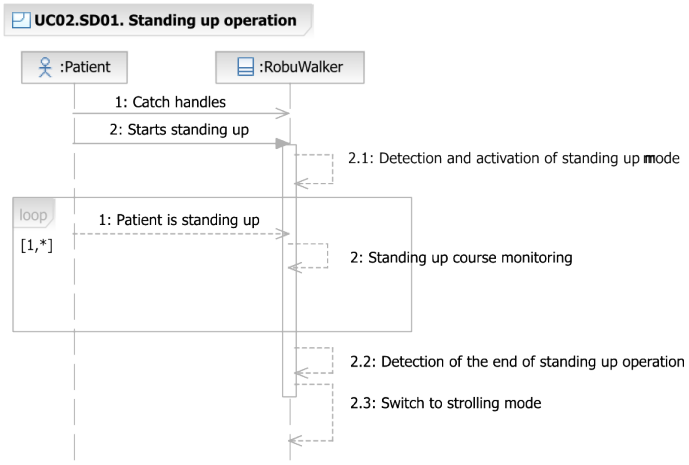


Fig. 3. Sequence diagram UC02.SD01 giving main scenario of UC02 "Standing up operation"

During the workshops, participants try to consider all the possible causes of hazards in the system (e.g., environmental, electrical, mechanical, hardware/software and human). For each cause, the participants identify the hazards that can arise. In the MIRAS project, the PHA led to the identification of 45 hazards (Tab. III).

We then apply the UML-HAZOP method. We adapt the HAZOP [2] method to analyze deviations of the UML use cases and sequence diagrams. According to the Defence Standard 00-58 [11], HAZOP analysis is the systematic identification of every deviation of every *attribute* of every *entity*. Each deviation is a potential hazard that can lead to a harmful event. We adapted the guideword lists to apply them to attributes of use cases and sequence diagrams. The guideword list we use for the use case entity is given in Table V and an extract of the analysis of UC02 (*Standing up operation*) is presented in Table VII. All guidewords are applied to generate deviation. The analyst then establishes the effect at the use case level, and the result in the real world. The other columns of the table guide the analyst to establish a severity level, to deduce requirements and otherwise make remarks on that deviation. The complete method is presented in [9].

In the MIRAS project, the analysis of 297 deviations led to the identification of 13 hazard classes (Table IV). This table presents the main hazardous situations of the system.

TABLE III

EXTRACT OF THE HAZARDS IDENTIFIED BY THE PRELIMINARY HAZARD ANALYSIS (45 TOTAL)

Project: MIRAS	**PHA Hazards**		**Date:** **Prepared by:**	28/10/09 Jérémie Guiochet Damien Martin-Guillerez			
Number	**Hazard**	**Categories**	**Who?**	**Solution/Answer**	**Version**		
					Dev	**Eval**	**Final**
H13	Wet ground / Risk of grip loss	Environments					X
H27	Electrical current drop / Risk of sudden reboot with wrong parameters.	Hardware	Robosoft			X	X
H40	Robot runs over patient's foot	Mechanical	Robotsoft	Wheel diameter is reduced to 100mm. Foot detection using IR sensors.		X	X

TABLE VII

EXTRACT OF THE UML-HAZOP ANALYSIS TABLE OF UC02 "STANDING UP OPERATION"

Project: MIRAS **HAZOP table number:** UC02 **Entity:** UC02					**Use case description** **Use case name: Standing up operation**				**Date:** **Prepared by:** **Revised by:** **Approved by:**	04/08/09 Damien Martin-Guillerez Jérémie Guiochet	
Line Number	**Element**	**Guideword**	**Deviation**	**Use Case Effect**	**Real World Effect**	**Severity**	**Possible Causes**	**Integrity Level Requirements**	**New Safety Requirements**	**Remarks**	**Hazard Number**
15	Battery charge is sufficient to do this task and to help the patient to sit down (precondition)	No/none	Battery charge is too low but the robot starts the standing up operation	The robot interrupts its movement (standing up or walking)	Loss of balance or fall of the patient	Serious	HW/SW Failure Specification error	Battery charge sensors should be SIL2	Worse-case electrical consumption must be evaluated beforehand. Take the lower bound of the battery charge estimation	If the robot stops during standing operation, the most probable scenario is that the patient will fall back on the seat.	2,6
16		Other than	**cf L15**								
17			Battery charge is high enough but the robot thinks otherwise	Robot refuses to start stand up operation	Patient is confused	None	HW/SW Failure Specification error	None	None		

TABLE IV

HAZARD CLASSES AND THEIR ASSOCIATED SEVERITIES

Num.	**Description**	**Severity**
HN4	Fall of the patient without alarm or with a late alarm.	Severe
HN5	Physiological problem of the patient without alarm or with a late alarm.	Severe
HN6	Fall of the patient caused by the robot.	Severe
HN7	Failure to switch to safe mode when a problem is detected. The robot keeps moving.	Severe
HN1	Incorrect position of the patient during robot use.	Serious
HN2	Fall of the patient during robot use.	Serious
HN3	Robot shutdown during its use.	Serious
HN8	Robot parts catching patient or clothes	Serious
HN9	Collision between the robot (or robot part) and the patient.	Serious
HN10	Collision between the robot and a person other than the patient.	Serious
HN11	Disturbance of medical staff during an intervention	Moderate
HN12	Patient loses her balance due to the robot	Moderate
HN13	Patient fatigue	Minor

TABLE V

ATTRIBUTES, GUIDEWORDS AND INTERPRETATIONS FOR USE CASE ENTITY IN THE UML-HAZOP METHOD

Entity = Use Case		
Attribute	**Guideword**	**Interpretation**
Preconditions / Postconditions / Invariants	No/none	The condition is not evaluated and can have any value
	Other than	The condition is evaluated true whereas it is false The condition is evaluated false whereas it is true
	As well as	The condition is correctly evaluated but other unexpected conditions are true
	Part of	The condition is partially evaluated Some conditions are missing
	Early	The condition is evaluated earlier than required (other condition(s) should be tested before) The condition is evaluated earlier than required for correct synchronization with the environment
	Late	The condition is evaluated later than required (condition(s) depending on this one should have already been tested) The condition is evaluated later than required for correct synchronization with the environment

In the HAZOP analysis, each deviation that potentially leads to a hazard class is labeled with the corresponding number (column "Hazard Number", Table VII). Table VI) gives an extract of the list of recommendations resulting from application of the UML-HAZOP method. This list is derived from the "new safety requirements" column of the UML-HAZOP tables.

C. Risk estimation

During the first iteration of the process, a qualitative approach is followed to estimate the risk associated with each hazard class (severity column of Table IV). Safety Integrity Level (SIL [12]) requirements are estimated when

TABLE VI

EXTRACT OF RECOMMENDATIONS ISSUED DURING THE UML-HAZOP ANALYSIS (26 TOTAL)

Project: MIRAS	**HAZOP Recommendations**	**Date:** **Prepared by:** **Revised by:**	28/10/09 Damien Martin-Guillerez Jérémie Guiochet		
Number	**Description**	**Solution/Answer**	**Version scope**		
			Dev	**Eval**	**Final**
Rec1	The standing-up profile should be validated by a human operator	The profil can be validated only after at least one try			X
Rec2	Worst-case elictrical consumption must be evaluated beforehand.	Display of the mean battery time left by the robot.			X
Rec22	Send regularly a network heartbeat from the robot to the medical staff control panel. Launch alarm on time-out.				X
Rec26	Plan a procedure to check that the robot has correct knowledge of the environment.	Display of the map used by the robot on the control panel.			X

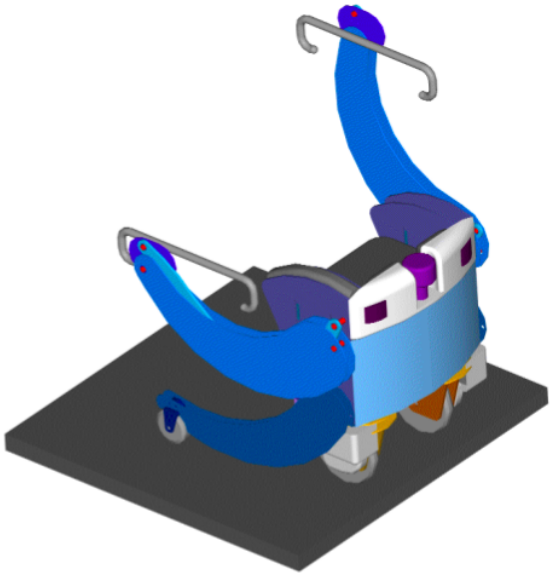


Fig. 4. Mechanical design of the second prototype of the MIRAS robot

performing HAZOP. To do this, an indicative SIL is assigned to each severity level (cf. Table I , SIL column) and thereby to safety-related components associated with each HAZOP deviation (cf. Table IV, integrity level requirements column) according to the most severe hazard induced by this deviation (e.g. the patient position detection system should be SIL1).

During the second iteration of the safety assessment process, we extract FTA top events from the hazard class list of Table IV. The FTA gives top event occurrences and enables the computation of quantitative risk. Fault tree analysis was carried out in the usual way so, given the space limitation, we do not detail it there. The only point we want to stress is the link between FTA and the other artifacts of the safety analysis process.

D. Risk evaluation

As mentioned before, risk evaluation consists in comparing the estimated risk with given risk criteria to determine the acceptability of the risk. Even if some proposals have been made for acceptability criteria [13] there is no set of generally accepted risk acceptance principles for service robotics. This is an important issue but it deals with political and ethical concerns. Nevertheless, criteria are needed for engineers to determine which risks have to be reduced. This step is again an iterative process. In the MIRAS study, after the first iteration (coming after a preliminary risk estimation only considering hazard severity), each hazard and possible recommendations were proposed to the robotics experts, and on the basis of our risk analysis results they proposed a classification of risk acceptance according to three versions of the robot : *development* version for using the robot in the laboratory, *evaluation* version for the prototype used in hospitals for clinical evaluation, and *final* version for operational life.

In a second iteration of this activity, the final risks as presented as in Table IV with a severity and an occurrence estimation are classified using the ALARP (As Low As Reasonably Practicable) principle. Three risk level zones are defined : intolerable region, the ALARP region, and the broadly acceptable region. A risk can stay in the ALARP region only if further risk reduction is impracticable or if its cost is disproportionate to the improvement gained. For each risk, the corresponding sources of deviations and hazards, and the associated recommendations are evaluated for judging acceptability. The final result is an argumentation for the final acceptable risk of the system.

E. Risk reduction

Recommendations were issued as a result of each step of the safety analysis process. UML modeling, PHA and UML-HAZOP give rises to general recommendations. Integrity level requirements are issued during UML-HAZOP analysis. A specific integrity level requirement leads to specific safety recommendations given by IEC 61508 [12]. These recommendations are applied to reduce risks if their current level is not tolerable. Of course, design recommendations should be applied first, then protective devices then, if no other solution exists, information for users [6].

Some risk reduction techniques will reduce the severity of the corresponding hazard (e.g., a bumper added to the robot will reduce the severity of a collision) while others will reduce the probability of occurrence (e.g., infrared sensors to detect the patient's feet will decrease the probability of a collision between the patient and the robot). The next version of the MIRAS robot (Fig. 4) is now under validation. In addition to various ergonomic corrections, the MIRAS robotics experts[1] took into account the risk reduction recommendations resulting from the first cycle of our safety analysis.

[1]ISIR - Institut des Systmes Intelligents et de Robotique (http://www.isir.fr) and ROBOSOFT (http://www.robosoft.fr)

The new version reduces the severity of hazard classes 8, 9 and 10 and reduces the occurrence rate of hazard classes 1, 6, 7, 8, 9 and 11 (Tab. IV, columns Severity and Occurrence rate). This new version especially applies the recommendations and meets the requirements asked for the *evaluation* version of the robot (i.e. a robot that will be used in the laboratory and used for clinical evaluation).

IV. Evaluation of the method

We evaluate our approach from four different perspectives: **integrability** into the development process, and **usability**, **validity** and **applicability** of the method.

Integrability: in our approach, UML design models are shared with the development process. Deviation analysis can be carried out at the same time as design refinement or testing/coding by the development team. The results of risk assessment and of testing can be used to generate another cycle (and modifying either the design or the implementation) or to accept the prototype as a final version. Another noticeable point is that the early integration of the safety assessment process enables the design to be modified as a result of identified risk reduction recommendations.

Usability: the overhead of using this method in the overall process is quite low. For instance, in our first iteration of the cycle, apart from the design that was shared between stakeholders, the only critical path overhead that was induced by the meetings specific to the risk assessment process: two workshops of two hours each for the Preliminary Hazard Analysis and two sessions of two hours to present the outcomes of the UML-HAZOP Analysis and the assessed risks. The time devoted to the safety analysis itself (not to the critical path) consisted of the time to prepare guidelines for PHA workshops (a few hours), the time to carry out the UML-HAZOP analysis (2 weeks) and the time to format the results (a few weeks). UML-HAZOP already proved to be usable even by hand but would be even more so with a tool to assist book-keeping and result formatting [9].

Validity: Our approach relies on the UML-HAZOP method that identifies a large set of operational hazards. Other hazards, especially environmental hazards should be covered by PHA. Of course, all hazardous situations cannot be forseen however much effort is put into safety analysis. Nevertheless, using these techniques, all classical hazards of robots were identified and several new hazards were discovered (e.g., incorrect position of the patient) together with the corresponding misuse or system failure. This gives use confidence that the coverage of UML-HAZOP and PHA are sufficient to identify most hazards of service robots.

Applicability: the first iteration of our risk assessment process leaded to conclusive evidence that the first prototype of the MIRAS robot was unsafe. The hazard list and the recommendations issued were accepted by the robotic experts in the MIRAS project. Our recommendations were taken into account in the design of the next prototype (which also includes several ergonomic changes).

V. CASE Tool description

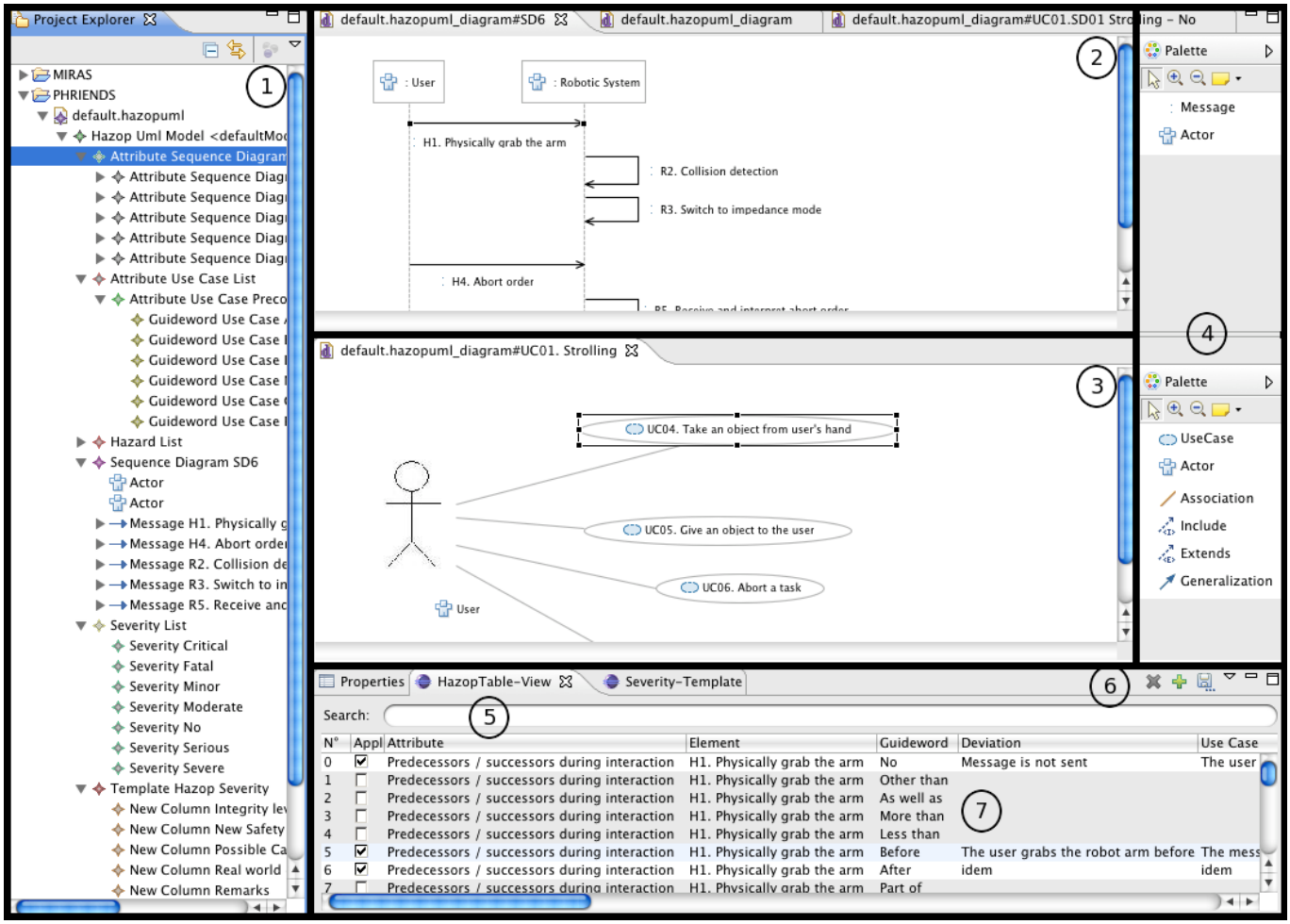
Following two case studies where we use our method to analyse safety of service robots (PHRIENDS [14] and MIRAS [4] projects) we developed a Computer-Aided Software Engineering tool (c.f. Figure 5) to support the method, with the following motivating features:

- Support for UML modelling (Use Case and Sequence Diagrams)
- Partial automatic generation of HAZOP tables for systematic analysis
- Management of the combinatorial aspects of the HAZOP
- Guaranteed consistency between UML model and risk analysis tables
- Support for building a safety argumentation for risk acceptability
- Profiles allowing project-dependent configuration of severity levels, HAZOP table columns and guideword lists

The tool is built as an Eclipse plugin (www.eclipse.org) using the Graphical Modelling Framework (GMF). A first prototype has been released implementing most of the steps of our method. Users can draw and document UML use case and sequence diagrams. Based on configurable guideword listings and on UML models, HAZOP tables are partially filled. The user can then define severity levels, and enter all data required for the analysis. Final documents (Hazard List, Recommendations and Required Integrity Levels) are then extracted from HAZOP or PHA Tables, and traceability links are displayed to link hazards to causes. The tool is easy to use because of its simplicity and integration to a common environment (Eclipse). The method and guideword list can be adapted thanks to HAZOP Table templates. Furthermore, the analysis can be exported in CSV to reuse the results outside the tool. However, rich formats like HTML or Excel are not yet available for exportation, which currently limits the integration of our tool with other software. A second prototype is under development integrating visual improvements and a more user-friendly production of documentation.

VI. Related Work

There have been several previous studies aimed at linking model-based development with risk analysis. For instance, in the CORAS project [15], [16], developed a framework, exploiting risk analysis and object oriented modelling concepts, for risk assessment of security critical systems. In our case we focus on safety and not on security, but the objectives of our study are quite similar to CORAS. Nevertheless, we do not have the same claims in terms of UML diagrams (we only focus on use case and sequence diagrams) and risk analysis techniques (we only focus on HAZOP and FTA). A major difference is that we strongly interconnect UML models and techniques such as HAZOP whereas that is not the case in CORAS. For instance, they use HAZOP without any real link with UML models (their HAZOP guidewords are not applicable on UML elements). Actually, they identify critical



N°	Appl	Attribute	Element	Guideword	Deviation	Use Case
0	☑	Predecessors / successors during interaction	H1. Physically grab the arm	No	Message is not sent	The user
1	☐	Predecessors / successors during interaction	H1. Physically grab the arm	Other than		
2	☐	Predecessors / successors during interaction	H1. Physically grab the arm	As well as		
3	☐	Predecessors / successors during interaction	H1. Physically grab the arm	More than		
4	☐	Predecessors / successors during interaction	H1. Physically grab the arm	Less than		
5	☑	Predecessors / successors during interaction	H1. Physically grab the arm	Before	The user grabs the robot arm before	The mess
6	☑	Predecessors / successors during interaction	H1. Physically grab the arm	After	idem	idem
7	☐	Predecessors / successors during interaction	H1. Physically grab the arm	Part of		

Fig. 5. Main view of the CASE Tool to support the UML-HAZOP method

assets and analyse for instance any deliberate/unintentional manipulation of this asset [17]. In safety, assets are not elements of the system but they are the system itself or system users. Hence, their approach is hardly applicable for safety analysis [16].

Our risk analysis approach is based on a re-interpretation of the guidewords for Hazard and Operability Studies (HAZOP) in the context of different UML models. The proposal in [18] followed by a more systematic study in [19], also considers a HAZOP guideword interpretation for the deviations of UML elements such as class, association, classifier role, message, etc. A similar approach was followed in [20] and [21], which also present a statistical analysis of the usability of this method. The guideword interpretation for the static UML diagrams in those studies aims to inspect the model to determine development faults and not to identify operational deviations. Nevertheless, for the UML dynamic diagrams (use case, sequence, activity, and statechart diagrams) many guideword interpretations can be used for exploring deviations during operational life. This is the case in studies presented in [22] and more formally in [23], which focus on use cases. The latter study led to a method that has been successfully used in [24] and [25]. This work on use cases also inspired a similar approach for security where new interpretations of guidewords have been proposed [26]. Even if this work is more oriented towards malicious behavior of actors, several interpretations can be applied in safety-critical systems with human-machine interactions. We combine and extend the results of those studies, but focus only on use case and sequence diagrams in order to explore deviations during operational life. We also give a particular attention to the integration of HAZOP-like human error analysis techniques as presented in [27]. Indeed, human factors methods [28] are a major issue in safety-critical systems but their analysis is often uncorrelated with preliminary system modeling activities. On the contrary, a key point of our approach is to consider human factors from the outset, by including them in model based risk analysis.

VII. Conclusion

To tackle safety of autonomous systems, appropriate analysis methods are needed especially when the system physically interacts with humans. Standard risk assessment methods are however limited to simple systems and usual model-based risk assessment methods do not enable to model dynamics. Thus we adapted the classical process with a new

model based approach for autonomous systems in physical contact with humans. We model the system using a subset of the well-known standard format UML. We apply PHA and our adaptation of HAZOP to identify hazards. A qualitative method is used to evaluate the risk on the preliminary design allowing the safety process to be integrated early in the development process. FTA is used to evaluate the risk on the other iterations of the safety process. A tool was developed to support the process.

We applied the process to the robot assistant developed in the MIRAS project. The first iteration of the safety process in that project confirmed the needs for high-level design analysis. Furthermore, starting the safety assessment process at the very first step of the design is helpful. In MIRAS, we obtained several results on the first iteration and the recommendations issued in that process enabled integration of safety constraints in the design of the second prototype. Assistant robotic is lacking of standard especially regarding the modeling of humans in the safety assessment process We are currently checking the design for the second prototype with our partners in the MIRAS project and plan to apply a second iteration of the safety assessment process, including a quantitative risk estimation using FTA.

ACKNOWLEDGEMENTS

This work was partially supported by the MIRAS Research Project, funded under the TecSan (Technologies for Healthcare) program of the French National Research Agency (French ANR).

REFERENCES

[1] OMG-UML2, "OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2," Object Management Group, formal/2007-11-02, 2007.

[2] IEC61882, "Hazard and operability studies (HAZOP studies) – Application guide," International Electrotechnical Commission, 2001.

[3] IEC61025Ed2.0:2006, "Fault tree analysis (fta)," International Electrotechnical Commission, 2006.

[4] MIRAS, "Multimodal Interactive Robot for Assistance in Strolling," Project supported by the French ANR (National Research Agency) under the TecSan (Healthcare Technologies) Program (ANR-08-TECS-009-04), http://www.miraswalker.com/index.php/en.

[5] ISO/FDIS14971:2006, "Medical devices - Application of risk management to medical devices," International Standard Organisation, 2006.

[6] ISO/IEC-Guide51, "Safety aspects - Guidelines for their inclusion in standards," International Organization for Standardization, 1999.

[7] ISO/IEC-Guide73, "Risk management - Vocabulary - Guidelines for use in standards," International Organization for Standardization, 2002.

[8] M. Rausand and A. Hyland, *System Reliability Theory: Models, Statistical Methods and Applications, 2nd Edition*. Wiley, 2004.

[9] D. Martin-Guillerez, J. Guiochet, D. Powell, and C. Zanon, "A UML-based method for risk analysis of human-robot interactions," in *2nd International Workshop on Software Engineering for Resilient Systems*. ACM, Apr. 2010.

[10] AIS98, "The abbreviated injury scale," Association for the Advancement of Automotive Medicine, Des Plaines, IL, USA, Tech. Rep., 1998.

[11] DefStan00-58, "HAZOP studies on systems containing programmable electronics," Defence Standard, Ministry of Defence, UK, 2000.

[12] IEC61508, "Functional safety of electrical/electronic/programmable electronic safety-related systems," International Electrotechnical Commission, 2000.

[13] M. Nokata and N. Tejima, "A safety strategy for rehabilitation robots," in *Advances in Rehabilitation Robotics*, Z. B. Stefanov and D., Eds. Springer-Verlag Berlin Heidelberg, 2004, pp. 177–185.

[14] PHRIENDS, "Physical Human-Robot Interaction: Dependability and Safety," Project supported by the European Commission under the 6th Framework Programme (STReP IST-045359), http://www.phriends.eu/.

[15] CORAS, "A platform for risk analysis of security critical systems," http://coras.sourceforge.net, http://www2.nr.no/coras/, 2010.

[16] R. F. Bjørn Axel Gran and A. P.-J. Thunem, "An approach for model-based risk assessment," in *23rd International Conference, SAFECOMP 2004, Potsdam, Germany*. Springer Berlin / Heidelberg, 2004, pp. 311–324.

[17] R. Winther, O.-a. Johnsen, and B. A. Gran, "Security assessments for safety critical systems using hazops," in *In: Proceedings of SAFECOMP 2001*. Springer, 2001, p. 1424.

[18] K. Lano, D. Clark, and K. Androutsopoulos, "Safety and security analysis of object-oriented models," in *SAFECOMP '02: Proceedings of the 21st International Conference on Computer Safety, Reliability and Security*. London, UK: Springer-Verlag, 2002, pp. 82–93.

[19] K. M. Hansen, L. Wells, and T. Maier, "Hazop analysis of uml-based software architecture descriptions of safety-critical systems," in *Proceedings of NWUML*, 2004.

[20] J. Gorski and A. Jarzebowicz, "Development and validation of a hazop-based inspection of uml models,," in *3rd World Congress for Software Quality, Munich, Germany*, 2005.

[21] A. Jarzebowicz and J. Górski, "Empirical evaluation of reading techniques for uml models inspection." *ITSSA*, vol. 1, no. 2, pp. 103–110, 2006.

[22] P. Johannessen, C. Grante, A. Alminger, U. Eklund, and J. Torin, "Hazard analysis in object oriented design of dependable systems," in *2001 International Conference on Dependable Systems and Networks, Göteborg, Sweden*, 2001, pp. 507–512.

[23] K. Allenby and T. Kelly, "Deriving safety requirements using scenarios," in *Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on*, 2001, pp. 228–235.

[24] A. Arlow, C. Duffy, and J. McDermid, "Safety specification of the active traffic management control system for english motorways," in *The First Institution of Engineering and Technology International Conference on System Safety*, 2006.

[25] I. Frantz, G. Andy, M. John, and I. Toyn, "Integrating safety and formal analyses using uml and pfs," *Reliability Engineering and System Safety*, vol. 92, no. 2, pp. 156–170, 2007.

[26] T. Srivatanakul, "Security analysis with deviational techniques," Ph.D. dissertation, University of York, 2005.

[27] J. Guiochet, G. Motet, C. Baron, and G. Boy, "Toward a human-centered uml for risk analysis - application to a medical robot," in *Proc. of the 18th IFIP World Computer Congress (WCC), Human Error, Safety and Systems Development (HESSD04)*, C. Johnson and P. Palanque, Eds. Kluwer Academic Publisher, 2004, pp. 177–191.

[28] N. Stanton, P. Salmon, G. Walker, C. Baber, and D. P. Jenkins, *Human Factors Methods: A Practical Guide for Engineering And Design*. Ashgate Publishing, 2006.