Experience with Model-Based User-Centered Risk Assessment for Service Robots

Jérémie Guiochet*†, Damien Martin-Guillerez*†, 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 now a major concern in many computer-based systems and more particularly for autonomous systems such as service robots in physical contact with humans. The traditional approach to analyze the safety of such systems is to use risk assessment methods based on models of system structure, or system behavior. Unfortunately, such models are hard to produce for autonomous systems. We propose an approach based on the standardized risk assessment process which is applied during the initial phases of the development process. We first use the common Unified Modeling Language (UML) and a preliminary application domain hazard analysis without considering any robotic device. Then, during the specification phase, a risk assessment of the robotic system is carried out. It consists in modeling tasks in UML, identifying hazardous situations (including human errors), and estimating associated risks. We base this analysis on an adaptation of the guideword-based collaborative method HAZOP (HAZard OPerability) applied to UML models. The process has been successfully applied to the development of an assistive robot providing assistance for standing up, sitting down and walking, and health-state monitoring. Results in terms of integrability, usability, validity and applicability of the method are really encouraging. Majors benefits are a good management of the level of abstraction (and thus combinatory explosion is controlled), an easy communication between different stakeholders using basic UML diagrams, and a structured safety documentation required for certification.**

***Keywords*-Risk assessment; safety; UML; HAZOP; autonomous systems; service robot**

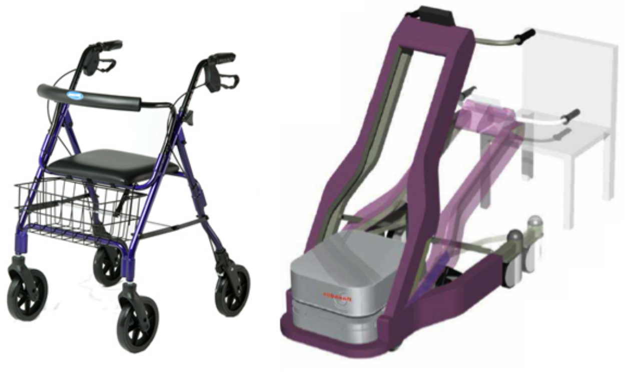
I. Introduction

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 humans. The traditional approach to analyze safety of such systems is to use risk analysis methods such as Fault Tree Analysis (FTA) or Failure Mode, Effects, and Criticality Analysis (FMECA). Those methods are usually based on representations of the system such as block diagrams for functional structure, and automata for dynamics. However, such models are inadequate for autonomous systems. First, the decisional software of such systems cannot easily be decomposed into functional blocks. Second, automata are not suitable for modeling the dynamics of goal-driven deliberative systems [1] that operate autonomously within an unstructured environment, that may include human beings. Moreover, the fact that environment is not structured means that the number of operating conditions is essentially infinite, which leads to a combinatory explosion using classic risk analysis methods.

We propose an approach to cope with these issues through the combination and adaptation of several well-known techniques. The proposed approach is driven by two important considerations: it is limited to the initial phases of the development process (Requirement Elicitation phase and Specification phase) and it is based on a standardized risk assessment process. In recent standards [2], it is composed of risk analysis (definition of intended use of the system, hazard identification, and risk severity and probability estimation) and risk evaluation (decision as to whether the risk is acceptable or not). The process is usually performed iteratively, in parallel with the development process. A recent trend is for risk assessment to be based on the same models as those used for system development. Thus, we adapt this classic risk assessment process to a concrete and usable model-based approach, which integrates some human factors activities.

During the *Requirement Elicitation* phase, we describe scenarios of use with the *de facto* standard Unified Modeling Language (UML [3]), and analyze application domain hazards without considering any robotic device. Then, during the *Specification* phase, a risk assessment of the robotic system is carried out. We consider that the earliest models used during system development, usually describe scenarios of use of the system. We claim that the analysis of deviations of such scenarios allows the identification of major hazards. Hence we restrain our approach to UML sequence and use case diagrams. Based on these models, we identify hazardous situations including human errors, and estimate associated risks. This analysis is based on an adaptation of the guideword-based collaborative method HAZOP (HAZard OPerability) [4] applied to UML models.

This process has been successfully applied to several robotic projects. We illustrate each step of the process using the MIRAS project [5]. which aims 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 where a classic wheeled walker (or “rollator”), such as in Figure 1(a), is not sufficient for patient autonomy. The robotic rollator is composed of a mobile base and a moving handlebar (Figure 1(b)).



(a) Rollator (b) First prototype of MIRAS RobuWalker

Figure 1. From classic walker to robotic assistant

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. We present related work in Section V. Section VI concludes the paper.

II. Method overview

In this section we give an overview of the process and its activities. The method will be detailed further in section III.

A. *Model-based risk assessment*

The concept of risk is now widely used in many domains, from financial mechanisms to embedded systems. Several standards provide generic definitions of risk and elements of the risk management process [2], [6]–[8]. The generally-accepted definition of risk in the safety domain is the combination of the likelihood of harm and its severity [7]. Tolerable risk is achieved by an iterative process of risk assessment (risk analysis and risk evaluation) and risk reduction (see left-hand part of the Figure 2). This process and its terminology are now quite stable in industrial standards and are widely accepted. In the safety domain, risk analysis aims to identify hazards and estimate the associated risk (i.e., estimate severity and probability of risk). Risk evaluation then consists of comparing the estimated risk against given risk criteria to determine the acceptability of the risk. Risk reduction is a process in which decisions are made and measures implemented with the aim of reducing risks to specified levels. As presented on Figure 2, the process is iterative because for each risk reduction decision it is necessary to estimate if risks are effectively reduced and check that new risks have not been introduced. A point that is usually not mentioned is that this process is also incremental, because it follows the development process, and for each new iteration of the development, another cycle of risk assessment is performed.

When adapting this process, the main challenges are to:

- describe the target of evaluation at the right level of abstraction;
- facilitate communication and interaction between different stakeholders involved in the risk assessment process (e.g., in our case, stakeholders are patients, medical staff, and 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.

In this context, we base our approach on the generic risk assessment process, in an iterative and incremental approach. The same cycle is repeated until the designed system achieves tolerable risk. It is strongly linked with the development process as it shares the same system description models. For this, we chose a subset of UML. This language was also used to 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 [3]. This abstract model 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 diagrams 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-conditions and post-conditions. Each sequence diagram represents one particular scenario of one use case.

B. *Activities during the requirement elicitation phase*

During this phase (cf. Figure 2), system analysts define the system requirements in collaboration with users, customers and other stakeholders. This is a wide field of research in system engineering, but we will only deal here with activities related to risk assessment and reduction. We first consider general usage scenarios of the application domain, using UML use case models to describe the various tasks involved. By "application domain", we refer here to the medical domain of rehabilitation (without any robot). For instance, in our case study, we gather knowledge on the use of classic walkers (wheeled or not), and we model it with a use case diagram. This work is then completed with a brainstorming meeting on application domain hazards, and investigations on the severity and probability of occurrence of potential harm. The analysis carried out during this phase provides an important point of reference for assessing the benefits in terms of safety of the new robotic system.

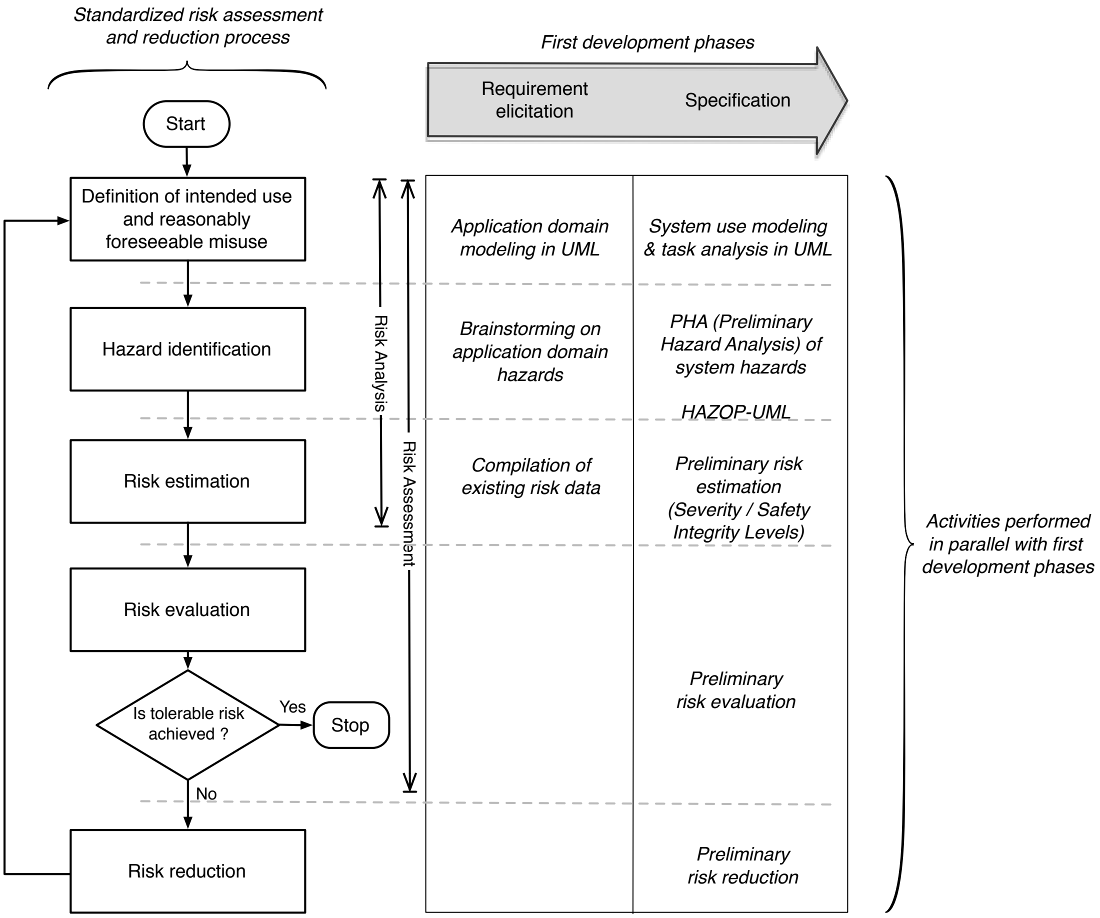

Standardized risk assessment and reduction process
First development phases
Requirement elicitation
Specification
Start
Definition of intended use and reasonably foreseeable misuse
Hazard identification
Risk estimation
Risk evaluation
Is tolerable risk achieved ?
Yes
Stop
No
Risk reduction
Risk Analysis
Risk Assessment
Application domain modeling in UML
System use modeling & task analysis in UML
Brainstorming on application domain hazards
PHA (Preliminary Hazard Analysis) of system hazards
HAZOP-UML
Compilation of existing risk data
Preliminary risk estimation (Severity / Safety Integrity Levels)
Preliminary risk evaluation
Preliminary risk reduction
Activities performed in parallel with first development phases

Figure 2. Risk assessment and risk reduction activities adapted from standardized process [6]

C. Activities during the specification phase

During the specification phase, we transform the application domain models into system models by integrating the robot. This activity, “System modeling & task analysis in UML”, is strongly linked with the human factors domain. Indeed, the activity dedicated to the analysis and the allocation of tasks between the robot, the patient and the operators, is a part of task analysis and task allocation, which are two main topics in human factors [9]. Task analysis aims to identify the details of specified tasks, including the knowledge, skills, attitudes, and personal characteristics required for successful task performance. During system analysis, this activity is linked to task allocation, which aims to determine the distribution of work between human actors and machines. For instance, it is particularly important to define non-ambiguous and consistent tasks for humans who are using the robot. We choose to represent robot tasks with the UML message concept, which is simple to manipulate and understand, even for non-experts in modeling (such as doctors). Task analysis and task allocation can be considered independently of the risk assessment process, but in case of safety-critical systems with important human-robot interaction (even physical interaction), they are strongly linked, so we include them in our process. The outputs of the first activity of the specification phase consist of system models, which are shared with the development process, and initial remarks and recommendations about inconsistency or preliminary ideas on safety. All these artefacts are then presented and validated during meetings with robotic and medical experts.

The second activity during the Specification phase is hazard identification, which is performed using Preliminary Hazard Analysis (PHA [10]) and an adaptation of HAZOP (HAZard OPerability [4]). PHA is a simple, inductive method of analysis whose objective is to identify the hazardous situations and events that can cause harm. It is most commonly carried out early in the development of a project when there is little information on design details. A list of hazards and generic hazardous situations is formulated during workshops and meetings by considering characteristics such as mechanical parts of the robot, hardware, software, robot’s human and physical environment.

It also produces a list of recommendations. 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 PHA in a classic way but we adapt the HAZOP method to apply it to UML use cases and sequence diagrams (see HAZOP-UML [11] [12]). This activity produces a large amount of data, from which we extract a list of hazards and recommendations. One important point is that we use both techniques (PHA and HAZOP) because they are complementary. Indeed, PHA is a top-down approach through which analysts directly identify hazardous situations and then potential sources, whereas HAZOP is more bottom-up, i.e., analysts identify misbehavior sources and then consequential hazardous situations.

Risk estimation is then carried out. By definition, it should consist in estimating severity and probability of occurrence of each potential harm. For that, the probability of hazardous situations needs to be estimated. Quantitative and even qualitative estimations of probability are unfortunately usually impossible at an early design stage, mainly due to the lack of data on human or software failure rates, or even on functional parts that have not yet been developed. It is nevertheless important to identify serious weaknesses of the system specification. Consequently, we base our risk evaluation at this stage on just the *severity* of the potential harm (a similar approach is considered in [13] when assessing the risks of rare large impact events in security-critical systems). A harm severity level is thus estimated for each hazard potentially inducing that harm (see Table I. This leads to two outputs. First, critical functions are identified and assigned a required integrity level, using the concept of Safety Integrity Levels (SIL)[1] in the IEC61508 standard [14]. Second, it produces a list of hazards and hazardous situations ranked according to their severity.

The next two activities (risk evaluation and risk reduction, cf. Figure 2) will not be presented in detail in this paper since at this stage in the development process they essentially depend on discussions and agreements between robotics experts and doctors. Indeed, risk evaluation, as it was mentioned before, is the decision of the acceptability of the risk. As we will see later in section III-E, our decision of acceptability of risk depends on the development version of the robot (development version, clinical evaluation version and final version). The risk reduction activity is the application of recommendations, and its integration in mechanical design for instance. This is carried out by the robotic and integration team of the project.

Table I
SEVERITY LIST USED IN THE MIRAS PROJECT, DERIVED FROM THE ABBREVIATED INJURY SCALE OF [15]

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

D. Risk assessment and reduction in other phases of the development process

We actually performed more iterations than those presented here (Requirement Elicitation and Specification), but we focus on the two first iterations in this paper partly due to space limitation but above all because it is during these iterations that most of the risk is identified and treated. During subsequent specification iterations, UML models are refined and the HAZOP analysis is updated. In later phases, which are outside the scope of this paper, we use Failure Mode, Effects, and Criticality Analysis (FMECA) and Fault Tree Analysis (FTA) to assess the detailed design.

III. RISK ASSESSMENT AND REDUCTION PROCESS ACTIVITIES

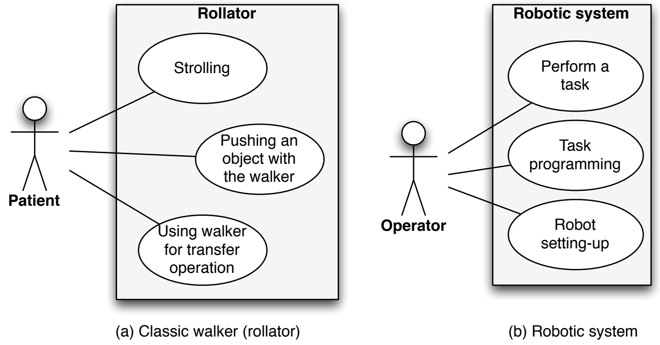
In this section, we detail the various activities of our risk assessment process. They are illustrated by their application in the MIRAS project. We present the activities performed during the first two phases of the development process: Requirement Elicitation and Specification.

A. Application domain modeling and associated hazard identification during Requirement Elicitation

We first model the application domain tasks without the robotic system (i.e., using a classic walker, or a "rollator" with UML use case diagrams (Figure 3(a)). During this step, application domain modeling increases the understanding of the domain and facilitates communication, particularly between engineers and doctors. In the next phase (Specification) this model will be reused to define tasks allocated to the robot and to the patient. For each use case, a textual description specifies more precisely the possible scenarios and their conditions of execution. Even if the use case *strolling* seems to be the most important for the design, other use cases which can later be critical for safety have to be analyzed. For example, while walking, the patient can *push an object* (for instance a door) with the walker. The future system should allow all use cases to be carried out safely, and particularly that one.

Hazard identification in this phase was carried out with brainstorming sessions or during meetings with doctors. We

[1] A SIL in the IEC61508 standard [14] corresponds to a level of confidence that can be accorded to a function or a component to operate correctly. It maps to a failure probability or rate with respect to hardware faults and to prescribed defences against software faults.

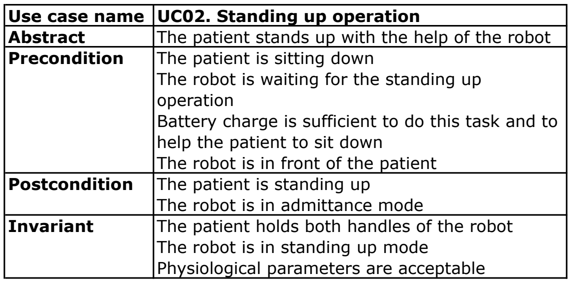


(a) Classic walker (rollator)

(b) Robotic system

Figure 3. Application domain use cases

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

found no database concerning accidents with classic walkers. The only studies that could be found focused on patient falling consequences, with hardly any statistical data. So we performed the analysis using generic hazard tables and discussions with doctors. We do not give the complete list here, but we used tables of generic hazards, such as in ISO14121 [16] (safety of machinery), to classify identified hazards according to four types : mechanical hazards (cutting edge, unstable structure, brake failure, flat tire, wheel blocked), hazards generated by neglected ergonomic design (unhealthy postures or excessive forces, patient feet collision with wheels, lateral movements impossible, fingers pinched by hand-brakes, handbrakes too hard to press), human errors (falling, release of the two handles, walking in hazardous places like stairs) and hazardous environmental conditions (holes, low obstacles, slippery floor).

Despite lack of documented knowledge of walker accidents (other than falling), the standard on (non-robotic) rollators ISO 11199 [17] gives very useful guidance for reducing probability or severity of harm by defining prescriptive requirements (e.g., for stability).

B. System use modeling & task analysis in UML

In this activity, we define the use cases of a robotic system (Figure 3(b)) and then merge with the use cases of a classic rollator (Figure 3(a)). This led us to modify the specifications of the previous use cases and identify new actors. Use cases can be described graphically, but the most usable tabular textual description will be used here. The use cases are completed by textual 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.

For instance, in the MIRAS project, we identified 13 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 moving to a waiting position* (UC06), *Positioning the robot by hand* (UC07), *Alarm handling* (UC08), *Patient profile programming* (UC09), *Patient profile learning* (UC10), and *Robot setting-up* (UC11), *Pushing an object with the robot* (UC12), *Use robot for transfer operation* (UC13). All use cases from domain application (cf. Figure 3(a)) can be found here, but new extensions of use result in new use cases (UC02 to UC06 for instance). A generic robotic use case like *Task programming* from Figure 3(b), has been split into two use case : *Patient profile programming* (UC09), *Patient profile learning* (UC10). Once all textual description has been done, pre-conditions, post-conditions and invariants are identified. By way of an example, Table II lists the conditions of use case UC02 (*Standing up operation*),

We chose to specify tasks and subtasks with the UML concept of messages in sequence diagrams. The sequence diagram of Figure 4 presents as an example the main scenario of the use case *Standing up*. Sequence diagrams are often used for high-level representations of a system. They are easily understood by 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 modeling process, preliminary safety remarks and recommendations can be issued.

C. PHA and HAZOP-UML

To identify hazards that can arise from the use of the robot, we used two complementary techniques: Preliminary Hazard Analysis (PHA) and HAZOP-UML (a method combining the HAZard OPerability technique with UML). To carry out our Preliminary Hazard Analysis, the various stakeholders of the project meet together for several workshops (two in the MIRAS project). 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 (see

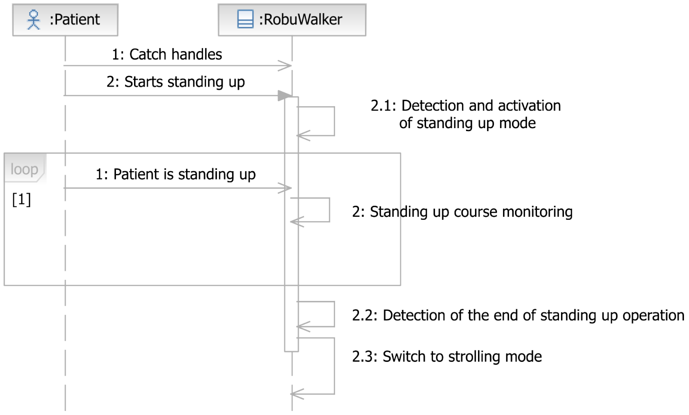

:Patient
:RobuWalker
1: Catch handles
2: Starts standing up
2.1: Detection and activation of standing up mode
loop
[1]
1: Patient is standing up
2: Standing up course monitoring
2.2: Detection of the end of standing up operation
2.3: Switch to strolling mode

Figure 4. Sequence diagram UC02.SD01 giving main scenario of UC02 "Standing up operation"

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

Project: MIRAS	**PHA Hazards**	**28/10/09** **Prepared by: JG** **Revised by : DMG**
Number	**Category**	**Hazard**
H13	Environment	Wet ground / Risk of grip loss
H27	Hardware	Electrical current drop / Risk of sudden reboot with wrong parameters.
H40	Mechanical	Robot runs over patient's foot

an extract in Table III).

We then apply the HAZOP-UML method, which adapts the HAZOP [4] method to analyze deviations of the UML use case and sequence diagrams. According to the Defence Standard 00-58 [18], 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 deviations. 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 [11]. As the choice of guidewords has also been done integrating human error models, the

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 HAZOP-UML 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

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

Project: MIRAS	**HAZOP Recommendations**	28/10/09 Prepared by:DMG Revised by: JG		
Number	**Description**	**Version scope**		
		Dev	**Eval**	**Final**
Rec1	The standing-up profile should be validated by a human operator			✓
Rec2	Worst-case electrical consumption must be evaluated beforehand (and display of the mean battery time left by the robot)			✓
Rec22	Send regularly a network heartbeat from the robot to the medical staff control panel. Launch alarm on time-out.			✓
Rec31	Safety margins should determined for maximum and minimum height of the robot (monitoring is required)		✓	✓

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

Project: MIRAS **HAZOP table number:** UC02 **Entity:** UC02					**Use case description**				**Date:** 04/08/09 **Prepared by:** Damien Martin-Guillerez **Revised by:** Jérémie Guiochet **Approved by:**		
					Use case name: Standing up operation						
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	Worst-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		

HAZOP tables also include the analysis of human errors. An important point is that human errors are analyzed in well-identified scenarios of use, showing also system response, which is not the case in many human error analysis methods (see [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. 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 resulting list of recommendations. This list is derived from the "new safety requirements" column of the HAZOP-UML tables. Actually, recommendations were issued as a result of each step of the risk assessment process. UML modeling, PHA and HAZOP-UML all gave rise to general recommendations.

D. Preliminary risk estimation

As previously mentioned, it is not possible to estimate probabilities for each hazard. Indeed, at this stage of the development, many choices are still not done. Also, it is difficult to estimate failure rates of humans, and software components under development. So, we estimate a severity level for each hazard identified in the HAZOP tables. This level is noted in the severity column of the HAZOP tables (see for instance Table IV, based on the severity level list of Table I). The "Type of Injury" column has been discussed with the stakeholders according to the MIRAS project objectives but this list can be generalized to all service robots.

During the HAZOP analysis we also assign preliminary SILs (Safety integrity Levels [14]) to safety related components. To do this, a SIL is mapped to each severity level (cf. Table I, SIL column). Components associated with each HAZOP deviation are then assigned a SIL (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).

E. Preliminary risk evaluation

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 for rehabilitation robots [19] there is no set of generally accepted risk acceptance principles. This is an important issue, which addresses political and ethical concerns. Nevertheless, criteria are needed for engineers to determine which risks have to be reduced. In the MIRAS study, after the preliminary risk estimation only considering hazard severity, each hazard and possible recommendations were proposed to the robotics experts. We also compared those hazards to the ones identified in the Requirement Elicitation phase to estimate the benefits of using such a system. On the basis of our risk analysis results, we proposed a classification of risk acceptance criteria 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 in the presence of medical staff, and *final* version for operational life without medical staff. The classification of risk acceptance criteria was used to determine the applicability of recommendations, as presented in the last three columns of Table VI (Dev, Eval and Final). This classification and the level assignment have been discussed with robotics and medical experts. Discussions have been driven by the tables with traceability notations (not presented here), linking recommendations of this table and covered hazardous situations. For example, recommendation 22 (Rec22 in Table VI), is not required in Dev and Eval versions, because the robot will be always used in the presence of medical or robotics experts.

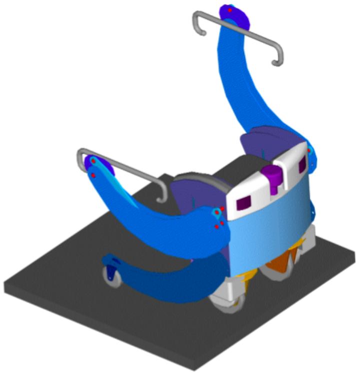


Figure 5. Mechanical design of the second prototype of the MIRAS robot

F. Preliminary risk reduction

Finally, selected risk reduction recommendations were applied to produce a new version of the robot. These recommendations can be classified in the following categories: modification of allowed use (e.g., restricted or forbidden environment), modification of the specification (e.g., behavior of the robot in case of release of the arms by the patient), and modification of the design (mechanical, hardware or software). Some recommendations reduce the severity of the corresponding hazard (e.g., a bumper added to the robot reduces the severity of a collision) while others 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 second version of the MIRAS robot (Figure 5) includes most mechanical and hardware recommendations issued after the analysis (it also includes several ergonomic changes). Some important recommendations are already included in the design of the second version of MIRAS robot, and other ones are still in discussion with the project's robotics experts[2]. This second version especially meets the requirements asked for the *evaluation* version of the robot (i.e., a robot that will be used in the laboratory and for clinical evaluation).

IV. Evaluation of the Method

We evaluate our approach from four different perspectives: **integrability** into the development process (sharing of the models, synchronization with development phases), **usability**, **validity** and **applicability**.

Integrability: in our approach, UML design models are shared with the development process. This helps to avoid inconsistencies. In MIRAS, we found it relatively easy to change models following design changes and to trace corresponding hazards. Another benefit is that our safety analysis can be carried out at the beginning of the development, but also during design refinement. However, HAZOP-UML cannot be applied to detailed UML models because of the combinatory explosion of the number of deviations.

Usability: the overhead of using this method in the overall process is reasonable low. For instance, in our Specification phase, apart from the modeling that was shared between stakeholders, the only critical path overhead that was induced were the meetings for hazard identification, risk estimation and evaluation: two workshops of two hours each for the Preliminary Hazard Analysis and two sessions of two hours to present the outcomes of the HAZOP-UML Analysis and the assessed risks. The time devoted to prepare guidelines for PHA workshops (a few hours), the time to carry out the HAZOP-UML analysis (2 weeks) and the time to format the results (a week) resulted in a one-month study. HAZOP-UML already proved to be usable even by hand but would be even more so with a tool to assist traceability and result formatting. We have developed a first prototype of such a tool [11].

Validity: Our approach partly relies on the HAZOP-UML method, which identifies a large set of operational hazards. Other hazards, especially environmental hazards are covered by PHA. Of course, all hazardous situations cannot be foreseen. Nevertheless, using these techniques, all classic hazards of robots (that can be found in standards) were identified and several new hazards were discovered. Other work in the context of MIRAS, based on the analysis of deviations of the behaviour of the robot modeled with state diagrams, has not yet identified any new hazards. This gives us confidence that the coverage of HAZOP-UML and PHA is sufficient to identify most hazards of service robots.

Applicability: the first iteration of our risk assessment process led 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. Our recommendations were taken into account in the design of the second prototype.

V. Related Work

There have been several previous studies aimed at linking model-based development with risk analysis. For instance, in the CORAS project [20], [21], a framework has been developed to exploit risk analysis and object-oriented modeling concepts, for risk assessment of security-critical systems. We focus on safety rather than 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). 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 to UML elements). Actually, they identify critical

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

assets and analyse deliberate/unintentional manipulation of these assets [22]. In safety, the entities of interest are system's users and environment rather than system assets. Hence, their approach is hardly applicable for safety analysis [21].

Our risk analysis approach is based on a re-interpretation of HAZOP guidewords in the context of certain UML models. The proposal in [23] followed by a more systematic study in [24], 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 [25] and [26], 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 [27] and more formally in [28], which focus on use cases. The latter study led to a method that has been successfully used in [29] and [30]. This work on use cases also inspired a similar approach for security where new interpretations of guidewords have been proposed [31]. 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 [32]. Indeed, human factors methods [9] 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.

VI. Conclusion

To tackle safety of autonomous systems, appropriate analysis methods are needed especially when the system physically interacts with humans. Even if standards and research papers converge on approaches based on risk assessment process, it is still not obvious how to apply such approaches, and to link them with the development process.

We thus adapted the classic risk assessment process to a concrete and usable model-based approach, which integrates some human factors activities. We presented the activities performed during the first phases of the development process. During the Requirement Elicitation phase we model the application domain tasks performed by the users without any robotic device, and establish a list of application domain hazards. During the specification phase, we model tasks and the system use, and perform risk assessment based on the UML models.

The artefacts produced by the method are a list of hazards, and for each hazard, a list of potential sources. This leads to a list of recommendations, and for each recommendation an applicability level depending on the robot version.

This method presents qualities of integrability, usability, validity and applicability. Majors benefits are: manageability of the level of model abstraction (and thus we control complexity and combinatory explosion), ease of communication between different stakeholders, and a structured safety documentation as required for certification. We are now improving the quality of this method by developing a tool (a first prototype has been developed and presented in [11]), that provides assistance for drawing UML diagrams and HAZOP tables, checking consistency between models, and producing documentation and reports.

An important drawback is that since our method is applied at the very first steps of the development process, it is impossible to collect data for estimation of probability of harm occurrence. This leads to an approach essentially driven by severity levels, considering that all deviations leading to a high severity should be treated. It is then difficult to justify that an acceptable level of risk has been reached. For this reason, we are now working on improving this method by allocating quantitative estimations of probabilities of hazard occurrence, building fault trees from the PHA and HAZOP-UML analyses. Another drawback is that the consequences of deviations are estimated by the analyst without any prescriptive model, and only with descriptive models (sequence diagrams for instance). It is thus strongly dependent on the analyst's expertise. We are now also working on an extension of HAZOP-UML to state-transition diagrams, in order to integrate deviations into the behavior models, so as to automatically identify their consequences.

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] L. Gunderson and J. Gunderson, "Chapter 10 - Deliberative System," in *Robots, Reasoning, and Reification*. Springer, 2009, pp. 121–137.

[2] ISO/IEC-31010, "Risk management - risk assessment techniques," International Standard Organisation, 2009.

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

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

[5] 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.

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

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

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

[9] 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.

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

[11] 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.

[12] D. Martin-Guillerez, J. Guiochet, and D. Powell, "Experience with a model-based safety analysis process for an autonomous service robot," in *IARP Workshop on Technical Challenges for Dependable Robots in Human Environments (DRHE 2010), Toulouse, France*, 2010, pp. 1–8.

[13] K. Hole and L.-H. Netland, "Toward risk assessment of large-impact and rare events," *IEEE Security and Privacy*, vol. 8, pp. 21–27, 2010.

[14] IEC61508, "Functional safety of electrical/electronic/programmable electronic safety-related systems," International Electrotechnical Commission, Ed. 2, April 2010.

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

[16] ISO14121-1, "Safety of machinery - risk assessment - part1 principles," International Standard Organisation, 2007.

[17] ISO11199-2:2005, "Walking aids manipulated by both arms – requirements and test methods – part 2: Rollators," International Standard Organisation, 2005.

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

[19] 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.

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

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

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

[27] 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.

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

[29] 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.

[30] 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.

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

[32] 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.