Human-robot interactions: model-based risk analysis and safety case construction

Quynh Anh Do Hoang, Jérémie Guiochet, David Powell, Mohamed Kaâniche

Human-robot interactions: model-based risk analysis and safety case construction

Quynh Anh DO HOANG*†, Jérémie GUIOCHET*†, David POWELL*†, and Mohamed KAANICHE*†
* CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse Cedex 4, France
† Université de Toulouse ; UPS, INSA, INP, ISAE ; UT1 ; UTM ; LAAS ;
F-31077 Toulouse Cedex 4, France
Email: *firstname.lastname@laas.fr*

Abstract

Recent advances in robotics technologies have opened multiple opportunities for the use of robots to support various activities of our daily life and to interact with humans in different ways. In such contexts, it is crucial to identify potential threats related to physical human-robot interactions and to assess the associated risks that might affect safety and dependability. Because of the complexity of human-robot interactions, rigorous and systematic approaches are needed to assist the developers in: i) the identification of significant threats and the implementation of efficient protection mechanisms to cope with these threats, and ii) the elaboration of a sound argumentation to justify the level of safety that can be achieved by the system. To fulfil these objectives, we believe that risk analysis should be carried out based on system models as soon as possible in the development process and hence provide elements to reason about system safety using a structured argumentation. The risk analysis method HAZOP-UML presented in this paper is a guided method to identify potential occurrences of harm, their causes and their severity. The results from risk analysis are then used as input for safety case construction in which we structure an argument about system safety. This process is illustrated by a case study on a robotized rollator.

I. Introduction

Since the invention of robots in the 1960's, they have been serving humans in many domains, especially in industrial applications. With current technology, robots are now able to assist medical staff in health-care activities such as robotised surgery, intelligent prosthetics and robotised patient-monitoring systems. For those systems, human presence in the working environment needs to be considered during risk assessment. In this paper, we present our experience in assessing risks and developing a safety case for an assistive robot for strolling, intended to be used by elderly persons.

Our case study is an innovative robot in the healthcare domain. The MIRAS (Multimodal Interactive Robot for Assistance in Strolling) Project started in 2009 and is scheduled to last 4 years. The challenge is to create a robot (Figure 1) that can replace current frame walkers or rollators to assist elderly persons with fragile balance, and provide them with increased autonomy while freeing medical staff for other activities. To help those people in their daily activities (strolling, standing up and sitting down), the robot is equipped with various sensors to determine the physical state of the user. Furthermore, while strolling, the robot should compensate for any loss of balance of the user.

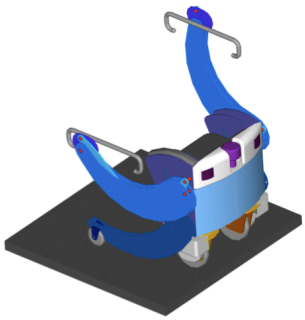

Fig. 1. The MIRAS multimodal assistive robot

Risk assessment methods such as Failure Mode, Effects, and Criticality Analysis (FMECA) or Fault Tree Analysis (FTA) have proved their efficiency for systems with well-known behavior. Unfortunately, that is not the case for the robot investigated in this paper which combines the knowledge from two major fields: robotics and healthcare. Not only it will be autonomous, but also it is capable of interacting with humans. Furthermore, it adapts itself to human movement, for example moving at walking speed or compensating for loss of balance while strolling. Those scenarios, if not handled correctly, can be very dangerous and cause elderly people to fall. As the robot interaction is subject to user behavior, current risk analysis methods are not easy to apply.

From risk assessment analysis one can derive many safety recommendations addressing separate aspects of the system design. However, while the safety of some functions can be assessed, it is difficult to have a comprehensive view about the overall safety achieved by the autonomous robot when it interacts with elderly people. Due to the complexity of human robot interactions and lack of data concerning rate of failures associated to human actions or some system failure modes related to design faults, traditional risk assessment techniques,

such as fault trees, are inconclusive.

Our approach to this problem is to use a model-based risk analysis method to describe human-physical systems interactions early in the design phase in order to identify possible threats and recommend appropriate protection mechanisms. Then, safety demonstration is carried out through the construction of a Safety Case providing a structured and solid argumentation about the robot safety focusing on interactions, using information derived from the models and additional evidence to support claims. This process is applied to the assistive robot investigated in our study and its results will be discussed in the following sections.

The next section gives an overview of the proposed risk assessment method which combines the modelling language UML with the risk analysis method HAZOP (HAZard OPerability), and uses the results to build a Safety Case. The application of this method in the context of the assistive robot case study is described in Sections III and IV, respectively. Finally, Section V discusses the main lessons learned and concludes the paper.

II. Model-based Safety analysis process

As is common in most safety risk analyses, we follow the process recommended in [5] and presented on the left side of Figure 2. In such standards (see also [9], [10]), the whole process is called risk management. In this paper we do not deal with all aspects of risk management but we focus on model-based risk analysis. For this we combine UML (Unified Modelling Language) [1] and the risk analysis technique HAZOP (Hazard Operability) [2] in order to perform hazard identification. Once the risk estimated (in terms of severity and probability of occurrence), it should be evaluated to decide whether the residual risk is acceptable or if additional risk reduction measures need to be implemented. It is actually rare to perform a complete and reliable estimation of risks. Indeed, in complex innovative systems, data about failure rates is often missing. It is difficult to estimate software or human failure rates, for instance. For this reason, we propose to use an argumentation process, supported by evidence, to justify that an acceptable level of risk is reached. The question "Is tolerable risk achieved?" in the risk management process is then supported by a structured argumentation, also called a Safety Case.

A standard from the UK Ministry of Defence [8] introduces Safety Cases and is widely adopted in many safety critical domains [11]. In fact, a Safety Case is a comprehensive document presenting complex arguments. It is defined as a structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment. In order to make the argument easier to understand and to avoid some problems such as ambiguity or purely textual description, a formalised notation supported by a tool named GSN (Goal Structuring Notation) [13] was also developed at University of York [12].

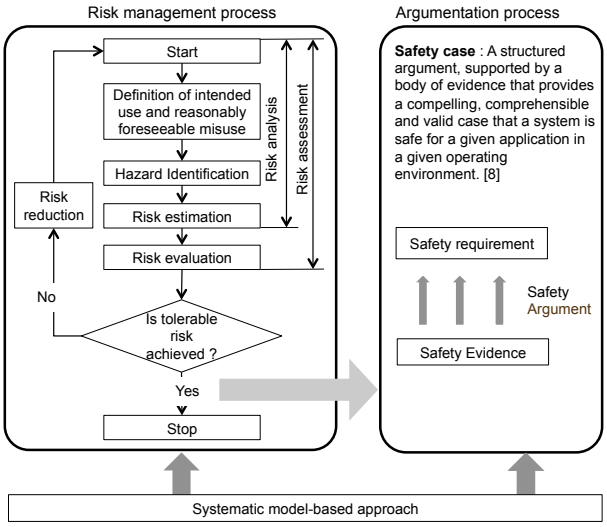

Fig. 2. Our model-based Safety process and link to Safety Case

III. Risk analysis with HAZOP-UML

We start the risk analysis process by modelling the system with UML focusing on 3 types of diagrams: Use Case, Sequence and Statechart diagrams. From our experience, those diagrams are the most suitable to describe the human-system interactions at the first step of the development process. The selection of UML as a modelling language to support these analyses is justified by its wide use in the community and in industrial processes.

In our case study, we identified 11 use cases. Typical examples are: assist the user while strolling by moving at the same speed as he/she walks (UC01), assist the user in standing up by lifting handgrips (UC02), raise an alarm when a physiological problem is detected during its use (UC08), etc. Interactions between the user and the robot in different scenarios are also described by Sequence Diagrams. Figure 3 shows an extract of the Use Case diagram of the robot and Figure 4 the nominal scenario for the Standing Up operation.

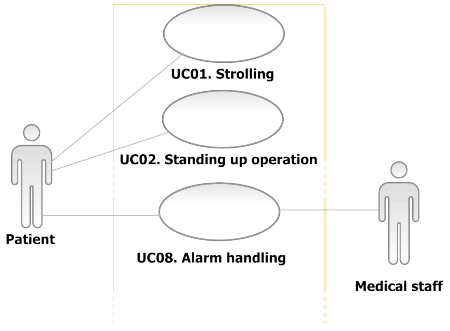

Fig. 3. An extract of the Use Case diagram

Besides such descriptive diagrams, we also used a statechart diagram to describe the behavior of the robot. This will be

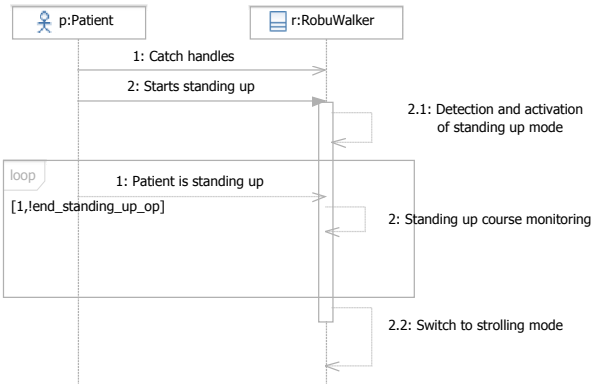

Fig. 4. Nominal scenario for Standing Up operation

helpful to identify hazards resulting from unwanted behavior of the robot. Figure 5 presents a simplified view of such a diagram.

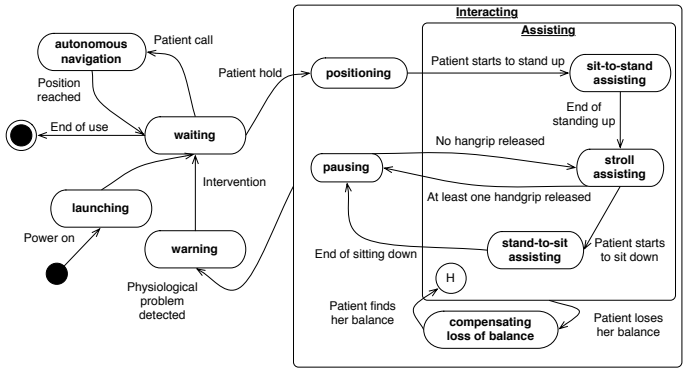

Fig. 5. Statechart diagram of the robot (nominal scenario)

Entity = Statechart		
Attribute	**Guideword**	**Interpretation**
Destination State	Other than	The destination state is not the one expected
Transition	Never	The transition is never triggered because the event is never true
	No/None	The transition is not triggered when it has to
Event	No/None	The event is false but the transition is triggered
	Other than	The event is true but the transition is not triggered
	Other than	The event is false but another event is true and triggers the transition
Condition	No/None	The condition is not evaluated but the transition is triggered
	Other than	The condition is true but the transition is not triggered
	Other than	The condition is false but the transition is triggered
	As well as	The condition is true and the transition is triggered but another unwanted condition is also true
	Part of	The condition is partly evaluated but the transition is triggered
	Early	The condition is evaluated and trigger the transition earlier than required
	Late	The condition is evaluated and trigger the transition later than required
Action	No/None	The transition is not triggered and the action does not happen
	Other than	The transition is triggered but another action happens instead
	As well as	The transition is triggered and the action as well as another unwanted action happen
	Part of	The transition is triggered but only part of action happens correctly
	Early	The transition is triggered but the action happens earlier than required for correct synchronization with the environment
	Late	The transition is triggered but the action happens later than required for correct synchronization with the environment
	More	The transition is triggered but the action result value is higher than expected
	Less	The transition is triggered but the action result value is lower than expected

Fig. 6. Adapted guide-words for Statechart diagram

Based on these diagrams, we proceed to the risk analysis combining HAZOP and UML. HAZOP is a collaborative method based on guide-words. For instance, guide-words such as MORE, LESS, NONE, OTHER THAN are combined with parameters or elements of the system to identify possible behavioral deviations and analyze their impact from the safety perspective. For example, the guide-word MORE combined with the system parameter SPEED makes us aware of the situation "the speed is too high for the system to work properly". If it is meaningful, then it is considered as a deviation that models a hazardous situation that can lead to an accident. By analyzing each deviation, possible causes are identified, and safety measures are suggested to reduce the severity of the potential harm and/or to prevent the accident from happening.

Using UML models, there are no system elements or parameters in the diagrams, so we adapted guide-words to UML elements. For instance, for statechart diagrams, guide-words are associated to states, transitions, events, conditions and actions (Figure 6). Adapted guide-words for Use Case and Sequence diagrams can be found in [3], [4].

Then, for each guide-word $\times$ element combination (Figure 7), a deviation description is given, and effects on the system and consequences for users are identified (real world effect). The severity is rated for each deviation based on a ranking predefined with medical experts (Catastrophic, Critical, Serious, Minor, Negligible). Possible causes and safety requirements are then investigated.

When we applied this method to the UML models describing the assistive robot case study, 4670 deviations were studied, leading to a list of 16 main hazards (Figure 8) and 57 safety recommendations (Figure 9). The investigation of such a high number of potential deviations was made possible thanks to the automation of the process that is inherent to the UML-HAZOP method that consists in systematically exploring all the UML model elements and systematically analyzing possible deviations. A tool supporting this method was also developed.

An example of safety recommendation was to inlcude in the final version a heartbeat mechanism to regularly check the state of the robot and send an alarm to the medical staff in case of robot failure (Rec22 in Figure 9).

Project: MIRAS **HAZOP table number:** UC02 **Entity:** UC02					**UC02. Standing Up operation**			**Date:** **Prepared by:** **Revised by:** **Approved by:**	04/08/09 Damien Martin-Guillerez Jérémie Guiochet	
Line Number	**Element**	**Guide-word**	**Deviation**	**Use Case Effect**	**Real World Effect**	**Severity**	**Possible Causes**	**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 interruptss its movement (standing up or walking)	Loss of balance or fall of the patient	Serious	HW/SW Failure Specification error	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	Battery charge is high enough but the robot detects otherwise	The robot refuses to start stand up operation	Patient is confused	Negligible	HW/SW Failure Specification error	None		

Fig. 7. An example of HAZOP table based on UML models

HN	Description	Severity
HN1	Incorrect position of the patient during robot use	Minor
HN2	Fall of patient during robot use	Catastrophic
HN3	Robot shutdown during its use	Critical
HN4	Fall of patient without alarm or with a late alarm	Catastrophic
HN5	Physiological problem of the patient without alarm or with a late alarm	Critical
HN6	Fall of the patient caused by the robot	Catastrophic
HN7	Failure to switch to safe mode when a problem is detected. The robots keep moving	Serious
HN8	Robot parts catching patient or clothes	Critical
HN9	Collision between the robot (or robot part) and the patient	Serious
HN10	Collision between the robor and a person other than the patient	Critical
HN11	Disturbance of medical staff during an intervention	Minor
HN12	Patient loses her balance due to the robot	Serious
HN13	Patient fatigue	Serious
HN14	Injuries of the patient due to robot sudden movements while carrying the patient	Catastrophic
HN15	Fall of the patient from the robot seat	Catastrophic
HN16	Frequent false positive alarms	Serious

Fig. 8. Hazards list

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)		✓	✓

Fig. 9. An extract of recommendations to improve robot safety

IV. Safety Case Construction

In the generic example presented in Figure 10, the main goal (G1) is split into 2 subgoals, one for software components (G2) and one for hardware components (G3). The justification of safety related to subgoal G2 is based on the SIL concept and process level qualitative criteria such as those defined in the IEC 61508 standard, because for software and design faults, it is usually difficult to demonstrate quantitatively that an acceptable failure rate has been reached, as would be done in the case of hardware components. In our case study, one hazardous situation often involves simultaneously software, hardware, mechanical design and user. Hence, the argumentation about safety should take into account all these components in a comprehensive way, as illustrated in the following.

For the MIRAS project, we choose to compare the robot to a classic rollator or frame walker. If the robot shows higher performance from the safety perspective compared to a traditional robot, the project will be successful. Hence, we have set as top-goal G1 the claim that: “The MIRAS robot is at least as safe as a classical rollator” (see Figure 11). This goal is broken-down into sub-goals through two strategies: we argue safety claims with respect to, on one hand, risks induced by the robot technology and, on the other hand, risks that are equally relevant to a classic rollator. All the risks identified

earlier by the HAZOP-UML risk assessment method find their place in each strategy.

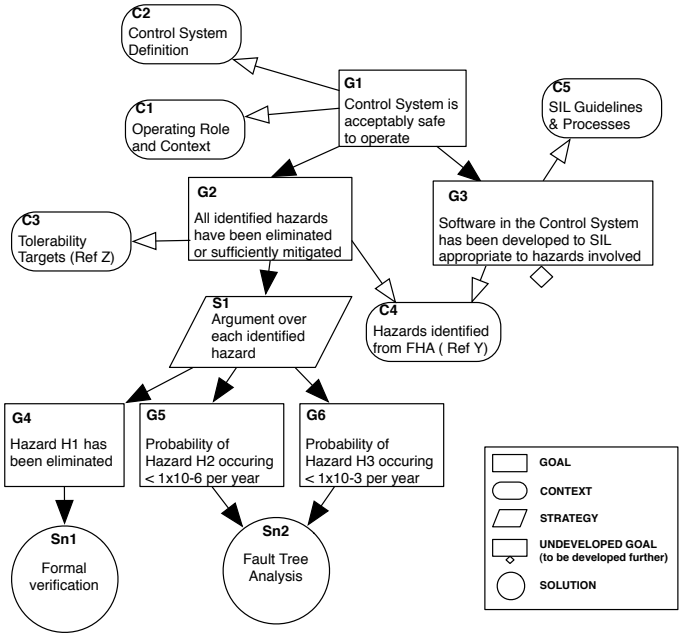

Fig. 10. GSN illustred by a generic safety argument [13]

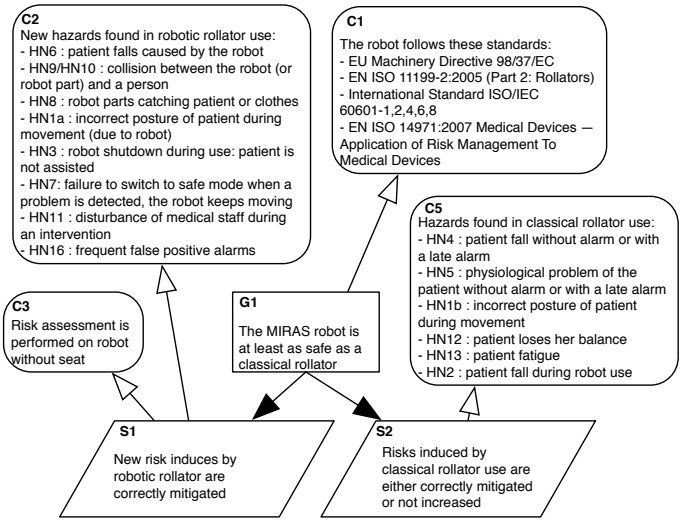

Fig. 11. The High Level Goal Structure of MIRAS

Among the 16 main hazards identified (Figure 8), the most dangerous usually result from a software failure. But it is difficult to estimate the failure rate of software. IEC 62304 [7] in medical field suggests a severity ranking system based on patient injuries (Figure 12). As the standard prescribes neither a process model nor particular software engineering methods to accomplish the normative requirements, we defined a mapping between this severity ranking and the Safety Integrity Level (SIL) of IEC 61508 (Figure 13). Four levels are defined in the standard but in the context of the assistive robot, it seems reasonable to consider that the highest SIL should not exceed SIL3, the highest direct harm being a patient fall (and not death).

Class	Description
A	No injury or damage to health is possible
B	Non-serious injury is possible
C	Death or serious injury is possible

Fig. 12. Severity ranking from IEC 62304

Severity	Class by IEC 62304	SIL by IEC 61508
Catastrophic	C	3
Critical	B	2
Serious	B	2
Minor	A	1
Negligible	A	0

Fig. 13. Cconnexion between IEC 61508 and IEC 62304

As we use the risk analysis results as an input to demonstrate robot safety, identified risks and approved safety recommendations help identify sub-goals as well. As for the basic solution (evidence), we provide various pieces of evidence according to the sub-goal to be solved: test results, estimation of fault detection coverage and compensation efficiency, proof of correct implementation of code, failure rate of physical components, compliance with standards, etc. In our case, we get a list of 44 pieces of evidence to be collected (Figure 14).

Code	Solution description
Sn9	Test of maximum speed in presence of fault
Sn29	Function which raises an alarm after physiological problem is class C (IEC 62304)
Sn19	Test of the fall detection system to calculate false positive/false negative rate
Sn32	Robot mechanical mechanisms are compliant with standard ISO 11999-2 on rollators
Sn34	A Heartbeat system is implemented to monitor the robot
Sn35	Emergency break is available, efficient and easy to activate

Fig. 14. An extract of solutions to be collected

During the GSN analysis, many choices rely on the expertise of the analyst. For instance, Figure 15 presents the argumentation for the mitigation of risk HN12 (patient loses her balance). To address this risk, the designers implemented a function able to accelerate or decelerate to compensate the patient unbalance. Such a function should reduce the consequences in this situation, the worst being a fall. Of course, we need to argue that such a system effectively reduces the risk, and does not add new risks (such as bad compensation). The acceptability of this function can then be argued while achieving three goals: i) reduce the risk of failures due to design faults (G9.1) using rigorous development methods as suggested for instance in the IEC 61508 standard; ii) show that the failure rate of the compensation system is acceptable (G9.2) using, e.g., fault tree analysis and iii) demonstrate the coverage factor of the compensation system (effective

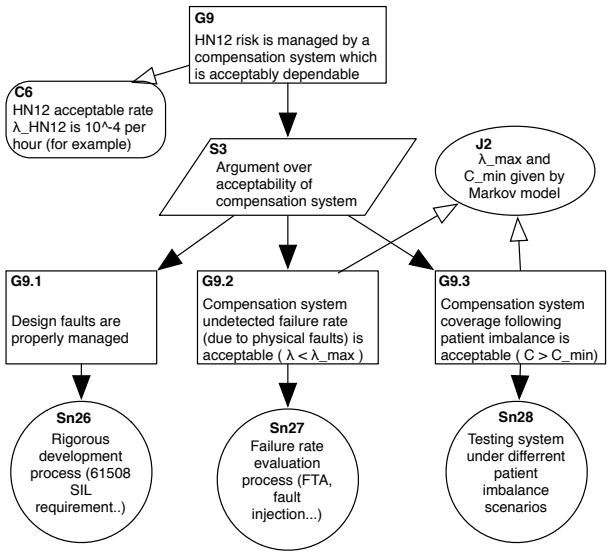

Fig. 15. Safety goal to be achieved by robot: compensate patient's loss of balance in time and in an efficient way

detection) is acceptable (G9.3) using testing under different patient imbalance scenarios. To demonstrate G9, for which the acceptability criterion is established through discussions with medical experts, we need to demonstrate that subgoals are satisfied. This can be done based on a Markov model presented in Figure 16. In this model, we consider four different states that result from the combined analysis of the patient and the system states: "patient OK - system OK", "patient not OK - system OK", "patient OK - system not OK", "patient fall", and the following parameters λ - undetected system failure rate, α - loss of balance rate, μ - compensation rate and C - compensation system coverage. λ is given by experimental or analytical studies, α by expertise, μ and C by dynamic test results.

Using traditional techniques for processing Markov models, we can estimate the rate characterizing the occurrence of risk HN12, which corresponds to the rate for the system to reach the "Patient fall" state starting from the initial state " patient OK - system OK". This rate is given by the analytical formula $\lambda_{HN12} = \lambda + (1 - C) \times \alpha$. For a given acceptable λ_{HN12} we can then deduce the acceptable values of C and α as presented in Figure 17. For instance, for $C = 0.9985$, $\lambda = 2 \times 10^{-5}$/hour, we obtain λ_{HN12} inside the feasible region, its value is less than 10^{-4}/hour. To demonstrate that such rates are obtained, we should then perform operational tests (Sn28) or use fault trees (Sn27).

V. Conclusion

The paper has focused on the benefits of associating model-based safety analysis with GSN. We have illustrated the proposed risk assessment method on a complex innovative system for which no safety standards have yet been developed: a robotised rollator that is able to assist elderly people in strolling, standing up and sitting down. Guide-words from

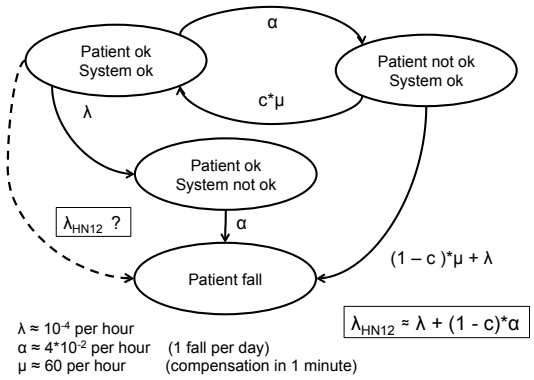

Fig. 16. Threshold estimation by Markov model

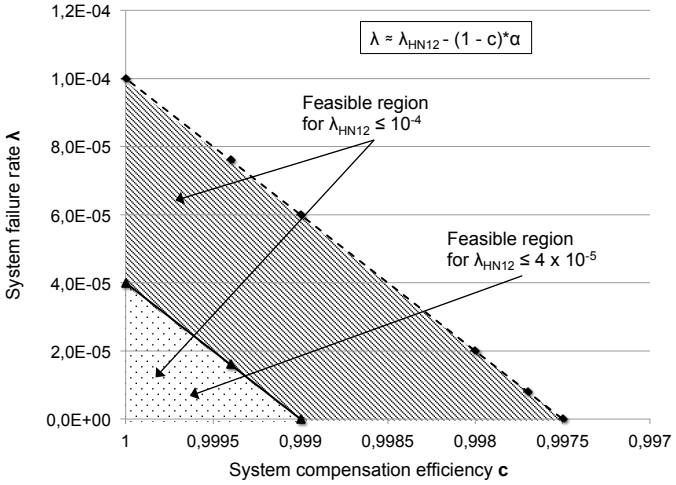

Fig. 17. Feasible region of values

HAZOP were adapted to elements of UML models to help identify the risks through deviation analysis. We then developed a GSN-based graphical safety case in which items of evidence are identified to support claims made during the argumentation. Indeed, safety case construction was beneficial to the risk evaluation process in that it helped us to identify needed evidence in support of safety claims. However, it is still up to experts to choose suitable methods for collecting some kinds of evidence.

The application of the method presented in the paper allowed several improvements to be made to the design of the assistive robot. Safety recommendations derived from the risk assessment approach have been integrated into the system and led to the development of a new version. Finally, it should be noted that this study required close interaction between the medical staff, the final users and the system design team. The model-based risk assessment approach was a useful means to facilitate such interactions.

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) and SAPHARI, Project funded under the 7th Framework Programme of the European Community.

REFERENCES

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

[2] IEC 61882, Hazard and operability studies (HAZOP studies) – Application guide, *International Electrotechnical Commission*, 2001

[3] D. Martin-Guillerez, J. Guiochet, D. Powell, C. Zanon."A UML-based Method for Risk Analysis of Human-Robot Interactions" In *2nd Int. Workshop on Software Engineering for Resilient Systems*, pp. 32-41, London, UK, 2010.

[4] J. Guiochet, D. Martin-Guillerez, D. Powell, "Experience with a model-based user-centered risk assessment for service robot", *International High Assurance Systems Engineering Symposium (HASE 2010)*, San Jose (USA), 1-4 Novembre 2010, 10p.

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

[6] IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related systems, *International Electrotechnical Commission*, 2000

[7] IEC 62304:2006 Medical device software - Software life cycle processes, *International Organization for Standardization*, 2006

[8] UK Ministry of Defence (2007) Defence Standard 00-56 Issue 4: Safety Management Requirements for Defence Systems

[9] IEC 31000, Principles and Guidelines on Implementation, *International Organization for Standardization*, 2009

[10] IEC 31010, Risk Management - Risk Assessment Techniques, *International Organization for Standardization*, 2009

[11] P. Bishop, R. Bloomfield, "A Methodology for Safety Case Development", *Safety-Critical Systems Symposium*, Birmingham, UK, Feb 1998

[12] T. P. Kelly. "Arguing Safety – A Systematic Approach to Managing Safety Cases." Ph.D. Dissertation, University of York, UK, 1998.

[13] GSN Standard Draft Version, *http://www-users.cs.york.ac.uk/∼katrina/GSN_ site/20100517_ GSNStandard_ v1.0.pdf*, visited on December 1st 2011