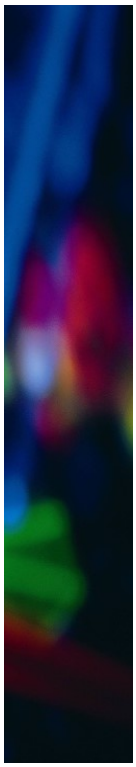
Méthodologie pour la conception d'espaces de travail sûrs, avec la collaboration de robot humain

Jose Saenz



2019-ENAM-0070

École doctorale n° 432 : Sciences des Métiers de l'ingénieur

Doctorat

T H È S E

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure d'Arts et Métiers

Spécialité " Automatique "

présentée et soutenue publiquement par

José SAENZ

le 17. Décembre 2019

Méthodologie pour la conception d'espaces de travail sûrs, avec la collaboration de robot humain

Directeur de thèse : **Olivier GIBARU**
Co-encadrement de la thèse : **Pedro NETO**

Jury

M. Marcello PELLICCIARI, Professeur, Università di Modena e Reggio Emilia	Président
M. Norbert ELKMANN, Professeur, Otto-von-Guericke Universität Magdeburg	Rapporteur
M. Nazih MECHBAL, Professeur, École Nationale Supérieure d'Arts et Métiers	Examinateur
M. Olivier GIBARU, Professeur, LISPEN, École Nationale Supérieure d'Arts et Métiers	Examinateur
M. Pedro NETO, CORLUC, Universidade de Coimbra	Examinateur
Mme. Julia ARLINGHAUS, Professeur, Otto-von-Guericke Universität Magdeburg	Invité

T H È S E

Arts et Métiers - Campus de Lille
Laboratoire d'Ingenierie des Systèmes Physiques et Numériques

Methodology for Design of Safe Workspaces featuring Human Robot Collaboration

by

José F. Saenz

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctorate

École Nationale Supérieure d'Arts et Métiers

in

Mechanical Engineering

Committee in charge:

Professor Olivier Gibaru, Chair
Professor Pedro Neto, Co-chair

2019

Methodology for Design of Safe Workspaces featuring Human Robot Collaboration

By
José F. Saenz

Abstract

Methodology for Design of Safe Workspaces featuring Human Robot Collaboration

by

José F. Saenz

Doctorate in Mechanical Engineering

École Nationale Supérieure d'Arts et Métiers

Collaborative robotics have a large potential for use in industrial applications. Nevertheless, this potential is currently unrealized and one of the reasons is the challenges in planning and designing while considering the safety requirements of these new types of applications. This thesis focuses on the development of a method for considering the safety aspects of industrial applications featuring human-robot collaboration (HRC applications) during the design phase based on systems engineering, ontologies, an Industry 4.0 concepts. Such HRC applications are characterized by their complexity due to the interdependencies between different system parameters (choice of hardware, parameterization of the system such as sensor range or robot speed) and can be considered system of systems. A review of the current practice for planning and designing HRC applications with a focus on the safety-related aspects was carried out to identify where the current challenges are from the designer's point of view. An additional review of the state of the art of using systems engineering principles and methods for robotics and safety revealed a large number of past efforts focusing on software engineering, hazard and risk analysis, and ontological development (which underlies all that work). Nevertheless, there were no efforts up to now that focused on the issues that the designer faces with regard to the choice and parameterization of components for HRC applications featuring speed and separation monitoring.

Requirements on a new approach to the consideration of safety based upon the needs of the designer and safety expert were formulated and guided the processes of architecture specification and component modeling that are the focus of this thesis. In particular, reusable models of sensors and robots are necessary in order to have digital information about the application in one place. This allows for easier use and execution of "what-if" analyses that are a part of the design process. An architecture that utilizes existing CAD/simulation tools that designers currently use was specified. The approach was implemented as a computer-aided safety tool (CAS Tool) and the design results were compared with traditional methods with two exemplary use-cases. The use-cases are adapted from real use-cases from the automotive industry.

With the CAS Tool it was possible to quickly make changes to the design, particularly the components such as the robot and safety sensors. The placement of the components and the process itself (i.e. the robot speed) can be easily adapted to achieve specific design goals such as minimum cycle time or minimal footprint of the application. The final outcome was validated against the original set of design requirements, which have been fulfilled. The safety zones calculated with the CAS Tool are up to 66% smaller than with traditional, worst-case methods.

The overall methodology of modeling safety-related attributes of the complete system can have wide-reaching consequences for future robotics applications that are flexible and feature online changes to the program during run-time.

Keywords: human-robot interaction, collaborative robots, safety, systems engineering, industry 4.0

Résumé

Méthodologie pour la conception d'espaces de travail sûrs, avec la collaboration de robot humain

pour
José F. Saenz
Doctorat Mécanique-Matériaux
École Nationale Supérieure d'Arts et Métiers

La robotique collaborative présente un grand potentiel d'utilisation dans les applications industrielles. Ce potentiel est cependant largement inexploité actuellement. L'une des raisons est liée aux difficultés de planification et de conception prenant en compte l'ensemble des des contraintes de sécurité pour ces nouvelles applications. Cette thèse propose le développement scientifique d'une méthode permettant de prendre en compte les aspects de sécurité pour les applications industrielles faisant intervenir la collaboration homme-robot (applications HRC). Cette méthodologie propose de prendre en compte ces enjeux de sécurité dès la phase de conception, sur la base des concepts d'ingénierie des systèmes, d'ontologies et d'Industrie 4.0. Ces applications HRC se caractérisent par leur complexité en raison des interdépendances entre différents paramètres du système (choix du matériel, paramétrage du système, telles que la plage d'utilisation des capteurs ou la vitesse du robot). Elles peuvent être considérées comme des systèmes de systèmes. Une analyse de la pratique actuelle en matière de planification et de conception d'applications HRC tout en mettant l'accent sur les aspects liés à la sécurité, a été effectuée afin de déterminer où se situent les défis scientifiques de conception. L'étude de l'état de l'art en matière d'utilisation des principes et méthodes d'ingénierie des systèmes pour la robotique et de la sécurité a révélé de nombreux efforts passés axés sur l'ingénierie logicielle, l'analyse des dangers et des risques et le développement ontologique. Cependant, aucun effort n'a été consacré au traitement des problèmes concernant le paramétrage des composants pour les applications HRC intégrant une surveillance de la vitesse et de la distance de sécurité.

Les exigences relatives à une nouvelle approche de la prise en compte de la sécurité basée sur les besoins du concepteur et de l'expert en sécurité ont été formulées et ont guidé les processus de spécification d'architecture et de modélisation des composants qui sont au centre de cette thèse. En particulier, des modèles réutilisables de capteurs et de robots sont proposés. Cela facilite l'utilisation et l'exécution d'analyses d'hypothèses faisant partie du processus de conception. Une architecture utilisant les outils de CAD / simulation existants utilisés par les concepteurs a été spécifiée. L'approche a été mise en œuvre sous la forme d'un outil de sécurité assisté par ordinateur nommé CAS Tool. Les résultats de la conception ont été comparés aux méthodes traditionnelles sur deux exemples d'utilisation. Ces cas sont adaptés de cas réels de l'industrie automobile.

Avec notre outil CAS, il est possible de modifier rapidement la conception, en particulier les composants tels que le robot et les capteurs de sécurité. Le placement des composants et le processus lui-même (c'est-à-dire la vitesse du robot) peuvent être facilement adaptés pour atteindre des objectifs de conception spécifiques, tels qu'un temps de cycle minimal ou un encombrement minimum de l'application. Le résultat final a été validé par rapport aux exigences de conception initiales. Les zones de sécurité calculées à l'aide de l'outil CAS sont réduites jusqu'à 66% par rapport aux méthodes traditionnelles les plus défavorables.

La méthodologie globale de modélisation des attributs liés à la sécurité du système complet peut avoir de vastes conséquences pour les applications de robotique futures. L'outil proposé dans cette thèse, résultat de notre analyse scientifique, propose des modifications "en ligne" du programme. Ceci permet une plus grande flexibilité.

Mots clés: Interaction homme-robot, robotique collaborative, sécurité, ingénierie des systèmes, industrie 4.0

Table of Contents

Disclaimer - the software implementation of the CAS Tool is outside the scope of this thesis and was the work of the Robotic Systems Business Unit of the Fraunhofer IFF. This work makes no claims about the software implementation methods or software development tools used for that implementation.

List of Figures

List of Tables

Acknowledgements

This work began in 2016 with the idea that there has to be a better way to design collaborative robotics applications than the existing methods. The H2020 funded project “ColRobot” was underway and the planning of the safety-related aspects was slow, iterative, and time-consuming. It was within this project that I first collaborated with Prof. Gibaru and Prof. Neto.

It has since been a long journey from this first idea to the conclusion of this thesis. This work has been the result of close collaborations with my colleagues at the Fraunhofer IFF in Magdeburg, Germany as well as with with Prof. Gibaru and Prof. Neto.

I would like to extend my heartfelt thanks to the IFF team, who aren’t afraid to argue with me and challenge assumptions. I would also like to extend special thanks to Prof. Norbert Elkmann for his support and for creating the working conditions so that I could pursue a doctoral thesis alongside normal work.

I would like to give thanks to Prof. Gibaru and Prof. Neto for their time, their reviews of my work, their patience and motivating words. You both offered the right mix of support and hard deadlines and have pushed me harder than I thought possible. I look forward to future collaboration with you both.

This work is dedicated to my wife and three children who also supported me in this endeavor, gave me time alone to work, and helped me regenerate in our time together. I love you and I promise I won’t try this again.

José Saenz

Magdeburg, October 31, 2019

Acronyms

CAD	computer-aided design software is used to design and create mechanical structures
CE mark	is a certification mark that indicates conformity with, safety, and environmental protection standards for products sold within the European Economic Area (EEA)
ECAD	electronic computer-aided design software is used to design and create electronic structures
EU	European Union
FMEA	Failure Mode and Effect Analysis
FMECA	Failure Modes, Effects and Criticality Analysis
HAZOP	HAZard Operability
IFR	International Federation of Robotics
PLC	programmable logic controller
POV	program override (%) – it is the velocity of the robot motion as a percentage of the programmed speed vs the maximum possible speed.
ROI	return of investment
SE	systems engineering
SysML	systems modeling language is a general-purpose modeling language for systems engineering applications
UML	unified modeling language

Résumé Chapitre 1 – Introduction

Récemment, la robotique collaborative a fait des progrès impressionnants, travaillant en toute sécurité côte à côte avec les humains. Cependant, malgré toute l'attention médiatique accordée à ce sujet, il y a relativement peu de robots collaboratifs dans les applications industrielles par rapport aux applications des robots industriels standard [39]. En 2015, les ventes de robots collaboratifs ne représentaient que 1,6 % des 239 000 robots industriels vendus dans le monde [29]. En 2017, dernière année pour laquelle des statistiques sont disponibles, ce pourcentage est passé à environ 4 % des 381 000 robots industriels installés dans le monde [30]. Cela indique qu'en dépit de leur croissance rapide au cours des dernières années, leur potentiel n'a pas encore été pleinement atteint.

Il existe un grand potentiel dans le monde entier pour la réalisation d'applications industrielles faisant appel à la collaboration homme-robot (applications HRC), donnant aux fabricants un moyen d'accroître la productivité, d'améliorer l'ergonomie et la santé des travailleurs et d'améliorer la qualité des composants. Néanmoins, la robotique collaborative n'en est qu'à ses débuts. Ce n'est que maintenant que des limites sont définies en ce qui concerne les collisions entre les humains et les robots [83]. Une fois ces limites mieux connues, les risques seront présentés plus clairement aux travailleurs d'usine et au grand public dans le but de les faire accepter par le plus grand nombre. L'acceptation des travailleurs dans les usines est une question à laquelle les fabricants et les intégrateurs de robots sont très sensibles. Il existe un risque important de réaction défavorable du public si des blessures et/ou des décès surviennent en raison de considérations de sécurité insuffisantes d'un robot collaboratif. C'est pourquoi de nombreux experts de la sécurité dans les usines ont un état d'esprit conservateur.

Pour en revenir à la question du nombre relativement faible de nouvelles installations de robots collaboratifs, l'une des raisons possibles de cette situation est que, malgré la publication récente de normes de sécurité pour la robotique industrielle, les exigences de sécurité strictes posent un défi difficile aux intégrateurs de systèmes et aux concepteurs d'applications robotiques. La planification des applications HRC faisant appel à la collaboration homme-robot (applications HRC) suit actuellement un processus non linéaire et hautement itératif qui présente souvent des inconvénients inattendus. En conséquence, les systèmes complets sont souvent coûteux et leur conception, leur construction et leur validation demandent beaucoup de temps. En outre, l'incertitude qui règne pendant la phase de conception conduit souvent à des situations où un prototype est d'abord construit et validé afin d'arriver à des estimations réalistes des paramètres du processus, ce qui entraîne des coûts supplémentaires.

Un certain nombre de facteurs technologiques importants ont eu lieu qui permettent aux robots de devenir collaboratifs. L'augmentation de la puissance et de la vitesse de calcul, ainsi que les capteurs permettant de détecter l'environnement autour d'un robot ont été des facteurs déterminants. Alors que la première génération de robots industriels était constituée de lourds géants de l'acier déplaçant de grandes masses à grande vitesse mais pratiquement aveugles au monde réel, les robots peuvent utiliser des capteurs de plus en plus sensibles et dotés d'une grande puissance de calcul pour prendre des décisions judicieuses en fonction des images des caméras et d'autres données des capteurs.

L'introduction récente de normes de sécurité pour les robots collaboratifs a contribué à faciliter leur utilisation. Les normes les plus pertinentes pour la sécurité des applications HRC sont les normes ISO 10218-1 et -2, ainsi que la norme ISO-TS 15066. La norme ISO 10218-1 décrit les exigences générales de conception pour la

sécurité des robots industriels destinés à être utilisés dans des applications collaboratives et spécifie la nécessité de systèmes de contrôle de sécurité, qui permettent l'arrêt et la reprise contrôlés des mouvements du robot. L'ISO 10218-2 s'adresse aux intégrateurs de systèmes et décrit les dangers spécifiques à un système complet, les moyens de s'en prémunir et les exigences à respecter lors de la mise sur le marché d'un système. Selon la norme ISO/TS 15066, l'homme doit être protégé contre les dangers dans le cadre d'une opération de collaboration par un mode de protection spécifique. Il existe actuellement quatre modes de protection différents définis dans la spécification technique. Chaque mode comporte des exigences de sécurité spécifiques et une spécification détaillée des mesures de protection obligatoires et des fonctions de sécurité

L'approche adoptée par les trois normes est une acceptation du fait que chaque application est différente et que les questions concernant le processus, le choix des composants de sécurité et leur impact sur l'homme et son travail doivent trouver une réponse spécifique à chaque cas. En aucun cas, les robots collaboratifs ne doivent causer de blessures aux humains travaillant à proximité ou avec eux.

L'objectif général de ce travail est d'étudier comment mieux concevoir, évaluer et assurer la sécurité des applications HRC protégées par la surveillance de la vitesse et de la séparation (SSM). Cela inclut des méthodes pour simplifier la manière de garantir la sécurité lors de la conception, de la planification, de l'évaluation et de la validation de ces applications. Actuellement, il peut s'écouler des années entre la première idée et la validation finale du système, et le processus n'est pas toujours simple.

Les applications comportant un contrôle de la vitesse et de la séparation seront au centre de ce travail en raison de leur pertinence industrielle actuelle. Une étude récente de l'IFR [30] a indirectement souligné l'importance de la MSS en reconnaissant que la majorité des applications du CRH mises en œuvre se caractérisent par une coopération séquentielle. En séparant le robot, ses outils et la pièce de l'homme, le concepteur d'une telle application dispose de plus d'options concernant la géométrie de la pièce, la charge utile du robot et la portée du robot.

Le processus de conception, qui commence dès la première esquisse et se termine avec l'application en cours d'exécution, exige un effort important pour répondre aux exigences de sécurité spécifiées par les normes et réglementations pertinentes. Les intégrateurs de systèmes et les utilisateurs finaux de robots collaboratifs doivent prendre en compte les aspects de sécurité lors de la planification ou après avoir apporté des modifications à leurs systèmes. Les interdépendances entre les opérateurs humains, les composants technologiques (robot, outils, capteurs), l'application (le processus, l'environnement, le rôle de l'homme) et les normes de sécurité sont complexes. Par conséquent, les sous-questions abordées dans cette thèse comprennent:

- *Quel est le processus actuel de conception et de prise en compte de la sécurité des applications HRC?*
- *D'où viennent les lacunes de ce processus (du point de vue de la sécurité)?*
- *Ces lacunes peuvent-elles être comblées par une nouvelle approche méthodologique et/ou de nouveaux outils d'ingénierie?*
- *Quelles sont les exigences relatives à cette nouvelle approche et à quoi peut ressembler une mise en œuvre?*
- *Quelles sont les interdépendances entre les différentes parties du système (homme, robot, outil, environnement, processus, etc.) et comment s'influencent-elles les unes les autres?*
- *Comment puis-je utiliser cette connaissance des interdépendances pour améliorer la conception et comparer d'autres conceptions?*

- *Ces connaissances peuvent-elles être mises à profit?*
- *Définir sur quoi la recherche future sur les capteurs de sécurité devrait se concentrer?*

Cette thèse est composée de plusieurs chapitres qui présentent l'état de l'art et expliquent ensuite la méthodologie développée pour la prise en compte de la sécurité lors de la planification et de la validation des applications HRC. Le travail est divisé comme suit :

1. État de l'art
2. Exigences et spécification de l'architecture de la nouvelle méthodologie
3. Développement de modèles de capteurs pour la surveillance de la vitesse et de la séparation
4. Introduction d'applications industrielles exemplaires de CRH
5. Études de cas d'utilisation avec des applications exemplaires et comparaison

État de l'art

Les processus de planification et de prise en compte des exigences de sécurité des applications industrielles faisant appel à la collaboration entre l'homme et le robot comprennent la connaissance des réglementations et des normes pertinentes en matière de robotique, les méthodes d'attribution des tâches entre l'homme et le robot travaillant en collaboration, et les méthodes d'identification des dangers et d'atténuation des risques. L'approche proposée applique des méthodes d'ingénierie des systèmes basées sur des modèles, il sera donc également important de discuter de l'état de l'art pour l'utilisation de l'ingénierie des systèmes en robotique.

Spécification de la nouvelle méthodologie

Dans cette section, les exigences relatives à la nouvelle méthodologie seront définies du point de vue des parties prenantes qui sont actuellement impliquées dans les processus de conception et de mise en service. Ces exigences sur la méthodologie seront utilisées dans une évaluation finale de la méthodologie pour valider si les objectifs de cette thèse ont été atteints.

Sur la base des exigences de la nouvelle méthodologie et en utilisant les concepts d'ingénierie des systèmes, l'architecture de la nouvelle méthodologie sera spécifiée. Cela comprend l'identification des principales composantes et la manière dont elles fonctionneront ensemble afin d'atteindre les résultats souhaités.

Développement de modèles de capteurs pour la surveillance de la vitesse et de la séparation

Un aspect particulièrement important de la méthodologie globale est l'adaptation des modèles de capteurs existants pour inclure des informations liées à la sécurité. Après une première description des aspects de sécurité pertinents en liaison avec le mode de sauvegarde Surveillance de la vitesse et de la séparation, après un aperçu des capteurs qui peuvent être utilisés pour la surveillance de la vitesse et de la séparation, les descriptions des modèles pour un groupe sélectionné de capteurs seront dérivées et expliquées.

Introduction d'applications exemplaires

Dans cette section, deux applications industrielles exemplaires qui sont actuellement réalisées entièrement manuellement seront présentées. L'un des cas d'utilisation comprendra un robot industriel traditionnel avec une charge utile > 15 kg,

et l'autre emploiera un robot léger avec une charge utile <15 kg. Après la description des cas d'utilisation, le processus de conception selon les méthodes traditionnelles sera expliqué pour les deux cas d'utilisation. Ces résultats serviront de référence à laquelle les résultats de la nouvelle méthodologie seront comparés dans la section suivante.

Études de cas d'utilisation avec des applications exemplaires

La méthodologie proposée et la mise en œuvre initiale des outils d'ingénierie qui en découlent seront utilisées pour examiner les deux applications exemplaires présentées dans la section précédente. Une comparaison des résultats et une discussion de la nouvelle méthodologie concluront cette section.

1 Introduction

Recently collaborative robotics have been making impressive gains. There have been many reports of new applications featuring them in industrial production, working safely side by side next to humans [25] [26] [27] [28]. However, despite these headlines and all the media attention paid to the subject, there are relatively few collaborative robots in industrial applications compared to standard industrial robots applications [39]. In 2015, sales of collaborative robots were only 1.6% of the 239,000 industrial robots sold worldwide [29]. In 2017, the latest year for which statistics are available, that percentage has risen to approximately 4% of the 381.000 industrial robots installed globally [30]. This indicates that despite their fast growth in the last few years, their full potential has not yet been reached.

Robotics is an important part of manufacturing today, as reflected in the attention and public funding initiatives from different nations worldwide. In Europe, the SPARC Public-Private Partnership [31] was the largest civilian robotics innovation program at the time of its initial contractual agreement on December 2013, with over €700M in funding from the European Commission for the period from 2014-2020. The Made in China 2025 (MIC2025) initiative, announced in 2015, was a strategic plan put forth by the Chinese government [21]. It focused on moving Chinese manufacturing capabilities towards the production of higher value products and services. This covered a large number of fields, from pharmaceuticals to aerospace, automotive, semiconductors, IT, and robotics, and committed roughly $300 billion US dollars to achieve this plan.

There is a large potential worldwide for realizing industrial applications featuring human-robot collaboration (HRC applications), giving manufacturers a means to increase productivity, improve worker ergonomics and health, and improve component quality. Nevertheless, collaborative robotics are still in a very early state. Only now are limits being defined regarding collisions between humans and robots [83]. Once these limits are better known, the risks be more clearly presented to factory workers and the general public with the aim of gaining widespread acceptance. Worker acceptance in factories is an issue that robotics manufacturers and integrators are very sensitive towards. There is a large risk of a public backlash were injuries and/or fatalities to occur due to poor safety considerations of a collaborative robot. This puts many safety experts in factories in a conservative mindset.

Returning to the topic of relatively low numbers of new installations of collaborative robots, one possible reason for this situation is that, despite the recent publication of safety standards for industrial robotics, the strict safety requirements pose a difficult challenge for system integrators and robotics applications designers. Planning HRC applications featuring human-robot collaboration (HRC applications) currently follows a nonlinear and highly iterative process that often features unexpected drawbacks [41, 36, 34, 33, 53, 71]. As a result, the complete systems are often costly and require large amounts of time to design, build, and validate. Furthermore, uncertainty during the design phase often leads to situations where a prototype is first built and validated in order to arrive at realistic estimates for the process parameters, leading to additional costs.

1.1 Overview of key enabling technologies for collaborative robots

A number of important technological factors have taken place which allow for robots to become collaborative. Increases in computing power and speed, as well as sensors for detecting the environment around a robot have been driving factors. Where the first generation of industrial robots were heavy steel giants moving large masses at high speeds but practically blind to the real world, robots can use increasingly sensitive sensors with large amounts of computing power to make meaningful decisions based on camera images and other sensor data.

The recent introduction of safety standards for collaborative robots has helped facilitate their usage. The most relevant standards for the safety of HRC applications are the ISO 10218-1 and -2, as well as the ISO-TS 15066. The ISO 10218-1 describes the general design requirements for safety of industrial robots for use in collaborative applications and specifies the need for safety-rated control systems, which allow for controlled stopping and restarting of robot motion. The ISO 10218-2 addresses systems integrators and describes the hazards specific to a complete system, the means for safeguarding against them, and the requirements when introducing a system to the market. According to ISO/TS 15066, the human in a collaborative operation must be protected against hazards through a specific safeguarding mode. Currently there are four different safeguarding modes defined in the technical specification. Each mode has specific safety requirements, and a detailed specification of mandatory protection measures and safety functions. Table 1 offers a brief summary of the four safeguarding modes and Figure 1 shows a few examples of how these safeguarding modes look in practice.

Table 1: Overview of the four modes for safeguarding HRC applications from the ISO/TS 15066.

SAFEGUARDING MODE	DESCRIPTION
Safety-rated monitored stop	In this mode, the robot stops before the operator enters the collaborative space. The operator can then enter the collaborative workspace to carry out their intended task. The robot resumes non-collaborative operation and continues its working process only after the operator has exited the collaborative space.
Speed and separation monitoring	The robot system and operator may move concurrently in the collaborative space. Risk reduction is achieved by maintaining a minimum separation distance between operator and robot at all times. This distance is dependent on the speed of the approaching human and the robot, and the robot speed can be varied to maintain the minimum separation distance. When the separation distance decreases to a value below the minimum value, the robot system must in the final instance stop its motion.
Hand-guiding	Hand guiding refers to a mode in which motion commands of a hand-operated device are directly transformed into robot motion. In this mode, the robot carries out a safety-rated monitored stop before the operator enters the collaborative space. Then, the intended task is carried out by actuating an appropriate guiding devices manually that is located at or in close proximity to the robot tool.
Power and force limiting	In this mode, physical contact between the robot (and including any physically connected components such as eventual tools and work pieces) and the human is allowed and even necessary to carry out the process properly. Risk reduction is achieved either through inherently safe means or through a safety control system. Collaborative operation with limited robot power and force requires specifically designed robot that are designed for this particular kind of safeguarding mode.

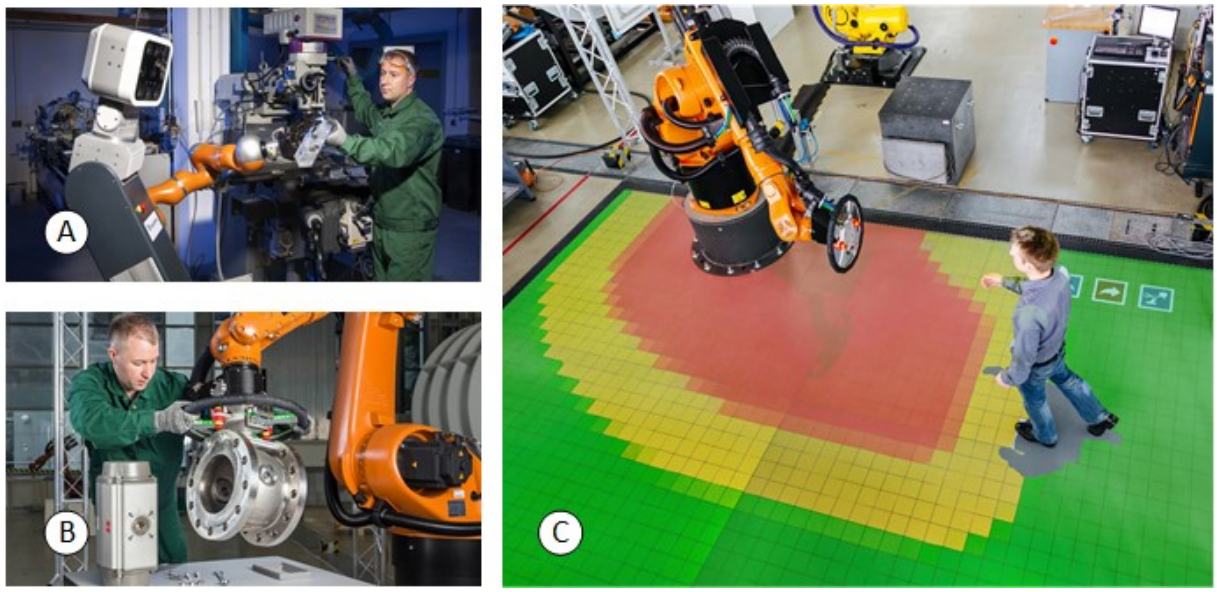


Figure 1: Examples of HRC applications with different safeguarding modes: A) Power and Force Limiting; B) Hand-guiding; C) Speed and separation monitoring. © Fraunhofer IFF

The approach taken by all three standards is an acceptance of the fact that every application is different and questions regarding the process, the choice of safety components, and their impact on the human and their work need to be answered specifically for each case. Under no circumstances should collaborative robots cause injuries to humans working near or with them.

1.2 Research objectives and contributions

The overall aim of this work is to investigate how to better design, evaluate and ensure the safety of HRC applications safeguarded by speed and separation monitoring (SSM). This includes methods to simplify how to ensure safety when designing, planning, evaluating, validating such applications. This currently can take years from the first idea to the final system validation and is not always a straightforward process.

Applications featuring speed and separation monitoring will be the focus of this work due to their current industrial relevance. A recent study from the IFR [30] indirectly highlighted the importance of SSM through the recognition that the majority of HRC applications implemented feature sequential cooperation. By separating the robot, its tools and the workpiece from the human, the designer of such an application has more options regarding part geometry, robot payload, and robot reach.

The design process, starting from the first sketch and finishing with the application running, requires a large effort in order to meet the safety requirements as specified by the relevant standards and regulations. System integrators and end-users of collaborative robots need to consider safety aspects when planning or after implementing changes to their systems. The interdependencies between the human operators, the technological components (robot, tools, sensors), the application (the process, the environment, the role of humans) and the safety standards are complex. Therefore, sub-questions addressed in this thesis include:

- *What is the current design process for designing and considering safety of HRC applications?*
- *Where do the shortcomings in the process arise (from a safety perspective)?*

- *Can these shortcomings be addressed through new methodological approach and/or new engineering tools?*
- *What are the requirements on such a new approach and how can an implementation look?*
- *What are interdependencies between individual parts of the system (human, robot, tool, environment, process, etc.) and how do they affect each other?*
- *How can I use this knowledge of the interdependencies to make better designs and compare alternate designs?*
- *Can this knowledge be leveraged:*
    - *Define where future research on safety sensors should be focused?*

1.3 Thesis structure

This thesis is made up of several chapters that present the state of the art and then explain the methodology developed for consideration of the safety during the planning and validation of HRC applications. The work is divided as follows:

1. State of the art
2. Requirements and specification of the architecture of the new methodology
3. Model development of sensors for speed and separation monitoring
4. Introduction of exemplary industrial HRC applications
5. Use-case studies with exemplary applications and comparison

State of the art
The processes of planning and considering safety requirements of industrial applications featuring human robot collaboration includes knowledge of relevant robotics regulations and standards, methods for assigning tasks between humans and robots working collaboratively, and methods for identifying hazards and mitigating risks. The proposed approach applies model-based systems engineering methods, therefore it will also be important to discuss the state of the art for the use of systems engineering in robotics.

Specification of the new methodology
In this section the requirements on the new methodology from the perspective of the stakeholders who are currently involved in the design and commissioning processes will be defined. These requirements on the methodology will be used in a final evaluation of the methodology to validate whether the goals of this thesis have been met.
Based upon the requirements of the new methodology and using systems engineering concepts, the architecture of the new methodology will be specified. This includes the identification of the major components and how they will work together in order to achieve the desired results.

Model development of sensors for speed and separation monitoring
A particularly important aspect of the overall methodology is adaptations to existing models of sensors to include safety-related information. After an initial description of the relevant safety-related aspects in conjunction with the safeguarding mode Speed and Separation Monitoring, After an overview of sensors that can be used for speed and separation monitoring, the model descriptions for a selected group of sensors will be derived and explained.

Introduction of exemplary applications

In this section, two exemplary industrial applications that are currently manually executed will be introduced. One use-case will feature a traditional industrial robot with a payload > 15 kg, and the other will employ a lightweight robot with a payload < 15 kg. Following the use-case descriptions, the design process according to traditional methods will be explained for both use-cases. These results will serve as a baseline against which results from the new methodology will be compared in the following section.

Use-case studies with exemplary applications

The proposed methodology and the initial implementation of the engineering tools based upon them will be used to consider the two exemplary applications introduced in the previous section. A comparison of the results and a discussion of the new methodology will conclude this section.

Résumé Chapitre 2 – État de la technique

Dans cette thèse, le terme "robot collaboratif" sera dérivé de la définition de l'ISO 10218 pour le fonctionnement collaboratif, ce qui signifie "...un état dans lequel des robots spécialement désignés travaillent en coopération directe avec un humain dans un espace de travail défini". Un robot collaboratif est donc tout robot industriel utilisé dans le cadre d'une opération collaborative. Cette définition n'impose aucune restriction quant à la forme, la charge utile ou d'autres caractéristiques physiques du robot, mais se concentre sur la manière dont ce robot est utilisé.

Les applications faisant appel à la collaboration homme-robot (applications HRC) consistent en un ensemble spécifique de composants et d'éléments interdépendants. Parmi les plus importants, on trouve l'homme, le robot, le processus, l'environnement, les outils utilisés par le robot, ainsi que des détails concernant l'intégration du système. Ces composants et leurs relations les uns avec les autres constituent l'espace de conception dont disposent les ingénieurs lorsqu'ils créent de nouvelles applications.

Cette section commence par décrire comment les applications HRC sont conçues dans la pratique industrielle actuelle. Le processus de conception, conformément à la directive Machines 2006/42/CE, peut être décrit par un flux de travail qui commence par un concept initial pour une application HRC, et se termine par une description du concept de sécurité final. Les différentes étapes du processus de conception sont les suivantes :

- Idée générale de fonctionnement en collaboration
- Conception axée sur la sécurité
- Évaluation des exigences générales et essentielles
- Modéliser le processus, assigner des tâches à l'homme et au robot
- Définir les limites et les exigences du système
- Identification des dangers et évaluation des risques
- Élimination des dangers et atténuation des risques
- Examen de la conception

Cette approche est très itérative et combine l'expertise d'une série de disciplines d'ingénierie distinctes, notamment l'ingénierie mécanique, électrique et de contrôle, ainsi que la santé et la sécurité au travail. Un certain nombre d'outils logiciels sont utilisés, notamment des programmes de CAD ou de simulation, des documents et des feuilles de calcul, avec un flux de données qui n'est pas géré de manière cohérente (par exemple, les données doivent souvent être transférées manuellement d'un système logiciel à un autre, ce qui augmente la marge d'erreur). Certains objectifs de design clés, notamment la dimension au sol, le temps de cycle et les coûts globaux, sont présentés et expliqués.

Après cette description du processus de conception, l'état de l'art des approches d'ingénierie des systèmes en robotique en généralDet de la tâche de conception des applications HRC est examiné. Les principaux thèmes de recherche également étudiés sont l'ingénierie des systèmes, l'ingénierie des systèmes, l'ingénierie des systèmes basée sur des modèles, l'analyse des risques basée sur des modèles pour la robotique, les ontologies et l'industrie 4.0.

L'ingénierie des systèmes (SE) est un domaine largement défini, qui cherche à comprendre les systèmes créés par l'homme en se concentrant sur l'ensemble du système par opposition aux parties [51, 49]. Elle commence par la définition des besoins à un stade précoce et par le rassemblement de différentes disciplines d'ingénierie pour concevoir et valider une solution tout en tenant compte de l'ensemble du problème. Cette approche a été utilisée avec succès pour un certain nombre

d'autres industries présentant un niveau de complexité élevé, notamment la défense [55], l'aérospatiale [56] et l'industrie automobile. Les exemples montrent clairement que la SE est particulièrement bien adaptée à l'analyse d'un système complexe tel que les applications HRC. Dans ce cas, l'application peut être considérée comme un système de systèmes, car les composants de l'application HRC comprennent le robot, ses outils, les senseurs de sécurité et le système de contrôle de sécurité, les opérateurs humains, l'environnement et d'autres systèmes de production (y compris d'autres applications HRC), dont beaucoup peuvent eux-mêmes être représentés comme des systèmes complexes. L'application de la SE peut contribuer à réduire la complexité, permettra une meilleure réutilisation des artefacts d'ingénierie, y compris les programmes. Bien qu'il existe quelques exemples d'application de l'ingénierie des systèmes à la robotique [58], il n'existe aucun exemple connu de son application dans le but de soutenir le développement et la mise en service de la robotique collaborative dans l'industrie dans les phases de conception et de développement du cycle de vie.

Comme les travaux de ce doctorat ont également des implications pour les robots contrôlés par intelligence artificielle, l'état de l'art en matière de sécurité de la robotique industrielle et intelligence artificielle est également brièvement abordé. Les normes les plus importantes pour la sécurité des robots industriels sont analysées pour comprendre comment les robots contrôlés par intelligence artificielle seraient manipulés. L'une des normes harmonisées de type A les plus pertinentes pour la robotique est la norme EN ISO 12100, qui spécifie les principes généraux pour l'évaluation et la réduction des risques. Le concept de comportement déterministe joue ici un rôle important, car l'évaluation des risques doit être concrète et tenir compte des spécifications et des paramètres de fonctionnement du système robotique. Par exemple, la charge utile, la géométrie de la pièce et les conditions environnementales jouent toutes un rôle important dans l'analyse des risques. Un robot transportant des objets tranchants représente un risque différent de celui d'un robot transportant des pièces lourdes ou de petits objets contondants. D'autres paramètres tels que le programme du robot, qui comprend l'enveloppe opérationnelle ainsi que les vitesses du robot, ont une incidence directe sur les paramètres liés à la sécurité, comme la taille de la distance de protection minimale ou la puissance et la force appliquées à une personne en cas de collision.

Les normes ISO 10218-1 et -2, ainsi que la norme ISO/TS 15066 précisent ce qui doit être validé par une mesure après la mise en place physique du système. Les exigences relatives à la validation des fonctions de sécurité, telles que le respect des limites de force et de puissance pour des parties spécifiques du corps, s'appliquent à un programme spécifique. Cette validation se fonde sur le concept de système robotique déterministe, où tous les paramètres et configurations pertinents sont documentés dans l'évaluation des risques. En résumé, les méthodes actuelles pour assurer la sécurité de la robotique collaborative reposent sur le concept de comportement déterministe du robot qui a été programmé par un humain et validé pour fonctionner correctement dans des conditions opérationnelles spécifiques. Ceci est en opposition diamétrale avec les possibilités offertes par l'intelligence artificielle et l'apprentissage machine pour les tâches de perception et de prise de décision. L'approche proposée dans cette thèse cherche à améliorer cette situation en fournissant un moyen de valider en simulation la sécurité d'applications spécifiques. Un effet secondaire de cette approche est que la validation pourrait également être utilisée pour les applications HRC qui impliquent des programmes définis par l'IA.

Lors de l'examen du processus actuel, les points suivants ressortent :

- Différents outils logiciels d'analyse. L'outil de CAD / simulation est le principal moteur du processus de développement, mais du point de vue de la sécurité, de nombreuses questions restent sans réponse. Passer d'un outil logiciel à l'autre (voir la vue d'ensemble dans le tableau 3) implique le transfert manuel de données de l'un à l'autre, ce qui entraîne des erreurs et des incohérences dans les données du système.
- Le processus est extrêmement itératif, en raison de la complexité et des options dont dispose le concepteur. Cela signifie qu'il n'y a aucun moyen de savoir si la solution est la meilleure, et la comparaison des solutions implique de nombreux ensembles de données différents qui deviennent difficiles à manipuler. Si le concepteur décide d'utiliser une combinaison de clôtures et de scanners, ou de clôtures et de rideaux lumineux, tous les calculs et hypothèses faits lors des phases de conception précédentes doivent être revus. Cela implique de s'entretenir avec des collègues ayant des compétences différentes (par exemple, en parlant avec des experts en capteurs, des experts en sécurité). Un changement à un endroit peut avoir de graves conséquences du point de vue de la sécurité.
- Les méthodes de calcul de la distance minimale de séparation requise sont limitées. Les calculs actuels sont basés sur les hypothèses les plus pessimistes et ne reflètent pas la réalité du système. La validation des hypothèses nécessite des mesures, ce qui signifie que le matériel doit être mis en place avant que le concepteur puisse être sûr que les objectifs de conception ont été atteints. En outre, les hypothèses peuvent facilement changer et les interdépendances peuvent facilement être négligées. Enfin, l'utilisation des hypothèses les plus pessimistes pour l'ensemble du programme de robot, bien que facile à gérer et à comprendre, peut conduire à de faibles performances et à des vitesses de robot plus faibles que nécessaire.

Le travail décrit dans cette thèse est basé sur l'approche de l'ingénierie des systèmes. Cette approche a été choisie parce que l'ingénierie des systèmes est capable de soutenir différentes disciplines d'ingénierie pour concevoir et valider des solutions. Elle se concentre sur l'ensemble du système. Elle a été appliquée avec succès à une série d'industries comportant des systèmes complexes (par exemple, l'aérospatiale, le secteur militaire, l'automobile) et est applicable aux applications HRC. Celles-ci peuvent être considérées comme un système de systèmes, car elles consistent en une combinaison de composants individuels comprenant le robot, ses outils, les capteurs de sécurité et le système de contrôle de sécurité, les opérateurs humains, l'environnement et d'autres systèmes de production (y compris d'autres applications HRC), dont beaucoup peuvent eux-mêmes être représentés comme des systèmes complexes.

2 State of the Art

The term collaborative robotics itself has become a source of confusion. The International Federation of Robots distinguishes between two types of collaborative robots, namely those that were specifically designed for collaborative operation in compliance with the ISO 10218-1, and those that were not [30]. Often the term “cobot” is used in popular science articles to refer to a class of lightweight robots that typically a low payload (max. 14 kg) and that were designed for allowing contact with humans under certain circumstances. In this thesis the term collaborative robot will be derived from the ISO 10218 definition for collaborative operation, meaning “…a state in which purposely designed robots work in direct cooperation with a human within a defined workspace.” A collaborative robot is therefore any industrial robot used in collaborative operation. This definition does not place restrictions on the form, payload, or other physical characteristics of the robot, but focuses on how it is used.

2.1 State of the art for designing HRC applications

In general, a HRC application consists of a specific set of the components and elements as shown in Table 2.

Table 2: Overview of essential components and elements of HRC applications.

COMPONENT / ELEMENT	DESCRIPTION
Human	This refers to any human who could be involved in the collaborative working area, from the operator to logistics personnel or bystanders passing through.
Robot	The robot working in the HRC application, regardless of how it is safeguarded.
Tool	The tool used by the robot to fulfil its intended tasks. This can range from a gripper for material handling to tools for processing mechanical parts and also includes tool-changers
Environment	The environment includes elements belonging to the working space such as fixtures, building structures, pathways, etc.
Process	The process refers both to the parts used in the intended tasks as well other aspects for the execution of the intended robot action such as required robot speeds or forces.
System integration	The term system integration refers to all components that are required for the implementation of the HRC application, including safety sensors, and PLCs.

These components and elements represent the design space that an engineer has available when designing a new HRC application. These components and elements exhibit a large number of interdependencies on each other, i.e. a change to one component can have a large effect on other elements. Additionally, the HRC applications designer needs to respect a number of requirements and constraints on the task, the environment, and human health and safety.

In industry, there are a number of domain specific tools for designing collaborative robotics applications. The standard method consists of a project manager working with mechanical and electrical engineers. The mechanical engineers work in CAD, possibly with a robotics simulation environment. They model the overall layout, the material flow, and define the type of collaboration with the human by also defining the human’s tasks. They check that the initial requirements are met and choose the robot type based on a few criteria such as payload, reach, and specific customer preferences. The electrical system can be designed in an electrical computer aided design (ECAD) program. There is usually no direct digital connection between the mechanical and electrical domains. Electrical components are physically modeled and the cable paths are considered by the mechanical engineers. Logical pathways, communication, and

electrical energy can be modeled in the ECAD. Once the system is physically ready, the programmer can begin the process of programming the robot. They rely on a variety of tools and middlewares, and can sometimes use the simulation built beforehand for their purposes.

The project manager oversees all sides and ensures that the status is compatible and that any updates or changes are communicated so that each individual team can update their individual models.

Usually the safety for a collaborative robotic application is important, and either an extra safety expert is brought into regular meetings or a member of one of the existing teams (electrical or mechanical). There are a number of dedicated tools such as Sistema [73] which support the choice of the electrical components and help ensure conformity according to functional safety standard ISO 13849-1.

Looking at this, we can see that the engineering tools are fragmented. Any questions regarding “what-if” scenarios involve teams of people with competing interests, unclear requirements, and unclear means of ensuring that an optimal goal is reached. This is a particular challenge with HRC applications, which feature a large number of actors and components with many interdependencies that cannot be easily managed.

Figure 2 illustrates the planning process as it is carried out today. Starting from a concept and moving to productive operation with the HRC application, the iterative nature of the step “replanning” due to unfulfilled safety requirements can insert uncertainty into the process of designing, building, and operating an application featuring HRC, both in terms of time to completion and outcome.

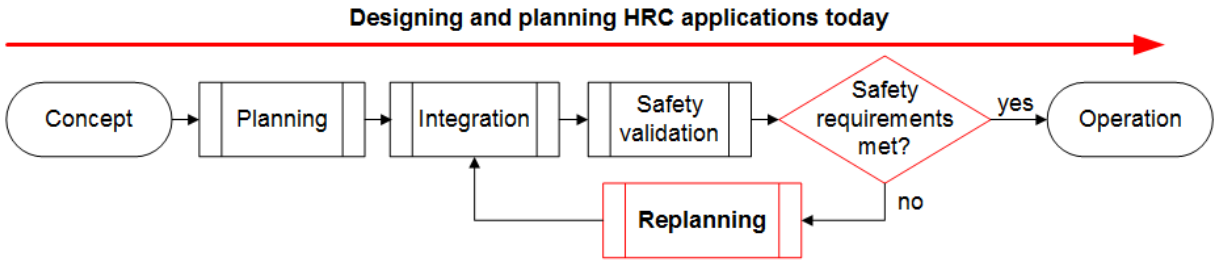


Figure 2: Planning industrial application featuring HRC today from the concept to operation

Figure 3 takes a closer look at the individual steps involved in the planning phase. In this diagram, the starting point is a general idea of collaborative application, meaning that the industrial application, which will feature a collaborative robot is well known, and that there is a rough idea of how the robot should assist the human operator. This also means that an initial choice for the type of robot (type, payload, size, position) has been made, as well as an initial layout plan for how the robot should be positioned in the environment. Previous work [17, 48] describes a decision-making framework for generating the layout of an HRC application and allocating tasks between robots and humans, with the aim in reducing the time and effort for generating a process plan. While these questions are valid and the methods they propose are sound, their work is less relevant for the safety-related considerations of the layout. Other research in this field [45] focuses on describing robot motions as small building blocks and then combining these for planning assembly tasks. While this work provides an interesting overview of the state of the art for robot assembly planning, this approach does not consider safety in the sense of ISO 10218-1 and -2 and is only peripherally relevant for this work.

Following this starting point, a safety oriented design is carried out. The concrete details of the design are then cross-references to ensure that the general and essential reqirements are met (e.g. do all safety components have Performance Level “d”, Category 3, etc.). Only once the general requirements (from the 10218-1) are met, does it make sense to look deeper at the process and to start detailing it, defining specifically which are done by the robot and which by the human, and what kind of HRC is envisioned (as well as the corresponding safeguarding mode) for different steps. Following this, the system limits need to be defined (e.g. where are no-go areas for the robot, what are max robot speeds, etc.), and then a check whether the specific application requirements are met (is the system fast enough to complete it in the cycle time). After the requirements have been met, a hazard identification and risk evaluation are carried out. By this time, the design is quite advanced and it is possible to identify real, specific hazards. Therefore, instead of high-level hazard such as “Possible clamping of human body parts between robot and environment”, a specific one would be “Possible clamping human hand and upper arm between robot gripper and part carrier during process x, and again during process y.” Should certain risks be considered too high, a hazard elimination and risk reduction needs to be carried out. This can be achieved for example through additional safety sensors, through changes to the process, the environment, and/or the material flows. Here quite a lot of work has focused on applying different safeguarding techniques [16, 34, 44], and on methods for calculating the required safety distances [50, 34, 16, 33, 10]. Due to the sheer number of options the designer has at their disposal to adapt an HRC application, there are a large number of possible variations for a given application. Furthermore, as indicated in the flow chart, if the final system still poses significant safety risks to the humans, the designer has to start all over again. The concept is only complete once the risk has been sufficiently reduced, opening the path for further development for implementation.

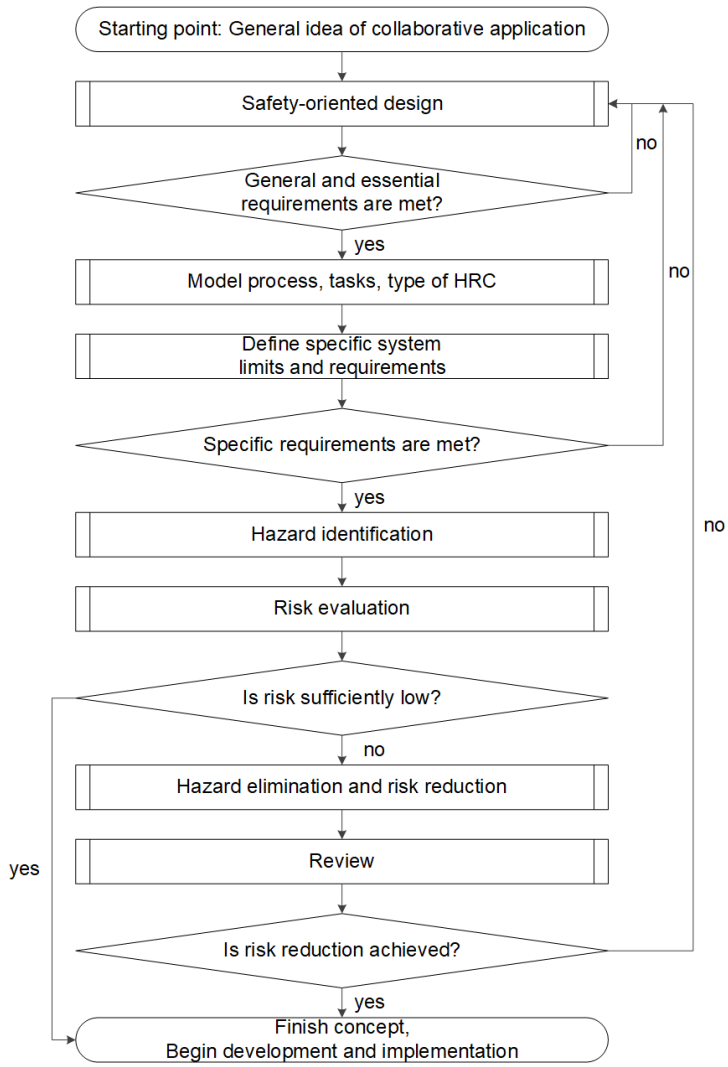


Figure 3: Flow model of different phases during concept (design) of a HRC application in manufacturing

In addition to the tasks involved in planning the HRC application, the designer also needs to consider the required steps to attain a CE mark [11]. Figure 4 shows the typical steps involved in this process. The engineering tools proposed in this thesis support designers in these tasks specifically during the steps “verify product specific requirements,” as well as “test the product and check its conformity”.

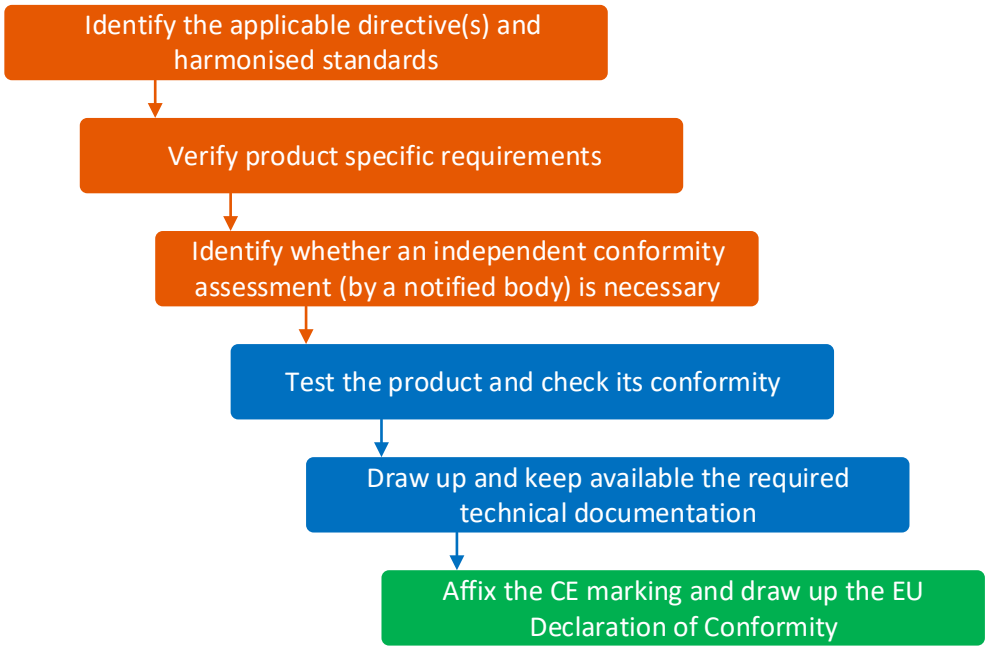


Figure 4: Path to CE Mark according to Machinery Directive 2006/42/EC

Figure 5 shows the generic lifecycle model from ISO/IEC/IEEE 15288 with additional clarification to highlight the processes involved in each lifecycle phase that are specific for HRC applications. This helps to highlight how the processes related to obtaining the CE mark from Figure 4 overlap with the concept processes from Figure 3. A further specific challenge for HRC applications is the issue that any changes to the robot program require a review of the safety validation and eventually an update to the CE mark.

Concept	Development			Production	Utilization / Support	Retirement
	HW	*SW*	*Safety Validation*	*Commissioning*	*Re-programming*	
				CE Mark	*Safety Validation*	
					CE Mark Renewal	

Figure 5: Life Cycle Model for HRC applications, based on Generic Life Cycle Model from ISO/IEC/IEEE 15288:2015

Returning to the state of the art description for how HRC applications are currently planned, and considering how often safety needs to be considered and validated across the entire lifecycle of an HRC application, the current decoupling the safety issues from the regular design processes is obviously problematic. The insights gained from this analysis of the state of the art will be used to derive the requirements on an improved approach for considering the safety of HRC applications.

2.2 Example of planning process with current methods

Building upon the generic flow model from Figure 3, the design process in accordance with the Machinery Directive 2006/42/EC, the most relevant standards, and with engineering tools currently available will be described. The following terms are used to identify the stakeholders involved in the design process:

- Designer: with this term, we are referring to the engineer(s) tasked with planning the robotics application. They can be process, electrical, or mechanical engineers by training and are concerned with all aspects of the robotic application. This includes from the mechanical side the layout planning, the mechanical design including tooling, the overall process planning. Relevant electrical and controls-related aspects to be considered include the overall electrical plan of system, e-Stop functionality (stop/ restart), choice of electrical components, integration into PLC or robot control system, programming (robot control and PLC program if necessary), choice of electric and electronic components including sensors for process control and safety.
- Safety expert: this is the person responsible for ensuring overall safety of the system, with particular view towards human occupational health and safety. Therefore, they are concerned with the ergonomics of all tasks, the hazard identification and risk analysis, and approval of risk mitigation measures (in collaboration with the designers). They are consulted by the designer (in the best case) at specific steps during the process of designing the application featuring HRC. In the worst-case, they are consulted at the end of the design process to review the status and offer suggestions for changes.

The description of the design process will also identify the types of software and/or documentation tools that the stakeholders use along the different phases of the design process.

The designer often uses design targets in order to evaluate the different designs. From the perspective of this thesis, the most important design goal is to maintain worker safety at all times. In the literature, it is not always entirely clear what is meant by being safe. In some instances, it is implied that the absence of an injury under controlled experimental circumstances constitutes proof of safety. For the purposes of this work, ensuring safety is achieved when the system conforms to the relevant laws, guidelines, and standards. This means that a risk analysis according to ISO 12100 has been carried out, that risk mitigation measures have been identified, that the robotics components meet the requirements from ISO 10218-1, and the complete robotic system the requirements from ISO 10218-2 and the ISO/TS 15066.

From the perspective of the designer, there are other production-oriented design targets that can be used to calculate a return-of-investment (ROI) for the system. This ROI can be used to evaluate whether a specific design will be implemented or whether the investigation will be stopped. In many cases, the most important reason for investigating a collaborative robotics solution is to create a significant improvement in operator ergonomics [15]. In these cases, the level of improvement to worker ergonomics can also be a deciding factor, beyond pure ROI considerations[1]. The following, key performance indicators are especially relevant for HRC applications:

- Costs – The system costs are often considered across the entire lifecycle, from design to implementation (e.g. hardware costs, engineering and installation costs), operating costs, maintenance costs, and de-commissioning costs. The designer has the ability to influence these costs through selection of type of HRC, sensor and robot choice.
- Floor space – The required floor space is important, both because the overall flow of materials, transport spaces, and human spaces place strong

[1] The author is not aware of any metrics currently in use by industry that relate an improvement in ergonomics to a cost, so that the ROI of an HRC application can also monetarily consider ergonomics improvements.

constraints on how much space can be used, and because floor space is indirectly associated with a specific overall cost.
- Cycle time – The achievable cycle time of an application is important both because it has a direct relation to the costs and overall production constraints (i.e. needs to fit into overall production cycle).

It is important to note that the priority for an individual production goal is very specific to the application in question and can vary. This implies that these production goals are part of the design space for the planner and the system design can be strongly influenced by which of these goals has the highest priority. These design targets will be used in this thesis to compare the outcomes of the traditional methodology for designing HRC applications with the proposed approach.

2.2.1 Starting point: General idea of collaborative application

As a starting point, the designer has a clear understanding of the application as it is currently manually carried out and has formulated an idea for how the robot should collaborate with the human operator in the application. The designer uses simple documentation to describe the application and to share ideas with other team members. Possibly an initial CAD layout is created.

2.2.2 Safety oriented design

Safety oriented design means that the designer considers requirements from the Machinery Directive 2006/42/EC and related standards and follows a paradigm using the following safety-oriented design principles:

- Machinery must be designed and constructed so that it fulfills its function and can be operated, adjusted and maintained without putting persons at risk when these operations are carried out under the conditions foreseen. Furthermore, any reasonably foreseeable misuse of the machinery must be taken into account. The aim of measures taken must be to eliminate any risk throughout the foreseeable lifetime of the machinery including the phases of transport, assembly, dismantling, disabling and scrapping/recycling.
- In selecting the most appropriate methods, the manufacturer must apply the following principles, in the order given:
    - Eliminate or reduce risks as far as possible (inherently safe design and construction),
    - Take the necessary protective measures in relation to risks that cannot be eliminated,
    - Inform users of the residual risks due to any shortcomings of the protective measures adopted, indicate whether any particular training is required and specify any need to provide personal protective equipment.
- When designing and constructing machinery and when drafting the instructions, the manufacturer must consider not only the intended use of the machinery but also any reasonably foreseeable misuse thereof.
- The machinery must be designed and constructed in such a way as to prevent abnormal use if such use would lead to a risk. Where appropriate, the instructions must draw the user's attention to ways — which experience has shown might occur — in which the machinery should not be used.
- Machinery must be designed and constructed to take account of the constraints to which the operator is subject as a result of the necessary or foreseeable use of personal protective equipment.
- Machinery must be supplied with all the special equipment and accessories essential to enable it to be adjusted, maintained and used safely.

After the checklist is filled out and all design principles have been applied to the current phase, the designer can move to the next phase.

2.2.3 General and essential requirements

The general and essential health and safety requirements as described in the Machinery Directive 2006/42/EC need to be fulfilled by the system. A layout of the collaborative workspace that includes the positions of the main elements, including humans, robots, tools, parts) is required. The layout is commonly made using CAD/simulation software by the designer. A checklist in the form of a text document or spreadsheet can then be used to check that the essential requirements based upon the initial layout have been fulfilled. A sample checklist for this step is in the Appendix.

2.2.4 Model process, assign tasks

In this step, the designer creates a simple listing of the tasks that are required to carry out the process. Individual tasks include movement to specific places in the workspace (e.g. table, pallet), as well as physical manipulation (e.g. picking and placing parts). Ideally, the designer considers different tasks along the entire product lifecycle, from commissioning to productive operation and maintenance tasks. An example of a task model in a tabular form will be presented later, in Table 15 and Table 18.

Using the methodology described by [16], it is also possible to identify the type of HRC that the designer would like to use. As described in that work, this decision is based on the available and meaningful types of collaboration that suit the task. The outcome of this task is therefore a process model, which highlights the process-related targets, the actions carried out to achieve the targets, and the resources used. Each process step has an associated form of collaborative operation and an explicit definition the role of the human. Consecutive operations with the same form of collaboration can be clustered into groups for simplification.

2.2.5 Define system limits and requirements

Based upon the floor plan and the task models previously defined, the designer specifies the system application limits and requirements. The limits serve to specify the intended use of the system, and include descriptions of the machines and process such as:

- the workspace of the robot and any spatial limits;
- the configuration of the robot to reach all the critical positions within the workspace;
- the speeds the robot will move;
- the payloads carried
- the temporal limits
- safeguarding mode specific limits

At the conclusion of this step, the designer checks that the system limits and requirements for the specific safeguarding mode are fulfilled. If not, the designer needs to change specific aspects of the design, including:

- modifying the collaborative workspace or layout
- selecting another safeguarding mode
- selecting another form of collaborative operation

This can be an iterative process, analog to the replanning pathway from Figure 2. The limits selected and assumptions made about the process (e.g. payloads, speeds,

etc.) can be documented in a spreadsheet or a text file. Eventually any changes to the layout are manually transferred to the CAD/simulation models.

2.2.6 Hazard identification and risk evaluation

The hazard identification process for designing HRC applications normally focuses on mechanical hazards such as geometrical characteristics of parts a human could possibly come in contact with (e.g. sharp edges), parts in uncontrolled movement (e.g. flying objects due to loss of grip), or parts in controlled movement (e.g. kinetic energy of the robot). The safety expert also has to consider hazards due to neglect of ergonomic principles, such as tasks that are mentally challenging or not challenging enough or unhealthy body postures. Human error also needs to be considered here, due to the use of personal protective equipment through unauthorized modification to safety measures. There are a number of dedicated software tools available for execution and documentation of this process [75]. By this point, the designer and safety expert should have a list of detailed information about particular hazards and the task (and lifecycle phase) where they might occur.

When evaluating risk, the standards ISO 12100, ISO 13849 and IEC 61508 differentiate between mechanical risk assessment and assessment of functional safety of controls related components. The mechanical risk assessment focuses on hazards that are caused by physical arrangement of components, mechanical properties or moving parts. The goal is to determine the risk of such hazards, commonly understood as the product of damage and hazard probability. This probability of risk is later used to specify appropriate safety measures. The second paradigm addresses safety-related components of the robot control system, and in this case, the probability corresponds with failure cases instead of occurrence. The result of this risk evaluation leads to the specification of safety performance level and category (or safety integrity level) that must be met by the design of the safety components. This information is appended to the list if detailed information about the hazards and is used for the following step of determining the necessary safety measures. The hazards and risks can be documented in a spreadsheet or text file, or specialized software for safety evaluation such as Safexpert can be used. CAD/simulation serves merely to support the process of identifying the hazards.

2.2.7 Hazard elimination and risk mitigation

The goal of hazard elimination and risk mitigation is to actively ensure the safety and health of persons. This task therefore includes the development of a comprehensive safety concept. The decision for using specific safety measures considers the following aspects in the order (with descending relevance):

1. Safety requirements: Safe operation for each phase in the lifecycle of the HRC application (e.g. as determined by the risk evaluation and the general requirements)
2. Economic requirements: Costs for purchase, operation and disassembling
3. Process requirements: Ability to carry out the intended process properly
4. Operation Requirements: Usability

The last requirement is particularly important to avoid operators bypassing safety mitigation measures. A general rule of thumb is that the safe method should also be the easiest for the operator to execute.

The general principles for risk mitigation as listed in Section 2.2.2 need to be observed. A the end of this task, the designer and safety expert have collaborated to define the methods for either eliminating hazards through changes to specific elements of the HRC application (e.g. adding cushioning or changing part geometries to

eliminate sharp edges) or through risk mitigation means such as using sensors to stop robot motion before a collision can occur. The calculations for estimating safety-related factors such as size of safety zones are a part of this process and are typically calculated with spreadsheets. The choices for hazard mitigation are documented either in a text file or with the specialized software for safety evaluation used in the previous step.

2.2.8 Review of design

In this final step, the designer reviews the final safety concept and validates the solutions against the requirements on the system. The requirements are related to the form of collaboration used and can include measurements, e.g. to ensure that the forces and pressures in case of a collision are below the biomechanical thresholds. The EU project COVR [43] offers support in this area by developing protocols that offer step-by-step instructions on how to carry out these validation measurements for a wide variety of HRC applications across a number of domains (e.g. healthcare and rehab, industrial manufacturing, etc.). The results of the review are documented in a text file or spreadsheet and the results are also ideally manually entered back into the CAD/simulation model of the system.

2.3 State of the art for systems engineering approaches to robotics

A literature research was carried out to better understand how systems engineering can support this process. The following key words were searched in IEEE Explore and Elsevier databases and on the web (e.g. Semantic Scholar):

- Systems Engineering / System of Systems Engineering / Robotics
- Model based systems engineering / Robotics
    - Model-based software engineering, skills-based programming
    - Model-based risk analysis
        - UML, SysML
            - RobotML
    - Ontologies
        - Ontology development efforts
- Industry 4.0

2.3.1 Overview of systems engineering

Systems engineering (SE) is a broadly defined field, which seeks to understand man-made systems by concentrating on the whole system as distinct from the parts [51, 49]. It starts by defining requirements at an early stage and bringing together different engineering disciplines to design and validate a solution while maintaining consideration for the complete problem. This approach has been successfully used for a number of other industries featuring a high level of complexity, including defense [55], aerospace [56], and automotive industries. From the examples of how it was applied, it is clear that SE is particularly well suited for analyzing a complex system such as HRC applications. In this case, the application can be considered a system of systems, as the components of the HRC application include the robot, its tools, the safety sensors and safety control system, the human operators, the environment, and other production systems (including other HRC applications), many of which themselves can be represented as complex systems. The application of SE can help reduce complexity, will allow for better reuse of engineering artefacts including software, and most importantly from my point of view, using SE for HRC applications will lead to more cost-effective engineering efforts with a reduction in errors and

uncertainty about the final system. While there are a few instances of applying systems engineering to robotics [58], there is no known instance of its application with the aim of supporting the development and commissioning of collaborative robotics in industry in the life cycle phases of concept and development. In the following sections I will investigate these instances where systems engineering has been used in robotics.

2.3.2 Model-based software engineering

Model-based software engineering uses an SE approach to address the programming of a robotic system. Issues addressed by this work include the fact that writing software code is a time-consuming and expensive process, that this code is nevertheless often not reusable and can become obsolete with changes to the robotic systems' hardware configuration. The BRICS project [18] supported the idea of model based engineering and aimed to support components reuse in robotics through dissemination of best practices. The models used were however not sufficiently specified to allow for significant re-use.

Recent work in this field [59, 46] focuses therefore on the meta-models to allow for more generic modeling of the robotic system with the specific aim of automatically generating software code, both offline (prior to robotic action being initiated) and during run-time. Here an entire toolchain for modeling was developed in a UML environment. In these cases, the aim is for a programmer to specify high level tasks in the task model such as "grasp cup" versus programming a specific set of actions. Furthermore, generic interfaces were created so that the models could be used with a number of different specific tools for software creation. In addition to generating software for a robot based upon the models, other quality of service aspects such as run-time ability were able to be evaluated. This work focuses on the software engineer's point of view, and less on the overall design and consideration of safety aspects from the mechanical point of view.

When viewed according to the life cycle model from Figure 5, we can see that the focus of these efforts starts in the development process and is relevant throughout the production, utilization, and support phases. This is in contrast to the stated aim of this work to support designers and system integrators during the concept and development phases.

More recently, the H2020 project RobMoSys [40] seeks to build upon this body of work by offering cascaded funding to develop a larger set of models and engineering tools that will serve the entire robotics community. RobMoSys has funded a project called eItus – "Experimental Infrastructure Towards Ubiquitously Safe Robotic Systems using RobMoSys" [76] with the aim of extending the RobMoSys metamodels to include safety concerns. The solution is based on Papyrus/SysML and allows for the creation of a safety assessment during run-time in the form of fault injection analysis with the aim of validating software code. As we will see in the following section, support for the safety assessment is an important topic that is the focus of many researchers.

2.3.3 Model-based risk analysis

There has also been recent research focused on the use of model based engineering methods to support risk analysis for robotics [54, 60, 61, 66]. Earlier work [23] initially used UML as the modeling language to analyze the safety of a medical robot. In particular, the work focused on an analysis of the task and of possible human errors, to understand and handle human errors while working with the system. Further, the authors sought to support the risk analysis by identifying hazards and applying the Failure Modes, Effects and Criticality Analysis (FMECA) and FMEA [62] techniques. Further work [24] proposes an approach to apply the HAZard Operability (HAZOP)

technique with UML models. The authors chose the HAZOP method since it can be used earlier in the engineering process than FMECA, and that it can include human activity as a source of hazard. Also of interest is the author's focus on the use of use-cases, sequence and state diagrams to analyze the system. The system is not able to identify all hazards, but focuses on operational hazards linked to human-robot interactions. The method has been applied to a number of robots from research projects, including lightweight industrial robots, mobile manipulators, and an assistive robot for healthcare applications. Further work [67] describes a domain specific modeling language for robotic systems called RobotML, which is used to generate a fault tree analysis and supports formal verification methods.

While the topics of safety in the aforementioned works are covered, they are not addressed in a way that a mechanical engineer approaches the issue during the concept and development of an application featuring HRC as described in Sections 2.12.2. The size of safety zones and the overall effect of safety requirements on the environment, on the type of interaction, and on the overall process are not considered. Furthermore, this approach does not sufficiently address the concept of requirements engineering to ensure conformity with the collaborative robotics standards 10218-1 and -2, as well as the ISO-TS 15066.

Gribov et al. [21] use SysML models and requirements engineering to check whether the standard ISO/DIS 13482 is fulfilled in the early design stages. However, this work assumes that the robotics hardware is not changing and only focuses on software, including how to represent software issues in a formalized way and how to define an engineering process which provided evidence of safety and allows for software reuse.

The work described in this thesis to support the design of HRC applications can build upon these previous efforts. In particular, the application of systems engineering methodology and the use of requirements engineering to ensure conformity to a specific standard early in the design process.

2.3.4 SysML systems engineering modeling language

SysML was initiated in 2001 as an open-source specification project, and is an extension of a subset of the Unified Modeling Language (UML). Whereas UML was created with the intention of supporting software engineering, SysML was created as a general-purpose graphical modeling language specifically designed for representing complex systems consisting of hardware, software, people, facilities, and processes. In [65] three main pillars of Model-Based Systems Engineering are described as the modeling language, the modeling method, and the modeling tool, with a note that these three aspects are all independent of each other.

In SysML there are nine kind of diagrams that are supported. These are:

- Block definition diagram (BDD)
- Internal block diagram (IBD)
- Use case diagram
- Activity diagram
- Sequence diagram
- State machine diagram
- Parametric diagram
- Package diagram
- Requirements diagram

These diagrams and their hierarchical arrangements are shown in Figure 6. The modeling effort in SysML is limited to using these specific kinds of diagrams. An important distinction is that SysML models are more than just a grouping of diagrams (which could be made using simple software like Microsoft Visio), but contain semantic connections between the different modeled elements and as such are not simply static representations of interdependencies and relationships.

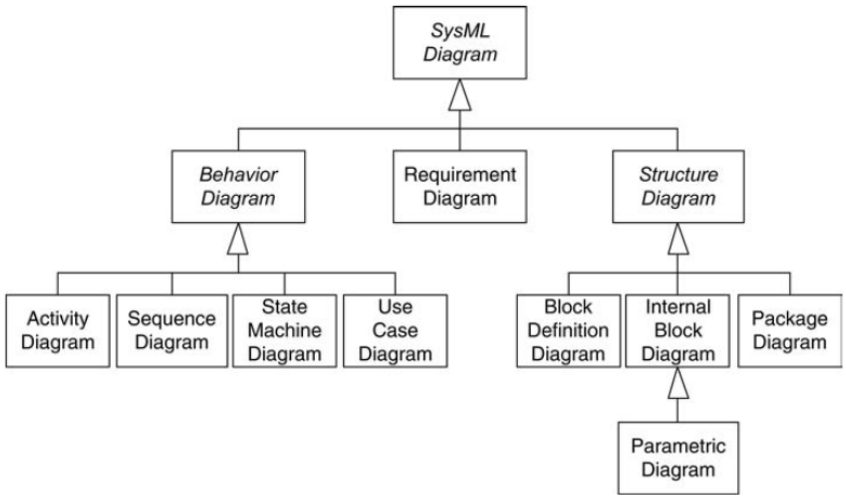


Figure 6: SysML diagram taxonomy [52]

These nine diagrams represent all of the modeling options available. In the literature [64], there are examples of simplified models for specific domains which contain a smaller sub-set of diagrams. In this work, a specific subset of these SysML diagrams will be used for modeling.

2.3.5 Ontologies for robotics

In researching model based system engineering approaches, the concept of ontologies and their use in robotics also becomes of interest. Ontologies formally describe a system so that it is machine understandable [80]. With the advent of the internet, there were a lot of efforts put into developing both general and domain specific ontologies to make content on the internet machine-readable and in an effort to organize information [70]. When creating models of robotic systems that are reusable, existing ontologies could be a useful source of inspiration or even a starting point. Nevertheless, there is the caveat that ontologies are imperfect [70] and are made to represent a specific domain or area of interest [69].

Notable efforts regarding robotics and sensor ontologies include the Roboearth project [68], whose aim was to use existing information from the World Wide Web (WWW) to support robotic task accomplishment and to create a world wide web for robots so that they can exchange information between one another and which focused on the semantic representation for actions, objects and environments. Further work in this direction [47] focused on using these ontologies to support AI-assisted task planning. In this case, the aim was to let a robot reason about its movements and generate executable motions that are adaptable for different robots, objects and tools.

Another notable effort at developing an ontology for the robotics community is the IEEE Core Ontology for Robotics and Automation (CORA) [1]. It has only specified a relatively high-level ontology, but it has formed the basis for the CORAX ontology, which provides concepts such as design, environment and interaction. The RPARTS ontology, also a subset of the CORA ontology, has defined concepts useful for designing HRC applications such as “Robot sensing part”. The position (POS) ontology

specifies concepts relating to position, orientation and pose, and is of course important for describing robotics movement. Working groups are currently working on refined the ontology by describing tasks and robotic skills from a variety of robotics domains, including industrial robotics, in order to make the CORA more usable within the community [72]. Sensors are also an integral part of a robotic system featuring HRC, and there have been other efforts at creating ontologies to describe these. Eastman et al. [20] present a good overview of currently available sensor ontologies. These typically focuses on technical attributes such as physical dimensions and the sensor output signals.

There are currently no ontologies dedicated to the safety-related aspects of HRC applications. The approach proposed in this thesis does not focus on the development of new ontologies. Nevertheless, in order to create reusable models and build upon existing knowledge, ontologies could prove to be a useful reference. The approach described in this thesis will therefore review these existing ontologies, in particular with regard to robotics, sensors, and environment, when creating models. The results of this thesis could be used for additions to existing efforts.

2.3.6 Industry 4.0 Framework

Another interesting modeling effort that has recently begun is the Industry 4.0 reference architecture [9]. It seeks to create a unified model of industrial components that are parts of “Industry 4.0” to allow for better design, interoperability, and usage. As a means to describe the parts completely, it combines a number of established standards including the IEC 62890 for defining the life cycle and value stream and the IEC 62264 and IEC 61512 for defining hierarchy levels. It is more engineering oriented than an ontology, and the aim is industrial standardization. Just like with ontologies, it would make sense to consider the framework when creating the models we propose. Any model that fully conforms with Reference Architectural Model Industrie RAMI 4.0 will be very rich with semantic information and will contain much more information than needed for modeling HRC applications. However in view of the fact that HRC applications are at the core of Industry 4.0, and with a forward view towards compatibility and reusability, it would at least be interesting to see how RAMI 4.0 can be taken into account in the modeling effort. A goal would be to have models that are RAMI 4.0 compatible, which only contain the information that necessary from the designer point of view.

2.4 State of the art for robotic safety and artificial intelligence

In robotics today, uses of AI range from support in perception tasks (e.g. detection and classification of specific objects by a single or group of sensors), as well as decision-making (e.g. deciding to carry out an action such as picking up a specific object, and planning that motion so that it is collision free).

In this section, we would like to discuss the state of the art for the use of artificial intelligence for safety-related components in HRC applications.

In Europe, all machines must fulfil the essential health and safety requirements as described in the Machinery Directive 2006/42/EC. This is therefore the starting point for achieving self-conformity as indicated by the CE mark, which is required by manufacturers of machines sold in the European Union. The directive also lists harmonized standards that offer supplemental information to the essential health and safety requirements. These standards are voluntary, and offer their users the presumption of conformity to the essential requirements (i.e. as a manufacturer of a machine, I know that my machine conforms to all health and safety requirements if I simply adhere to the harmonized standards listed in the MD/2006/42.). Applying

harmonized standards gives manufacturers more certainty due to the presumption of conformity that is afforded to them, and as such, this is often the method used by manufacturers of robotics applications. One of the most relevant Type-A harmonized standard for robotics is the EN ISO 12100, which specifies general principles for risk assessment and risk reduction. Here the concept of deterministic behavior plays a large role, as the risk assessment needs to be concrete and consider the specifications and operating parameters of the robot system. As an example, the payload, the geometry of the parts being carried, the environmental conditions around where the robot picks up, manipulates, and places parts all play a large role in the risk analysis. A robot carrying sharp objects represents a different risk than a robot carrying heavy parts, or small, blunt objects. Further parameters such as the program of the robot, which includes the operational envelope as well as the robot speeds, directly affect safety-related parameters such as the size of the minimum protective distance or the power and force applied to a person in case of a collision.

A further relevant harmonized standard is the Type B standard ISO 13849-1:2015 “Safety of machinery – safety related parts of control systems – Part 1: General principles for design”. Here requirements on the control system of safety related components such as sensors used to detect approaching humans are clearly defined. This includes requirements on safety-related embedded software, such as software verification, configuration management and functional tests (e.g. black-box or gray-box tests). This standard was written prior to the widespread use of machine learning techniques and places requirements on the software development process that currently cannot be fulfilled [77].

Finally, relevant Type-C standards for industrial robotics applications have already been identified. These include the ISO 10218-1 and -2, as well as the ISO/TS 15066, which specify how what should be validated by a measurement after the system has been physically set up and is in operation. The requirements on the validation of the safety functions, such as whether force and power limits for specific body parts have been respected, apply to a specific program. Such validation builds upon the concept of the robotic system being deterministic, where all relevant parameters and configurations are documented in the risk assessment. In summary, current methods for ensuring the safety of collaborative robotics build upon the concept of deterministic robot behavior that has been programmed by a human and validated to function properly under specific operational conditions. This is in diametrical opposition to the possibilities offered by artificial intelligence and machine learning for the tasks of perception and decision-making. The approach proposed in this thesis seeks to ameliorate this situation by providing a way to validate in simulation the safety of specific applications. One side effect of this approach is that the validation could also used for HRC applications that involve AI-defined programs.

2.5 Discussion on current methods for planning HRC applications

When reviewing the current process, the following issues stand out:

- Different software tools for analysis. The CAD / simulation tool is the main driver of the development process, but from a safety perspective, there are many unanswered questions. The arrows in Table 3 in step 5 and 8 indicate how the information in the documents is used to update the CAD/simulation models. Switching between different software tools (see overview in Table 3) means manually transferring data from one to the other, and leads to mistakes and inconsistent data about the system.
- The process is extremely iterative, due to the complexity and the options available to the designer. This means that there is no way of knowing if the

solution is the best, and comparing solutions means lots of different data sets that become unwieldy. Should the designer decide to use a combination of fences and scanners, or fences and light curtains, all calculations and assumptions made during previous design phases need to be revisited. This includes conferring with colleagues with different expertise (e.g. talking with sensor experts, safety experts). A change in one place can have grave effects from a safety standpoint.

- The methods for calculating the minimum required separation distance are limited. Current calculations are based on worst-case assumptions and do not reflect the reality of the system. Validation of the assumptions requires measurement, meaning that the hardware needs to be set-up before the designer can be sure that the design goals have been met. Furthermore, the assumptions can easily change, and interdependencies can be easily overlooked. Finally, the use of worst-case assumptions for the entire robot program, while easy to manage and understand, can lead to low robot performance and lower than necessary robot speeds.

Table 3: Overview of software associated with different steps of design phase

Step n°	Design phase	Responsible stakeholder	Software used
1	Starting point: General idea of collaborative application	Designer	Document, CAD/Simulation
2	Safety oriented design	Designer	Document (checklist)
3	General and essential requirements	Designer	Document (checklist), CAD/Simulation
4	Model process, assign tasks	Designer Safety Expert	Document/Spreadsheet
5	Define system limits and requirements	Designer	Document / Spreadsheet → CAD/Simulation
6	Hazard identification and risk evaluation	Designer Safety Expert	Document / Spreadsheet/ other safety evaluation tool CAD/Simulation
7	Hazard elimination and risk mitigation	Safety Expert	Document / Spreadsheet / other safety evaluation tool
8	Review	Designer	Document / Spreadsheet → CAD/Simulation

In the following section, we will present our methodology for designing HRC applications, with an aim towards addressing the current shortcomings and challenges.

Résumé Chapitre 3 – Spécification de la nouvelle approche

Cette section se concentre sur la spécification de la nouvelle approche pour aider Cette section se concentre sur la spécification de la nouvelle approche pour aider les concepteurs d'applications CRH à mieux comprendre et mettre en œuvre les aspects liés à la sécurité pendant les phases de conception et de développement du cycle de vie. Des travaux antérieurs ont utilisé des ontologies ou des modèles de système pour dériver les risques d'un système pour une analyse de risque [14] ou dans le but de réaliser une analyse des modes de défaillance et de leurs effets (FMEA) [17]. Cependant, aucun travail antérieur n'a appliqué les méthodes d'ingénierie des systèmes pour aider les concepteurs d'applications HRC pendant le déroulement des travaux spécifié dans le chapitre précédent.

Après le modèle en Vee de développement du système, les exigences des différents acteurs du système sont d'abord clairement formulées pour soutenir le développement de l'architecture globale. Les exigences les plus importantes identifiées sont les suivantes

- Suivi des exigences dans le processus de conception pour vérifier si les composants utilisés dans la conception répondent aux exigences spécifiées et explicitement formulées selon les normes de sécurité applicables à la robotique collaborative
- Soutien aux analyses de simulation couvrant tous les aspects d'une application HRC pour améliorer/optimiser les conceptions. Cela concerne à la fois le choix des composants et leur configuration, ainsi que les paramètres liés au processus et la disposition.
    - Indiquer la distance de protection minimale requise sur la base de ces paramètres et configurations spécifiques
- Soutenir l'intégration avec les outils d'ingénierie existants tels que les logiciels de CAD et de simulation afin que les informations numériques sur le système soient réunies en un seul endroit (par exemple pour soutenir la documentation, la mise en œuvre dans le monde réel),
- Permettre des résultats vérifiables et certifiables. À titre d'exemple, si le concepteur spécifie un capteur de sécurité et sa configuration dans l'outil d'ingénierie, et si cette configuration est ensuite utilisée dans le système d'application HRC mis en œuvre, alors une mesure de validation du système global ne devrait pas être nécessaire.

Suite à ces objectifs de haut niveau, des exigences plus concrètes sur l'approche ont également été formulées sur la base des besoins des différents ingénieurs impliqués dans le processus de conception. Les plus importantes d'entre elles sont :

- Montrer quelles mesures d'atténuation des risques sont possibles et valables
    - Cela signifie que le concepteur doit pouvoir choisir parmi un ensemble de capteurs de sécurité valables pour une utilisation dans des applications liées à la sécurité. Une façon de déterminer si un capteur spécifique est valable pour l'utilisation est de vérifier si le capteur a le niveau de performance requis selon la norme ISO 13849 ou le niveau d'intégrité de sécurité (SIL) requis selon la norme CEI 61508. Bien que cette vérification initiale puisse être considérée comme triviale, elle exige de l'utilisateur qu'il définisse un type de capteur réel et spécifique (fabricant, modèle), qui supporte les étapes suivantes de l'évaluation de la sécurité
- Calculer la distance de séparation minimale requise pour un capteur de sécurité spécifique et pour la configuration de système choisie et permettre

aux concepteurs de modifier les paramètres du processus (par exemple, la charge utile, le programme du robot, les vitesses) afin de déterminer leur influence sur la taille de la distance de séparation.
    - En ce qui concerne le mode de protection Surveillance de la vitesse et de la séparation (SSM), l'objectif est de maintenir une distance de séparation de protection par rapport au robot qui dépend du temps et qui ne doit pas être violée par l'homme pendant le fonctionnement du robot. Cette distance de séparation de protection est fonction de diverses propriétés du capteur et du robot, telles que la vitesse programmée du robot, la charge utile, le temps de réaction du capteur et la distance de freinage du robot pour la configuration donnée. Les modèles du robot et des capteurs doivent donc contenir toutes les informations et attributs nécessaires pour résoudre ces inégalités.
- Permettre aux concepteurs d'utiliser les paramètres réels du capteur et du robot (par exemple, la résolution du capteur, le temps de réaction du capteur, etc.) pour déterminer si les exigences de l'application spécifique ont été satisfaites.
    - Le concepteur doit donc être en mesure d'apporter des modifications à tous ces composants individuels pour voir comment le système global va réagir et en particulier pour comprendre les performances du système (par exemple, les temps de cycle réalisables, l'espace au sol requis, etc.). Les modèles physiques et orientés processus du système dans la CAD/simulation doivent donc être annexés pour inclure les informations relatives à la sécurité.
- Permettre aux concepteurs de valider que la position et la configuration des capteurs de sécurité sont valables.
    - Une question clé que les concepteurs d'applications robotiques doivent savoir est de savoir si les capteurs qu'ils choisissent sont positionnés correctement et sont capables de surveiller l'espace requis. En utilisant un exemple de scanner laser orienté parallèlement au sol pour surveiller un espace de travail autour d'un robot stationnaire, un concepteur voudra savoir où placer le capteur pour visualiser l'ensemble de la zone de sécurité requise, et si les paramètres du capteur configurés (par exemple, la portée, la résolution angulaire) sont valides. En supposant que la distance de sécurité minimale requise autour de l'ensemble de la trajectoire du robot pour un cycle de travail complet est connue et peut être affichée au concepteur, le champ de vision du scanner laser peut être superposé pour une comparaison. Le concepteur peut alors déterminer visuellement si les paramètres spécifiques du scanner laser, notamment la portée, l'angle de balayage (début et fin) et le champ de vision global, sont suffisants pour l'application. Une autre fonction utile serait la possibilité de vérifier si des objets spécifiques à l'environnement, tels que des montages, des tables ou des colonnes, se trouvent dans le champ de vision du capteur et nécessitent une attention particulière.

Une architecture est proposée qui utilise les logiciels de CAD/simulation actuellement utilisés par les concepteurs comme principal point d'accès pour les enquêtes de sécurité. Les modèles des capteurs et des robots de sécurité déjà utilisés dans l'outil de CAD/simulation seront complétés par une architecture complémentaire

pour permettre le calcul des éléments requis formulés précédemment. Cela présente l'avantage de simplifier le processus du point de vue du concepteur en ne l'obligeant pas à apprendre de nouveaux outils ou à trop modifier son processus global.

La méthodologie de l'ingénierie des systèmes a été appliquée à un certain nombre d'industries et est actuellement appliquée pour des questions spécifiques en robotique, plus particulièrement pour le génie logiciel et pour soutenir l'identification des dangers, les analyses HAZOP et l'injection de défauts. Cependant, elle n'a pas encore été appliquée pour soutenir les concepteurs d'applications HRC pendant les phases de conception, d'étude et de vérification du cycle de vie du système. Les efforts précédents dans le domaine du développement d'ontologies pour les capteurs et la robotique n'ont pas pris en compte les informations liées à la sécurité dans leurs modèles sémantiques. Cette thèse propose d'ajouter des modèles de capteurs existants avec des informations liées à la sécurité pour analyser les applications HRC.

Les processus et les flux de travail actuels pour la conception d'applications HRC peuvent être caractérisés comme étant fragmentés, dans lesquels des experts de différents horizons (mécanique, électrique, contrôles, programmation de logiciels, sécurité) travaillent avec différents outils logiciels et sans moyens clairs pour l'échange d'informations numériques. Des estimations simplifiées sur des aspects spécifiques de l'application HRC (par exemple, des hypothèses sur les vitesses du robot) conduisent souvent à des calculs du pire cas qui ne sont pas nécessairement représentatifs des performances du système final. Le processus est itératif, souvent avec des paramètres opérationnels clés spécifiques qui ne sont pas connus avant la validation finale avec le système réel. Cette situation oblige les concepteurs et les planificateurs à faire un bond en avant pour croire que le système final aura, d'une manière ou d'une autre, de meilleures caractéristiques de performance que celles prévues à l'origine ou que de nouvelles applications HRC ne sont pas mises en œuvre en raison des méthodes de planification prudentes et du pire cas qui sont actuellement pratiquées.

En résumé, ces travaux apportent de nouvelles contributions à l'état de l'art :

- La modélisation des normes ISO10218-1, -2, et ISO-TS15066 dans des modèles d'exigences afin que la conformité d'une conception puisse être vérifiée dès le début de la phase de conception
- Utiliser des modèles de systèmes et de composants qui décrivent non seulement les principaux composants du modèle, mais aussi les aspects liés à la sécurité pour des capteurs de sécurité spécifiques
- Développer des modèles compatibles avec les outils de CAD/simulation existants afin de simplifier le flux de travail du concepteur, de réduire la recherche d'informations (puisque les modèles contiennent une grande partie des informations requises) et de concentrer les informations numériques sur la conception en un seul endroit plutôt que sur différentes plateformes et documents. Cela présente l'avantage supplémentaire de soutenir les ingénieurs également pendant le processus de CE (figure 4) en simplifiant le processus de validation.
- Soutien aux concepteurs pendant leur flux de travail normal, en particulier pour:
    - Calcul de la distance de protection minimale pour les applications HRC comportant des SSM
    - Validation des mesures d'atténuation des risques
    - Validation de la position du capteur dans l'espace de travail
    - Exécution d'analyses de simulation portant sur tous les aspects de l'application HRC (par exemple, robot, outils, pièces, capteurs de sécurité, environnement, processus)

3 Specification of the new approach

Systems engineering methodology has been applied to a number of industries and to an extent is currently being applied for specific questions in robotics, most notably for software engineering [46]. In the literature, there have been instances where either ontologies or system models have been used to derive the risks of a system for a risk analysis [14] or for the purpose of carrying out a Failure Mode and Effect Analysis (FMEA) [17]. However, there has been no previous work which applies systems engineering methods towards supporting designers of HRC applications during the work flow specified in Figure 3.

The work in this thesis towards developing a new method for planning HRC applications adheres to the Vee model [79] of development as shown in Figure 7. The key stakeholders in the design of HRC applications were identified and described in Section 2.2. The requirements on the new method as well as the architecture will be specified in the following section. The models developed to support the new design approach are be described in Section 4. Finally, Section 6 serves as a validation of the approach (i.e. validation of the architecture and the developed models). In this case, the validation is achieved through a comparison against compare the results of the approach, two exemplary use-cases will be designed according to the traditional methods.

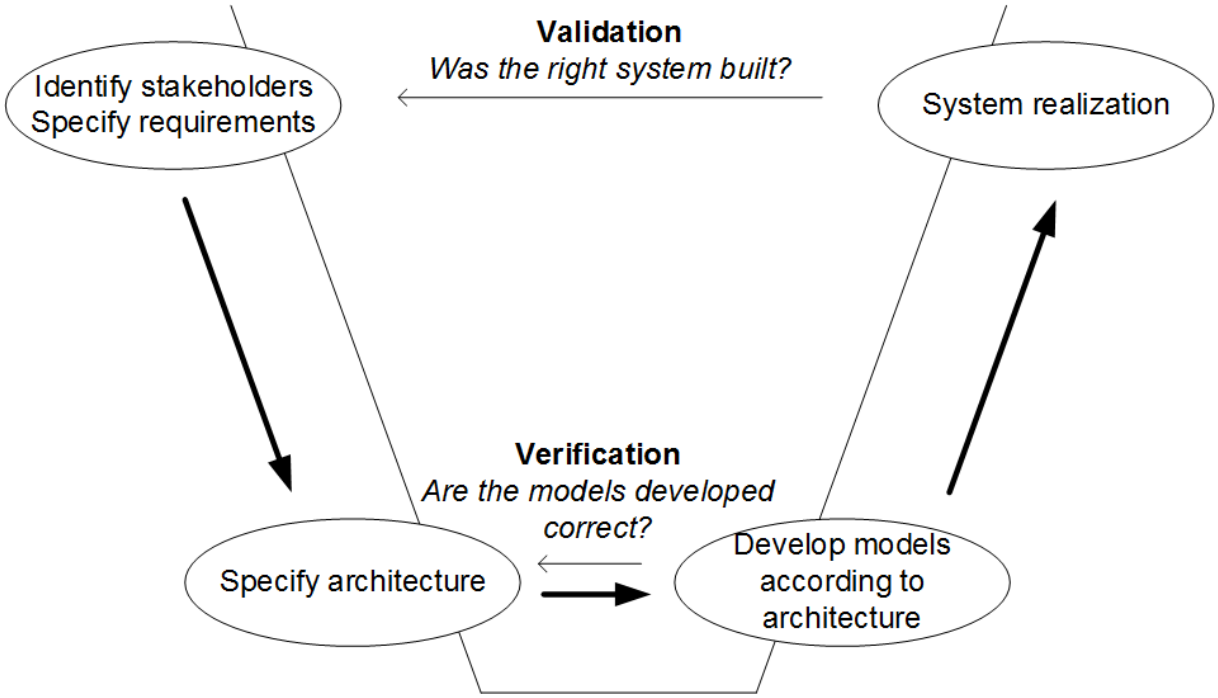


Figure 7: Schematic overview of the structure of the work in this thesis in adherence to the Vee model

3.1 Requirements specification

Following the identification of the stakeholders involved in the design of HRC applications in Section 2.2, the next step according to the Vee development model is to clearly define the requirements on the system. In this case, the system is a method and tool to support the safety analysis when designing and planning HRC applications. Analysis of the current methods for designing industrial applications featuring collaborative robots from Section , a number of high-level requirements for an engineering tool and/or approach to streamline the design process have been formulated:

- Requirements tracking in the design process to check whether the components used in the design fulfill the specified and explicitly formulated requirements according to the standards ISO 10218-1, -2 and ISO-TS 15066
- Support for what-if analyses covering all aspects of an HRC application to improve/ optimize designs. This refers both to component choice and to their configuration, as well as process-related parameters and the layout.
    - Show the required minimum protective distance based on these specific parameters and configurations
- Support integration with existing engineering tools such as CAD and simulation software so that digital information about the system is in one place (e.g. to support documentation, real world implementation),
- Allow for verifiable and certifiable results. As an example, if the designer specifies a safety sensor and its configuration in the engineering tool, and if this configuration is then used in the implemented HRC application system, then a validation measurement of the overall system should not be necessary. This means that the models and the calculations need to be based on the standards and regulations.

Figure 8 highlights how the final engineering tool should connect with CAD/simulation during concept and development phases and to the real implementation during the production, utilization and support phases. This is a powerful concept that places further sub-requirements on access to digital data from the planning phase during the later phases of real world implementation.

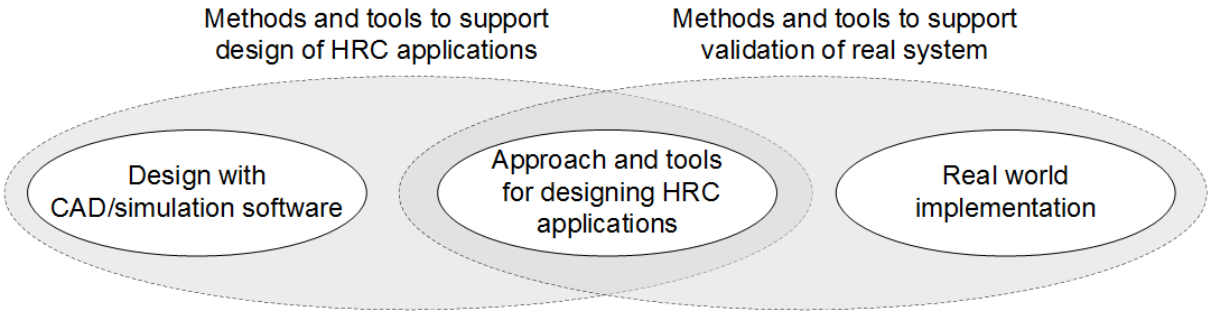


Figure 8: Schematic drawing showing connection between proposed engineering models, CAD/CAE for design, and real world implementations

The rationale behind this requirement is to overcome the current situation in planning as described in Figure 2. The requirement relating to verifiable and certifiable results will lead to the improvement as shown in the bottom of Figure 9, namely eliminating the need for replanning an HRC application after it has been built.

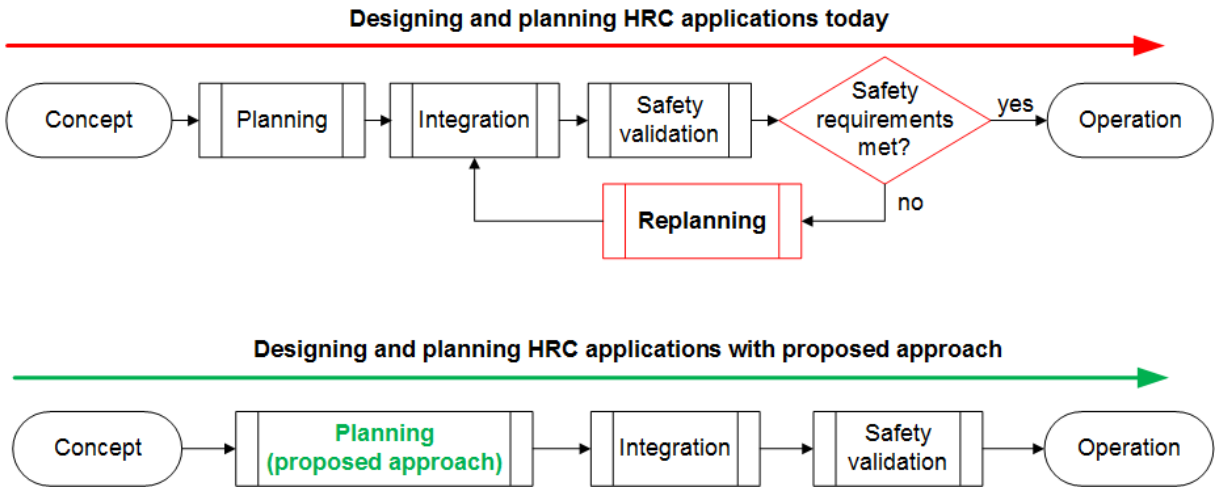


Figure 9: Effect of proposed design and planning approach on workflow for HRC applications, avoiding iterations after the system has been integrated and built

Looking more into detail at the individual planning processes from Section 2.2, the key system characteristics that designers should be able to investigate with the proposed approach are:

- What risk mitigation measures are possible and valid?
- What is the required minimum separation distance for a specific safety sensor and for the chosen system configuration? How do process parameters (e.g. payload, robot program, speeds) that the designer can change influence their size?
- Do all chosen sensor and robot parameters (e.g. sensor resolution, sensor reaction time, etc.) fulfill the requirements of the specific application?
- Is the configuration and position of my sensor valid to monitor the required safety zone?

These requirements will be described individually in the following sections, highlighting any further constraints they place on the proposed approach.

3.1.1 Validity of risk mitigation measures

As a first step, the designer should to be able to choose from a pool of safety sensors that are valid for use in safety-related applications. The simplest means to determine whether a specific sensor is valid for use is to check whether the sensor has the required Performance Level according to ISO 13849 or the required Safety Integrity Level (SIL) according to IEC 61508. While this initial check may be considered trivial, it requires the user to define a real, specific sensor type (manufacturer, model), which is relevant for further evaluation.

3.1.2 Definition of required minimum separation distance

When considering the safeguarding mode Speed and Separation Monitoring (SSM), the goal is to maintain a time-dependent protective separation distance $S_p(t)$ to the robot that must not violated by the human during robot operation. If the current distance between robot and human is S, the following constraint must be fulfilled for all times t

$$S(t) \geq S_{p,i}(t) \quad . \tag{3.1}$$

For the special case the robot does not move $|\hat{v}_r(t^*)| = 0$ the separation distance is zero $S_{p,i}(t^*) = 0$. Otherwise $S_{p,i}(t)$ depends on the velocity of the robot, its braking behavior and a couple of time-invariant parameters mostly related to the installed safety sensors that measure $S(t)$. In accordance to ISO/TS 15066, the two cases give

$$S_{p,i}(t) = \begin{cases} 0 & |\hat{v}_r(t)| = 0 \\ S_h(t) + S_{r,i}(t) + S_{s,i}(t) + C + Z_d + Z_r & \text{otherwise} \end{cases} . \quad (3.2)$$

Note that $S_{p,i}(t)$ represents the minimum separation distance for particle i on the robot surface, which theoretically consists of an infinite number of particles. Following this concept, $\hat{v}_r(t)$ is the velocity of the fastest moving particle i given by $\hat{v}_r(t) = \max\{v_{ri}(t)\}$. The contribution of the moving human to $S_p(t)$ is $S_h(t)$ and is calculated as follows

$$S_h(t) = \int_t^{t+T_{rs}} v_h(\tau) d\tau \quad . \quad (3.3)$$

It depends on the velocity of the human at time t, which is $v_h(t)$, during the stopping duration T_{rs} that is the time the robot needs from detecting the violation of (3.1) to stopping all joints to velocity zero. The total duration is $T_{rs} = T_r + T_s$, including both the reaction time T_r and the stopping time T_s.

In the absence of a system to measure the speed and direction of human motion, we can use a constant value of 1600 mm/s for approach speed of the human. In this case, the separation distance attributable to the speed of the approaching human, can be calculated as follows:

$$S_h = 1600\ mm\ \times (T_r + T_s) \quad (3.4)$$

The reaction time T_r is defined as the delay from detecting the violation of (3.1) to shortly before initiating the brakes. Accordingly, the duration from this particular moment to the one at which all axes reach velocity zero is the braking time T_s. The contribution of the robot movement to $S_{p,i}(t)$ is divided into $S_{r,i}(t)$ and $S_{s,i}(t)$, which represent the displacement of the robot particles during the reaction time T_r and T_S respectively

$$S_{r,i}(t) + S_{s,i}(t) = \int_t^{t+T_r} v_{r,i}(\tau) d\tau + \int_{t+T_r}^{t+T_r+T_s} v_{r,i}(\tau) d\tau = \int_t^{t+T_r+T_s} v_{r,i}(\tau) d\tau \quad . \quad (3.5)$$

Another contribution of the robot included in (3.2) is Z_r, which is the position uncertainty of the robot. The influence of the safety sensor is expressed by its non-detectable overreaching distance C and its minimal spatial resolution Z_d also denoted as the position uncertainty of a person in the collaborative workspace.

The models of the robot and sensors should therefore contain all the information and attributes necessary to solve these inequalities.

3.1.3 Validity of sensor positioning in environment

A key question that designers of robotics applications need to know is whether the sensors they choose are positioned correctly and are able to monitor the space required. Using an example of a laser scanner oriented parallel to the ground to monitor a workspace around a stationary robot, a designer will want to know where to place the sensor to view the entire required safety zone, and whether the configured sensor settings (e.g. range, angular resolution) are valid. Assuming that the minimum required

safety distance around the entire robot's trajectory for a complete working cycle is known and can be displayed to the designer, the laser scanner's field of view can be overlaid for a comparison. The designer can then visually determine whether the specific parameters of the laser scanner including the range, the scanning angle (start and finish), and the overall field of view are sufficient for the application. In Figure 10 the yellow surface represents the necessary size of the minimum protective distance for the entire robot program. In this example, the designer can see that the chosen range of the sensors is insufficient and a longer range is necessary.

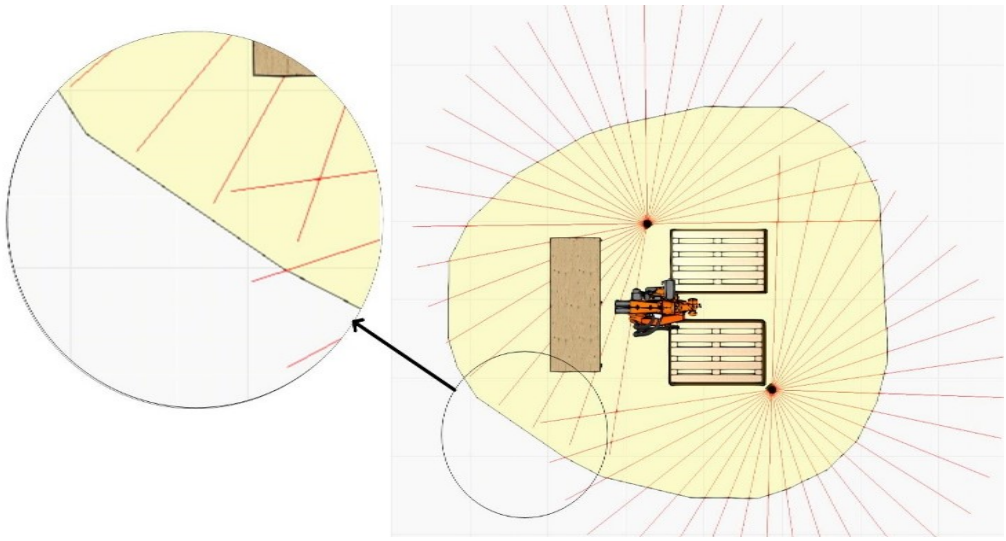


Figure 10: Layout with robot, two pallets, and a table. The red lines represent the range and angle of view of the two laser scanners placed in the scene. The yellow polygon around the robot represents the required separation distance over the entire robot program. The designer should be able to visually check that the sensor configuration and position is valid.

Another useful feature would be the ability to check whether environment specific objects such as fixtures, tables, or columns are in the field of view of the sensor and require special attention.

3.1.4 Changing process parameters to meet specific design targets

The advantage of considering the safety aspects of collaborative robotics during the design phase is the ability to optimize the parameters of the entire system to meet specific design targets before building the system. Different options and tradeoffs can be made visible and the design team can discuss these possibilities with management prior to larger investments. A systems engineering approach to the challenge of designing HRC applications necessarily looks at all elements of the system, from the components (robot, gripper, safety sensors, etc.), the process (robot speeds, work pieces, tooling, etc.), the environment, and human factors. The designer should therefore be able to make changes to all of these individual components to see how the overall system will react and in particular to understand the system performance (e.g. achievable cycle times, required floor space, etc.). The physical and process-oriented models of the system within the CAD/simulation should therefore be appended to include safety-related information.

3.2 Architecture specification

In order to reach the goals specified in the previous section, a simple architecture (Figure 11) that allows for data exchange with a CAD/simulation software is proposed.

The CAS Tool uses a solver to evaluate safety-related questions and display this information in the CAD/simulation environment. A library of components is integrated in the CAS Tool to represent all the components in the safety considerations described in the previous sections.. The data population of the individual models is an important factor to consider in the implementation of the CAS Tool. The architecture allows the following means for entering and updating model data:

- importing data from manufacturer e-Data Sheets [82],
- allowing the user via a dedicated frame in the CAD/simulation software to input the required data, or
- direct import of the the physical (static and dynamic) models that already exist in the CAD/simulation software.

A key feature of this architecture is that it builds on software tools that exist today that designers are already using and familiar with. Another key feature is that the CAS tool can be used with different CAD/simulation software available on the market today. It is not embedded into a single system, but meant to be an add-on or plug-in.

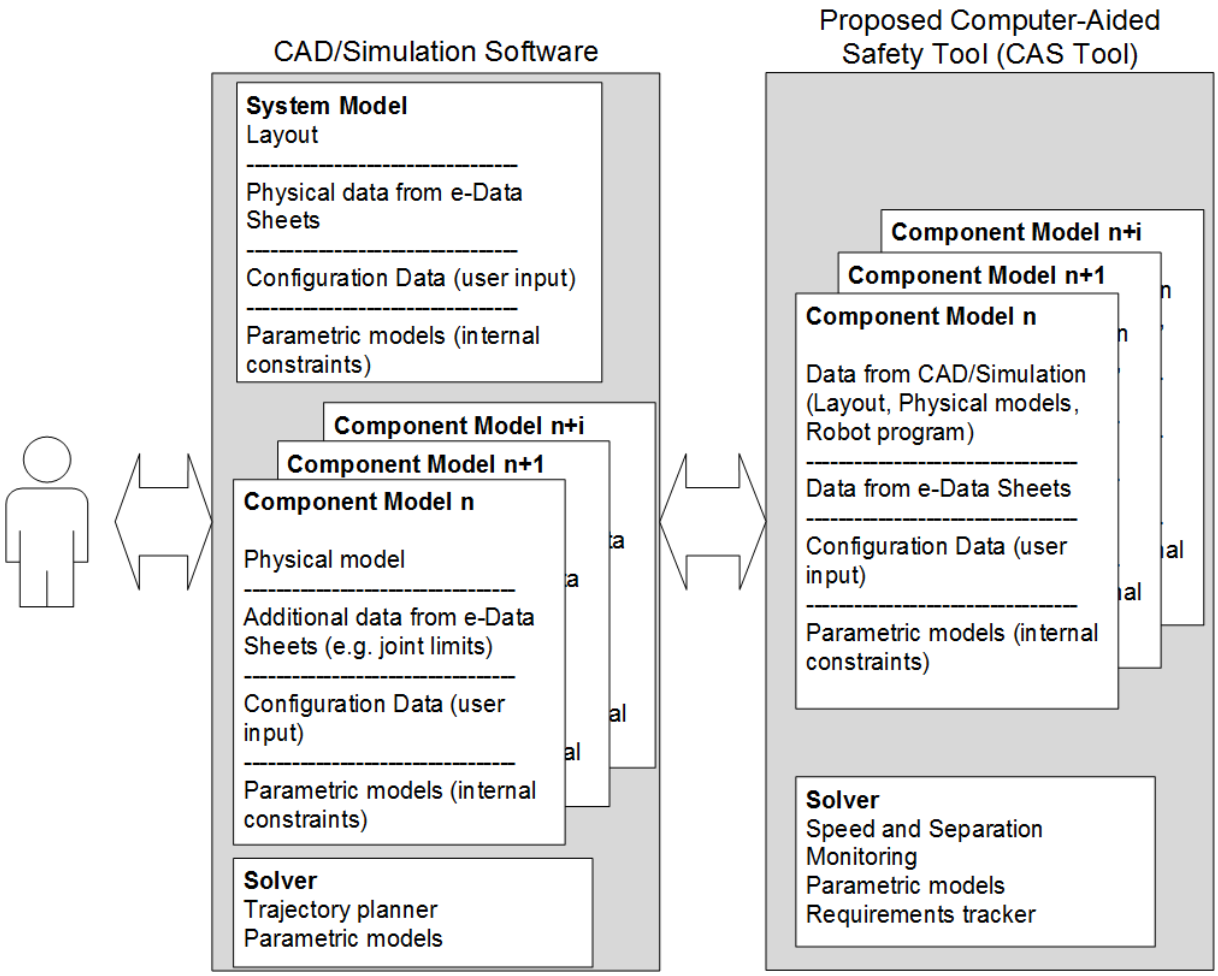


Figure 11: Proposed architecture for approach with user interacting with CAD/simulation software and the CAS Tool in the background

The sequence diagram in Figure 12 shows the workflow, the different software involved and the flow of information when planning an HRC application with the proposed tools. An important aspect is that the designer only accesses the CAS Tool through the CAD/simulation software. They do not need to learn new software tools, and the proposed approach does not seek to eliminate what is already available.

Indeed, a key aspect from the architectural standpoint is, based on the normal and well-established workflow as described in Section 2.1, finding a way to build on existing tools and reduce situations where digital information is lost.

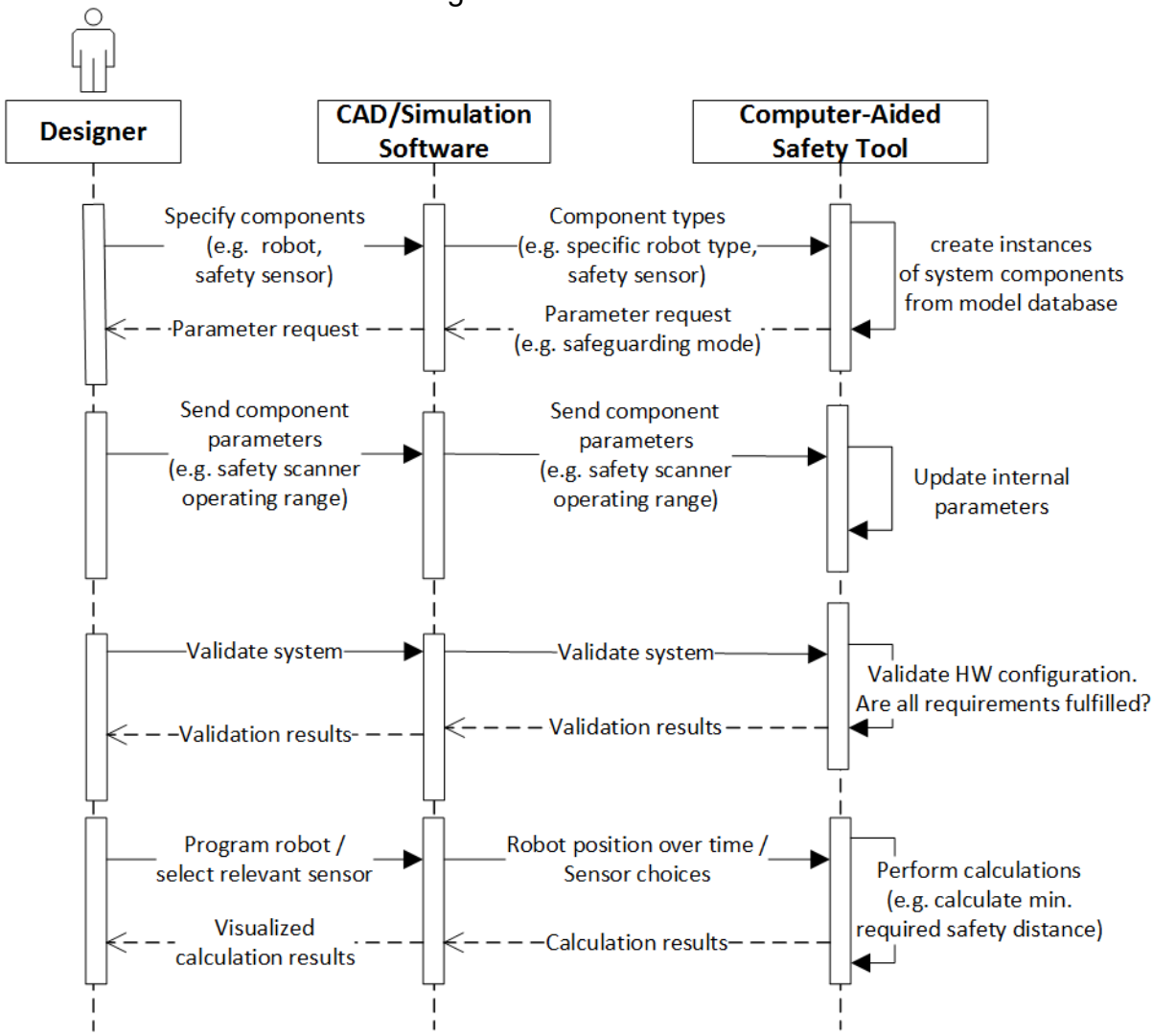


Figure 12: Sequence diagram of the workflow of the designer using existing CAD/Simulation software tools and the proposed Computer-Aided Safety tool

The Computer-Aided Safety Tool proposed here therefore collects all relevant information needed for the safety-related evaluation and dimensioning of components initially from the CAD/Simulation tool. This includes model information about the robot types, their position in the layout, the physical parameters, and importantly information about the process through the program and the physical information (e.g. tooling and part information). It then appends these models with safety-related information that is available through a number of sources, but remains firmly in the background from the designer perspective. This sensor model data is kept in an internal database within the Computer-Aided Safety Tool for later use.

A key aspect of the approach is the development of the sensor models to be used in the CAS Tool to fulfil the requirements defined in the previous section.

3.3 Novelty of approach

Systems engineering methodology has been applied to a number of industries and is currently being applied for specific questions in robotics, most notably for software engineering and for supporting identification of hazards, HAZOP, and Fault-Injection

analyses. However, it has not yet been applied to supporting HRC application designers during the concept, design, and verification phases of the system lifecycle. Previous efforts in the field of ontology development for sensors and robotics have not considered safety-related information in their semantic models. This thesis proposes appending existing sensor models with safety-related information to analyze HRC applications.

The current processes and workflows for designing HRC applications can be characterized as fragmented, whereby experts from different backgrounds (mechanical, electrical, controls, software programming, safety) work with different software tools and without clear means for exchange of digital information. Simplified estimates about specifics of the HRC application (e.g. assumptions about the robot speeds) often lead to worst-case calculations that are not necessarily representative of the final system performance. The process is iterative, often with specific key operational parameters not known until the final validation with the real system. This situation requires designers and planners to make a leap of faith that the final system will somehow have better performance characteristics than originally planned or new HRC applications are not implemented due to the conservative, worst-case planning methods currently practiced.

In summary, the approach proposed in this thesis of applying systems engineering modeling techniques to the concept, design, and verification of HRC applications represents a novel contribution to the state of the art. This includes:

- Modeling of the standards ISO10218-1, -2, and ISO-TS15066 in requirements models so that the conformity of a design can be verified early in the design phase
- Using system and component models which not only physically describe the main model components but also the aspects related to safety for specific safety sensors
- Developing models that are compatible with existing CAD/simulation tools so that the workflow for the designer is simplified, the search for information is reduced (since the models contain much of the required information) and the digital information about the design is more concentrated in one place rather than across different platforms and documents. This has the added advantage of supporting the engineers also during the CE process (Figure 4) by simplifying the validation process.
- Support for designer within the workflow identified in Figure 3, specifically for:
    - Calculation of minimum protection distance for HRC applications featuring SSM
    - Validation of risk mitigation measures
    - Validation of sensor position in the workspace
    - Execution of what-if analyses looking at all aspects of the HRC application (e.g. robot, tools, parts, safety sensors, environment, process)

Résumé Chapitre 4 – Développement de modèles pour l'analyse de la surveillance de la vitesse et de la séparation

Cette section se concentre sur le développement des modèles de capteurs pour soutenir l'analyse des applications HRC comportant la surveillance de la vitesse et de la séparation au sein de l'architecture spécifiée dans le chapitre précédent. Les exigences et la spécification de l'architecture du chapitre précédent sont utilisées pour définir la portée des modèles. Les modèles développés ici se limitent à l'utilisation prévue de l'aide à l'analyse de sûreté pendant le processus de conception.

Dans un premier temps, l'utilisation du SysML pour la modélisation des normes pertinentes en tant que différents types d'exigences est décrite. Un défi spécifique à la modélisation des normes est leur forme actuelle, qui n'est pas lisible par une machine et qui contient de nombreuses clauses et exceptions concernant le moment où des situations spécifiques s'appliquent. L'utilisation de hiérarchies et de stéréotypes spécifiques au SysML contribue à simplifier la tâche de modélisation. En particulier, les quatre stéréotypes SysML <<performanceRequirement>>, <<physicalRequirement>>, <<designConstraint>> et <<functionalRequirement>> ont été utilisés pour des clauses spécifiques de la norme. Une <<performanceRequirement>> est satisfaite par une propriété de valeur, et elle mesure quantitativement la mesure dans laquelle un système ou une partie de système satisfait à une capacité ou à une condition requise. Le stéréotype <<physicalRequirement>> est satisfait par un élément structurel, et il spécifie les caractéristiques physiques et/ou les contraintes physiques du système ou d'une partie du système. Le stéréotype <<designConstraint>> est satisfait par un bloc ou une partie, et il spécifie une contrainte sur la mise en œuvre du système ou d'une partie du système, telle que "le système doit utiliser un composant commercial du commerce". Une <<functionalRequirement>> est satisfaite par une opération ou un comportement qu'un système ou une partie d'un système doit effectuer. Cette section fournit quelques exemples de modélisation d'une norme en tant qu'exigence, et montre comment cela peut soutenir le processus de conception.

L'étape suivante de l'effort de modélisation a consisté à spécifier les modèles de capteurs. Les ontologies existantes pour la description des systèmes robotiques et des capteurs ont été examinées dans le but de s'assurer que les modèles décrits pour les capteurs sont compatibles avec ces travaux antérieurs. Dans cette situation, la compatibilité signifie que les types de données spécifiques déjà utilisés ou établis seront respectés. Le point de départ a été l'utilisation de types et d'unités de grandeurs selon les définitions du SI de la norme ISO 80000-1. Ces ontologies existantes étaient utiles pour décrire les caractéristiques mécaniques comme les dimensions et le poids, et les caractéristiques liées aux capteurs comme la plage de fonctionnement, mais elles ne contenaient pas les informations nécessaires pour atteindre les objectifs de conception définis dans le chapitre précédent.

L'effort de modélisation des capteurs s'est concentré sur l'identification des attributs nécessaires aux analyses de sécurité pour les capteurs les plus souvent utilisés dans des applications de surveillance de la vitesse et de la séparation. Ces modèles contiennent à la fois les attributs des composants et les informations paramétriques. Les informations paramétriques se présentent sous la forme d'équations qui décrivent comment des attributs spécifiques des capteurs jouent un rôle dans le calcul d'autres attributs des capteurs ou comment ils contribuent à la distance minimale de séparation requise. Les quatre types de capteurs étudiés sont les suivants :

- Scanner laser
- Rideau photoélectrique

- Tapis de protection
- Système de caméra par projection

Pour chaque type de capteur, un modèle de capteur générique a été dérivé. Les modèles de capteurs génériques étendent les définitions ontologiques existantes avec l'ensemble le plus élémentaire d'attributs nécessaires pour comprendre comment le capteur contribue à la taille de la zone de protection minimale. Les modèles spécifiques de capteurs (par exemple, un modèle spécifique d'un fabricant, comme le scanner laser de sécurité SICK microScan3) sont également analysés pour déterminer s'il existe une différence entre le cas générique et le cas spécifique.

Dans le cas d'un scanner laser générique, les attributs suivants se sont avérés les plus pertinents pour déterminer la taille de la distance de séparation minimale requise:

- temps de réaction du capteur, T_r,
- résolution du capteur, d_{LS},
- hauteur du capteur, H_{LS},
- l'orientation du capteur, θ_{LS}
- incertitude du capteur, Z_r

Deux scanners laser de sécurité spécifiques de deux fabricants différents ont également été modélisés pour mieux comprendre les relations paramétriques entre les attributs des différents capteurs.

Dans le cas du scanner laser de sécurité SICK microscan3, il existe des relations paramétriques entre la durée du cycle, la protection contre les interférences, le multi-échantillonnage et le temps de traitement des signaux d'entrée et de sortie, ainsi que des relations entre la durée du cycle de balayage, la portée et la résolution du capteur. Par conséquent, tous ces attributs et leurs informations paramétriques font nécessairement partie du modèle de capteur spécifique.

Dans le cas du scanner laser de sécurité Hokuyo UAM-05LP, le principe de fonctionnement du capteur et le paramétrage étaient très différents de ceux du scanner SICK. Dans ce cas, le temps de réponse minimum du capteur peut être choisi entre 60 ms et 270 ms, la robustesse contre les mesures faussement positives étant directement proportionnelle à la taille du temps de cycle (par exemple, un temps de réponse élevé est plus robuste contre les signaux faussement positifs). En outre, quatre autres attributs affectent également le temps de réaction, à savoir:

- Commutation de zone lorsque la zone de sécurité du capteur configurée est commutée par un système de contrôle, q_{switch}
- Muting activée/désactivée, q_{mute}
- Saut de balayage, qui peut être défini sur l'une des valeurs entières [0, 1, 2 ou 3] et qui peut être utilisé pour réduire les interférences avec l'équipement environnant en sautant un balayage pendant un nombre de cycles spécifié, s_{skip}
- Fonctionnalité maître/esclave [activée/désactivée], q_{ms}

L'influence de ces attributs sur le temps de réaction est déterminée par des tables de recherche. De plus, l'incertitude du capteur montre une dépendance à la réflectivité de l'environnement.

Dans le cas d'un rideau lumineux générique, les attributs les plus importants sont la résolution du rideau lumineux, d_{LC}, le temps de réaction du capteur, T_r, et l'incertitude du capteur, Z_r.

Si l'on considère le modèle de barrière immatérielle de sécurité C4000 de SICK, la liste des paramètres de configuration et des attributs des capteurs pertinents s'allonge considérablement par rapport au cas générique. Ces attributs et leurs relations paramétriques jouent un rôle important dans la détermination du comportement réel du capteur. Ils garantissent que toute configuration choisie est physiquement possible et valable pour la situation. Le temps de réponse de la barrière immatérielle de sécurité est défini en fonction des variables suivantes:

- résolution physique, d_{LC}
- hauteur du champ de protection, H_{LC}
- largeur du champ de protection, W_{LC}
- nombre de faisceaux utilisés, B_{LC}
- codage de faisceau, C_{LC}
- temps de réponse, T_{rt}
- configuration de la suppression flottante, F_{LC}
- nombre de systèmes en cascade, M_{LC}
- le dispositif de commutation du signal de sortie utilisé, O_{LC}
- nombre de rideaux lumineux utilisés, G_{LC}
- temps de réponse des barrières immatérielles utilisées, $T_{rt\_guest\_n}$

Un tapis de protection sensible à la pression utilisé pour le SSM a les mêmes propriétés générales que les autres capteurs, à savoir la valeur C et le temps de réaction, T_r. Parmi les autres attributs, on peut citer:

- hauteur d'une marche, sur laquelle le capteur est monté, h_{step}
- temps de réponse Tapis de protection, T_{rt_mat}
- temps de réponse contrôleur de sécurité, T_{rt_sc}
- largeur du tapis de protection, W_{mat}
- longueur du tapis de protection, L_{mat}.

Le dernier type de capteur étudié est le système de surveillance de l'espace de travail basé sur la projection, breveté et développé par le Fraunhofer IFF. Le système consiste en une combinaison d'un projecteur et de caméras, et est capable d'établir des zones de sécurité de forme, de taille et de position arbitraires dans un espace de travail collaboratif. Le capteur est spécialement conçu pour être utilisé avec le SSM et peut prendre en compte le robot, son outillage et toute pièce manipulée dans le calcul de la taille de la zone de sécurité requise. Dans ce cas, les attributs du capteur liés à la taille de la distance de séparation minimale requise ainsi qu'au champ de vision dans l'espace de travail sont importants pour la modélisation du capteur. Les attributs les plus pertinents sont les suivants:

- position du capteur dans la direction x, x_p
- position du capteur dans la direction y, y_p
- position du capteur dans la direction z, z_p
- angle du capteur autour de l'axe des x, α_p
- angle du capteur autour de l'axe des y, β_p
- angle du capteur autour de l'axe des z, γ_p
- hauteur de la pyramide de la lumière projetée (de la base au sommet), H_p

- largeur de la base de la pyramide de la lumière projetée, W_p
- longueur de la base de la pyramide de la lumière projetée, L_p
- angle du capteur par rapport au plan du sol, θ_{LS}
- temps de réaction, T_r
- résolution, d_{LS}
- incertitude du capteur, Z_r

En résumé, quatre modèles génériques ont été développés pour un scanner laser, un rideau lumineux, un tapis de protection et un système de surveillance de l'espace de travail par projection. En outre, trois modèles de capteurs spécifiques ont été dérivés pour illustrer le processus et montrer comment les modèles peuvent varier en fonction du modèle et du fabricant, même pour le même type de capteur générique. Si les modèles génériques sont suffisants pour une première analyse de l'application, les modèles spécifiques aux fabricants sont préférables pour éviter les configurations non valables et simplifier la mise en service du système.

4 Model development for speed and separation monitoring analysis

Building on the system architecture specified in the previous section, this section will describe the modeling efforts for sensors that can be used for SSM. Following a brief description of the general methodology applied to the model development individual sensor models will be derived and described. As previously mentioned in Section 2.3.4, SysML software was used to develop the sensor model. This serves both as documentation and to ensure adherence to systems engineering standards and methodologies.

4.1 General modeling background for speed and separation monitoring

In general, the modeling efforts were guided by the quality criteria as presented in [63]. The purpose of the models is clearly defined by the requirements and architecture specification in Section 3. The models developed here are limited in scope to focus on the intended use of assisting the safety analysis during various phases of the design process.

The starting point of the modeling process was to investigate the relevant standards that relate to the design of HRC applications and how to incorporate them into this work. The requirements as specified in the ISO TS 15066, the ISO 13855 are the most relevant for example for determining the size of minimum protective distances when safeguarding via Speed and Separation Monitoring or Safety-Rated Monitored Stop. The ISO 10218-1 and -2 also place relevant requirements on the HRC application.

A particular challenge when modeling these standards was their structure, which are in regular text, do not follow a machine-readable structure, and contain clauses and exceptions specifying under what conditions they apply. When working with a large number of requirements, it can be useful to add additional categorization. Table 4 shows an overview of different types of requirements used in SysML [63] that were applied to the standards in this work.

Table 4. Types of requirements used in SysML, excepted from [63] A Practical Guide to SysML: The Systems Modeling Language.

Stereotype	Constraints	Description
<<performanceRequirement>>	Satisfied by a value property	Requirement that quantitatively measures the extent to which a system or a system part satisfies a required capability or condition
<<physicalRequirement>>	Satisfied by a structural element	Requirement that specifies physical characteristics and/or physical constraints of the system or a system part
<<designConstraint>>	Satisfied by a block or part	Requirement that specifies a constraint on the implementation of the system or system part, such as “the system must use a commercial off-the-shelf component”.
<<functionalRequirement>>	Satisfied by an operation or behavior	Requirement that specifies and operation or behavior that a system or part of a system must perform

The stereotype <<performanceRequirement>>, which can be satisfied by a value property, was used for example to model clauses related to specific attributes of a component, such as the requirement that a safety sensor have Performance Level d, Category 3 according to ISO 13849. The sensor models in turn will need an attribute titled “Performance Level”, which can have one of the five discreet values, “a, b, c, d, or e” as described in the ISO 13849. An instance of a safety sensor will therefore have a value property for “Performance Level” that is derived from its data sheet. This value

for a specifically used instance can be checked against the requirement from the ISO TS 15066 to show the designer that the system is in conformity with specific standard clauses. Another requirement stereotype that will be featured in the model are <<physical Requirement>>, which are satisfied by a structural element. A simple example here could be that for a specific configuration, it is required that an Emergency Stop switch be present. Alone the presence of an emergency stop switch in the component configuration would then be enough to satisfy this requirement.

The standards have hierarchies built into the text. This is not only reflected in the sections and sub-sections, but also within the text itself, like in the example of the clause in Figure 13. In order to simplify the task of ensuring that requirements are satisfied with model elements, the requirements were organized to reflect these hierarchies. This is meant to reduce the overall number of high-level requirements to a more manageable level.

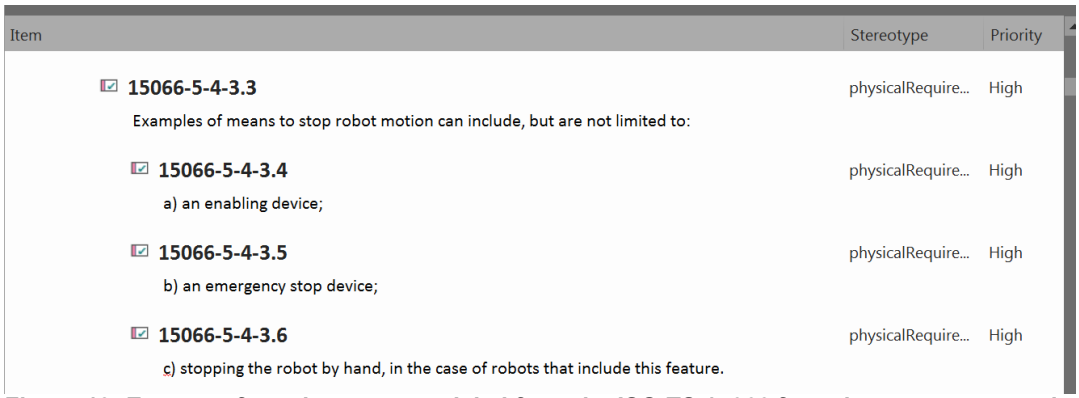


Figure 13: Excerpt of requirements modeled from the ISO-TS 15066 featuring stereotype and priority

Following the modeling of the standards as requirements, the specification of the sensor models was carried out. The focus of this modeling effort was identification of the attributes needed for the safety-related analyses as specified in the previous section. As identified in Section 2, there exist ontologies for describing robotic systems and sensors. These were reviewed with the goal of ensuring that the models described for the sensors are compatible with the previous work. In this situation, compatibility means that specific data types that are already in use or are established will be respected. The starting point was using quantity kinds and units according to SI definitions from ISO 80000-1 (Figure 14).

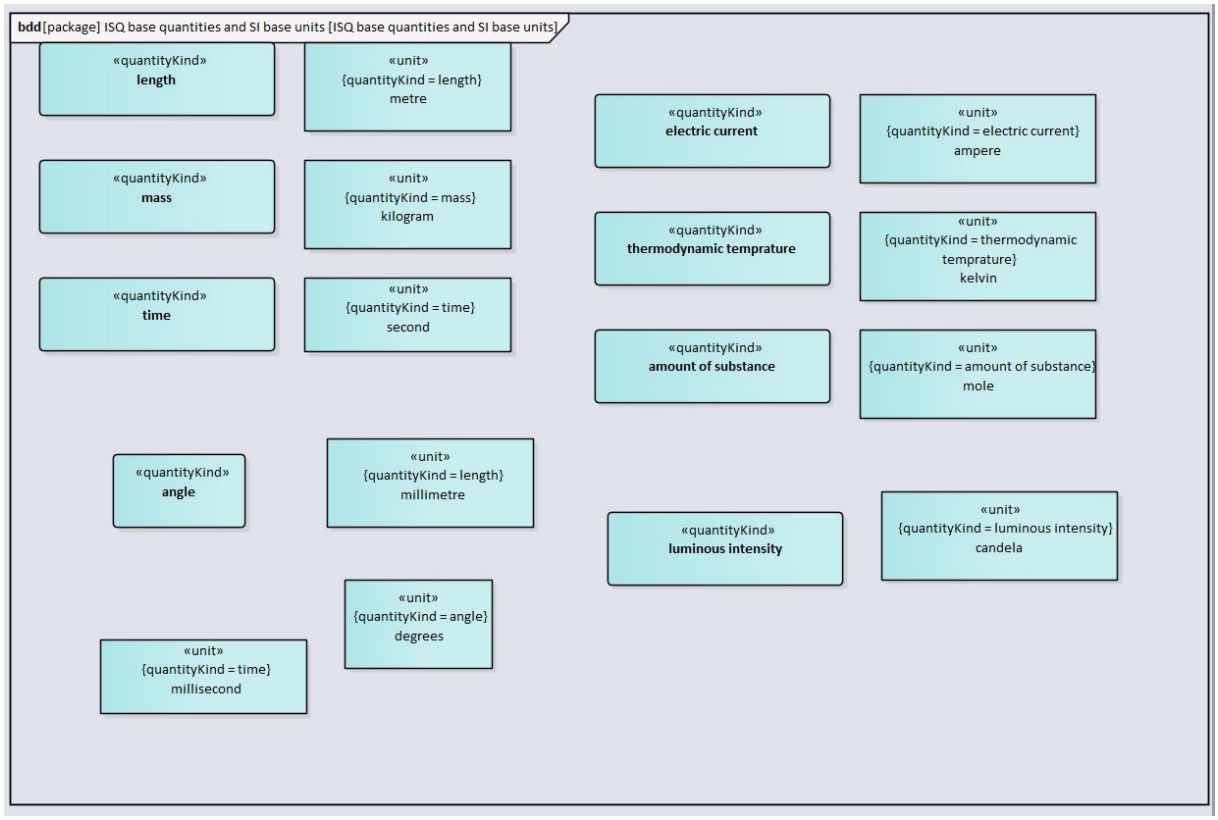


Figure 14: Example of definition of quantity kinds and units in SysML model to ensure compatibility with existing ontologies and standards

Moving on towards defining attributes of the components in the component library, previous work on sensor ontologies was reviewed. The authors in [20] focused on describing general attributes of sensor systems/ networks. An overview of the general attributes derived in this work is in Table 5.

Table 5: General attributes for sensors [20]

Attributes	Comments
Physical	Power, weight, size
Operating conditions	Environmental conditions required for operation
Immediate data	Characteristics of data, resolution (time/spatial/latency/frame rate/other)
Derived data	Results computed from raw data , both physical and semantic
Algorithms	Alternative algorithms for producing derived data
Integration/fusion	Data produced by combining data from multiple sensors
Capabilities	Functional applications of raw and derived data
Communication	Physical and logical protocols, and interoperability
Processing	On board processing power of sensors and network nodes
Calibration	Individual and joint sensor calibration information/algorithms
Provenance	Record of sensor and processing history of raw and derived data
Confidence	Levels of confidence in derived data

While these provide a useful high-level means for viewing sensor technology, they do not have a sufficient level of granularity for completely describing them from the standpoint of the design of HRC applications. Attributes such as the physical ones listed in Table 5 are important for the CAD depiction of the sensor, but other aspects regarding the capabilities of the system need further refining to support a meaningful evaluation of the safety related aspects of sensors. The sensor ontology SSN [81] has provided slightly more detailed list of attributes than from [20], including aspects like operating range. Table 6 shows the properties used to describe a laser scanner and

their specified types, including concepts such as “region of interest” and further properties relating to the angle scanned and the time and signal intensities involved. In particular, a generic representation of the response time of the sensor, here represented by the name “scan_time”, the range, and the scanned sector (described by “angle_min” and angle_max”) are all very relevant attributes. While the type is specified here, other relevant attributes such as quantity kind are not. Furthermore, while these attributes contain information that is perhaps necessary for the description of the components, it becomes clear that safety was not in the focus of the ontology development efforts.

Table 6: Excerpt of laser scanner properties from sensor ontology SSN [81]

Name	Sub-property Name	Type
RegionOfInterest		
	x_offset	uINt32
	y_offset	uINt32
	height	uINt32
	do_rectify	bool
	width	uINt32
LaserScan		
	header	Header
	angle_min	Float32
	angle_max	Float32
	angle_increment	Float32
	time_increment	Float32
	scan_time	Float32
	range_min	Float32
	range_max	Float32
	intensities	Float32

In order to determine whether these existing definitions are sufficient, a bottom-up approach that considers generic types of sensors will be used to analyze the information needed to complete the safety evaluation and calculations. Following the generic case, specific models of sensors are also analyzed to determine if there is a difference between the generic and the specific case.

4.2 Overview of available sensors for speed and separation monitoring

As previously mentioned, the emergence of new safety sensors has been one of the key enabling features for HRC applications. In particular, for the safeguarding mode SSM, there are a few key types of sensors that are used most often and that will be modeled for the purpose of use with the architecture specified in Section 3. These sensors are:

- Laser scanner
- Light curtain
- Floor mat
- Projection-based camera system

The modeling efforts will initially focus on describing a generic sensor of the type, before looking into more detail for specific sensor models. It is worth mentioning that the origin of the data (i.e. where do initial values come from?) is also considered in this analysis. Some parameters are inherent to the component and can be derived from

the data sheet, which will motivate the need for manufacturer created e-data sheets. Other data will be given from the designer and can be based upon other available information, such as results of the risk analysis. Finally, some attributes are based solely upon constraints, either as results of the simulation (e.g. the joint positions as programmed by the designer) or through other inequalities. In the following section, I will briefly describe the work related to modeling the individual sensors and the attributes needed to evaluate the safety of HRC applications.

4.3 Model development of laser scanner

Generic laser scanner

As an example, when considering the usage of a laser scanner for the safeguarding mode Speed and Separation Monitoring (SSM), the goal is to maintain a time-dependent protective separation distance $S_p(t)$ to the robot that must not be violated by the human during robot operation. The main relationship for calculation of the minimum required separation distance as specified in the ISO/TS 15066 with for SSM is in Equation 3.2. In particular, for the non-trivial case that the robot is not motionless, the main equation for determining the minimum required separation distance, $S_p(t)$, can be described as the sum of a number of individual terms in the following equation.

$$S_p(t) = S_h + S_r + S_s + C + Z_d + Z_r \tag{4.1}$$

Section 3.2.3 described the background for the terms, and for the model development. The value C is the overreach constant and indicated how far a body part can penetrate the safety area before it is sensed. This value is primarily determined by the type of sensor that is used. In the case of a laser scanner, the value is primarily determined by the orientation of the laser scanner, θ_{LS}, the height at which the sensor has been mounted, H_{LS}, and the resolution of the sensor, d_{LS} whereby:

$$\theta_{LS} = 0° \tag{4.2}$$

corresponds to the laser scanner in a horizontal position, with the scanning field parallel to the floor, and

$$\theta_{LS} = 90° \tag{4.3}$$

corresponds to the laser scanner in a vertical position, with the scanning field perpendicular to the floor.

$$C_{LS} = \begin{cases} 1200\ \text{mm} - (0.4 \cdot H_{LS}) & \theta_{LS} = 0° \\ 8 \cdot (d_{LS} - 14\ \text{mm}) & \theta_{LS} = 90° \end{cases} \tag{4.4}$$

Combining equations 5.2, 5.4 and 5.6 in 5.1, we see that the following sensor parameters:

- sensor reaction time T_r,
- sensor resolution, d_{LS},
- height of the sensor, H_{LS}, as well as
- orientation of the sensor, θ_{LS}
- sensor uncertainty, Z_r

all play a role in determining the required size of the separation distance. As a first instance, it is possible to model generic laser scanner featuring these key safety-related attributes

$$S_p(t) = \begin{cases} \left(1.6 \times (T_r + T_s)\right) + (T_r \cdot v_r) + S_s + \left(1200\ \text{mm} - (0.4 \cdot H_{LS})\right) \\ \quad + Z_d + Z_r, \quad \theta_{LS} = 0° \\ \left(1.6 \times (T_r + T_s)\right) + (T_r \cdot v_r) + S_s + \left(8 \cdot (d_{LS} - 14\ \text{mm})\right) \\ \quad + Z_d + Z_r\,, \quad \theta_{LS} = 90° \end{cases} \tag{4.5}$$

Table 7: Overview of attributes for generic laser scanner

Laser scanner attribute	Description	Units
H_{LS}	Height that sensor is positioned relative to ground	mm
θ_{LS}	Angle of sensor relative to the plane of the ground	deg
T_r	Reaction time	ms
d_{LS}	Resolution	mm
Z_r	Sensor uncertainty	mm

Specific laser scanner: SICK microScan3

To support real world implementation, specific models of individual sensors are needed and further configuration parameters such as multi-sampling or constraints between reaction time, sensor resolution and scanning range also need to be included in the model. In this case, we also see that we have static attributes (e.g. those that can be derived from a data sheet and that are not independent of a user-specific configuration or setting) and dynamic attributes that are chosen by the user.

As an example, the SICK laser scanner microScan3 [35] requires users to specify a number of configuration parameters to determine the reaction time, T_r, of the sensor. In particular, the following parameters are required to determine the sensor's reaction time:

- Scan cycle time, T_{st}
- Set interference protection, T_{ip}
- Set multi sampling, n
- Time for processing and output, T_{po}

$$T_r = \left(T_{st} + T_{ip}\right) \times n + T_{po} \tag{4.6}$$

The values for the scan cycle time are themselves determined from a look-up table and are dependent on the range and resolution of the sensor.

$$T_{st} = f(d_{LS}, r_{LS}) \tag{4.7}$$

The relationship between the scan cycle time, the resolution and the range as a look-up table gives the designer the freedom to choose which parameters are more relevant for their application and to set them. As an example, the designer can opt for the lowest scan cycle time and then use a corresponding combination of resolution and range values, or the designer can choose a specific resolution and then use the corresponding range and scan cycle time values.

Table 8: Overview of attributes for specific laser scanner SICK microScan3

Laser scanner attribute	Description	Units
H_{LS}	Height that sensor is positioned relative to ground	mm
θ_{LS}	Angle of sensor relative to the plane of the ground	deg
T_r	Reaction time	ms
d_{LS}	Resolution	mm

Z_r	Sensor uncertainty	mm
T_{st}	Scan cycle time	ms
T_{ip}	Set interference protection	ms
n	Multi-sampling	unitless
T_{po}	Time for processing and output	ms
r_{LS}	Range of laser scanner	mm

This relationship puts designers in a conundrum, as they are required to make assumptions for the range without knowledge of the size of the safety zone they need to monitor, which is the outcome of Equation 4.5 and for which the range is an input. Therefore, some means for checking the validity of the designer's assumptions is also necessary, and as addressed in Section 3.1.3.

Specific laser scanner: Hokuyo UAM-05LP

To highlight how the specific models for sensors can vary according to manufacturer and make, a second laser scanner model was developed. In this case, the safety laser scanner UAM-05LP from Hokuyo [74] was analyzed. The C-value for the laser scanner is calculated in the same manner as Equation 4.4.

Following the same method as used for the previous example, the reaction time of the sensor according to manufacturer data sheet was determined. In this case, the minimum response time , $T_{rt}$of the sensor is defined as 60 ms, and the max response time is 270. The user can freely choose a time within those limits in 30ms discreet steps. The manufacturer suggests default times of 90 ms for $\theta_{LS} = 90°$ and 120 ms for $\theta_{LS} = 0°$, and the robustness of the sensor against false positive measurements is increased with higher reaction time settings. There are four more settings that also affect the reaction time, namely:

- Area switching when the configured sensor safety area is switched by a control system , q_{switch}
- Muting on/off, , q_{mute}
- Scan skip, which can be set to one of the integer values [0, 1, 2, or 3] and that can be used to reduce the interference to the surrounding equipment by skipping a scan for a specified number of cycles , s_{skip}
- Master/slave functionality [enabled/disabled] , q_{ms}

Their influence on the reaction time is determined by look-up tables.

$$T_r = f\left(T_{rt}, \theta_{LS}, q_{switch}, q_{mute}, s_{skip}, q_{ms}\right) \tag{4.8}$$

The uncertainty of the sensor is also dependent on the reflectivity in the environment, R. This indicates that the environment also requires an attribute that the designer should set as an assumption and that needs to be validated in the implementation lifecycle phase.

$$Z_r = \begin{cases} 100\ mm, for\ low\ environmental\ reflectivity \\ 200\ mm, for\ high\ environmental\ reflectivity \end{cases} \tag{4.9}$$

Table 9 shows an overview of the attributes specific for the safety laser scanner UAM-05LP from Hokuyo.

Table 9: Overview of attributes for specific safety laser scanner Hokuyo UAM-05LP

Laser scanner attribute	Description	Units
H_{LS}	Height that sensor is positioned relative to ground	mm

θ_{LS}	Angle of sensor relative to the plane of the ground	deg
T_r	Reaction time	ms
d_{LS}	Resolution	mm
Z_r	Sensor uncertainty	mm
T_{rt}	Scan response time	ms
q_{switch}	Area switching	unitless
q_{mute}	Muting	unitless
s_{skip}	Scan skip count	unitless
q_{ms}	Master-slave function on/off	unitless
R	Reflectivity of the environment	

4.4 Model development of light curtain

Generic light curtain

Using the from Equation 4.1 as our starting point, and using the following equation from the ISO 13855 to determine the C-value of a light curtain:

$$C_{LC} = 8 \cdot (d_{LC} - 14\ \text{mm}) \tag{4.10}$$

We can arrive at the inequality for determining the size of the minimum protective distance for a generic light curtain as:

$$S_p(t) = \big(1.6 \times (T_r + T_s)\big) + (T_r \cdot v_r) + S_s + \big(8 \cdot (d_{LC} - 14\ \text{mm})\big) + Z_d + Z_r \tag{4.11}$$

Considering the equations 4.10 and 4.11, the sensor parameters required to determine the minimum protective separation distance with a light curtain are listed in Table 10.

Table 10: Overview of attributes for a generic light curtain

Light curtain attribute	Description	Units
d_{LC}	Resolution of the light curtain	mm
T_r	Sensor reaction time	ms
Z_r	Sensor uncertainty	mm

Specific light curtain: SICK C400 Standard

Analog to the situation with the laser scanner, the relevant configuration and attributes of a system play a large role in determining the real behavior of the sensor and usage of the generic parameters can lead to a configuration that is not physically possible or valid for the situation. The response time of the safety light curtain is defined a function of the following variables:

- physical resolution, d_{LC}
- height of protective field, H_{LC}
- width of protective field, W_{LC}
- number of beams used, B_{LC}
- beam coding, C_{LC}
- response time, T_{rt}
- configuration of floating blanking, F_{LC}
- number of cascaded systems, M_{LC}
- the output signal switching device used, O_{LC}

- number of guest light curtains used, G_{LC}
- response time of guest light curtains used, $T_{rt\_guest\_n}$

The dependencies between the individual parameters are provided in the form of look-up tables and can be represented by the following three general relationships:

$$d_{LC} = f(H_{LC}, W_{LC}) \quad (4.12)$$
$$B_{LC} = f(d_{LC}, H_{LC}) \quad (4.13)$$
$$T_{rt} = f(B_{LC}, C_{LC}) \quad (4.14)$$
$$T_r = f(T_{rt}, F_{LC}, M_{LC}, O_{LC}, G_{LC}, T_{rt\_guest\_n}) \quad (4.15)$$

Table 11 shows an overview of the additional attributes necessary to model the safety light curtain SICK C4000.

Table 11: Overview of attributes for specific safety light curtain SICK C4000 Standard

Light curtain attribute	Description	Units
d_{LC}	Sensor physical resolution	mm
H_{LC}	Height of protective field	mm
W_{LC}	Width of protective field	mm
B_{LC}	Number of beams used	unitless
C_{LC}	Beam coding	unitless
T_{rt}	Sensor response time	s
F_{LC}	Configuration of floating blanking	unitless
M_{LC}	Number of cascaded systems	unitless
O_{LC}	The output signal switching device used	unitless
G_{LC}	Number of guest light curtains used	unitless
$T_{rt\_guest\_n}$	Response time of guest light curtains used	s
Z_r	Sensor uncertainty	mm

4.5 Model development of safety floor mat

Generic safety floor mat

A pressure-sensitive safety floor mat used for SSM has the same general properties as the other sensors, namely the C-value and the reaction time, T_r. A further consideration is whether the mat is mounted over a step with height, h_{step}. In such a case, the C-value is determined by the following relationship:

$$C_{mat} = 1200\ mm - (0.4 * h_{step}) \quad (4.16)$$

This inequality simplifies to the constant value of 1200 mm for case when the $h_{step} = 0$:

$$C_{mat} = 1200\ mm \quad (4.17)$$

Further constraints are in the width of the mat, $w_{mat} \geq 750\ mm$. The reaction time of the sensor is the sum of the following components:

- response time safety mat, T_{rt_mat}
- response time safety controller, T_{rt_sc}

Table 12 shows the attributes required for a model of a rectangular safety floor mat.

Table 12: Overview of attributes for a generic safety floor mat

Floor mat attribute	Description	Units
h_{step}	Height of eventual step, upon which the mat is mounted	mm
T_r	Reaction time	ms
T_{rt_mat}	Safety mat response time	mm
T_{rt_sc}	Safety controller response time	ms
W_{mat}	Width of safety mat	mm
L_{mat}	Length of safety mat	mm

Specific safety floor mat: Fraunhofer IFF tactile floor mat

The tactile floor mat patented and developed by the Fraunhofer IFF differs from traditional safety floor mats in that it has a configurable spatial resolution and can detect the position of objects on the surface within the limits of this spatial resolution. Traditional safety floor mats only deliver binary information about whether an object is exerting pressure on the mat, without further information about its location. The spatial resolution of the Fraunhofer IFF tactile floor mat therefore can be use to measure the direction and speed of approaching humans. There are currently no sensors available on the market that have achieved performance level d, category 3 according to ISO 13849 and are able to deliver this information about the speed and direction of motion of approaching humans.

Table 13: Overview of attributes for IFF tactile floor mat

Floor mat attribute	Description	Units
h_{step}	Height of eventual step, upon which the mat is mounted	h_{step}
T_r	Reaction time	ms
d_{LS}	Spatial resolution (same in x- and y-directions)	mm
Z_r	Sensor uncertainty	mm
W_{mat}	Width of safety mat	mm
L_{mat}	Length of safety mat	mm
v_h	Approach speed of human	mm/s

4.6 Model development of projection based workspace monitoring system

The projection based workspace monitoring system [51] is a sensor system patented and developed by the Fraunhofer IFF. The system consists of a combination of a projector and cameras (Figure 15), and is capable of establishing safety zones of arbitrary shape, size and position in a collaborative workspace. The sensor is specifically designed for use with SSM and can take the robot, its tooling, and any parts manipulated into account in the calculation of the size of the required safety zone. Figure 16 shows a desktop application with a lightweight robot and its tool safeguarded by the projection based workspace monitoring system. The sensor is currently not commercially available. Nevertheless, initial evaluations have shown that it has a high potential for safety certification due to its robustness in different light conditions and the measures implemented for intrinsic safety.

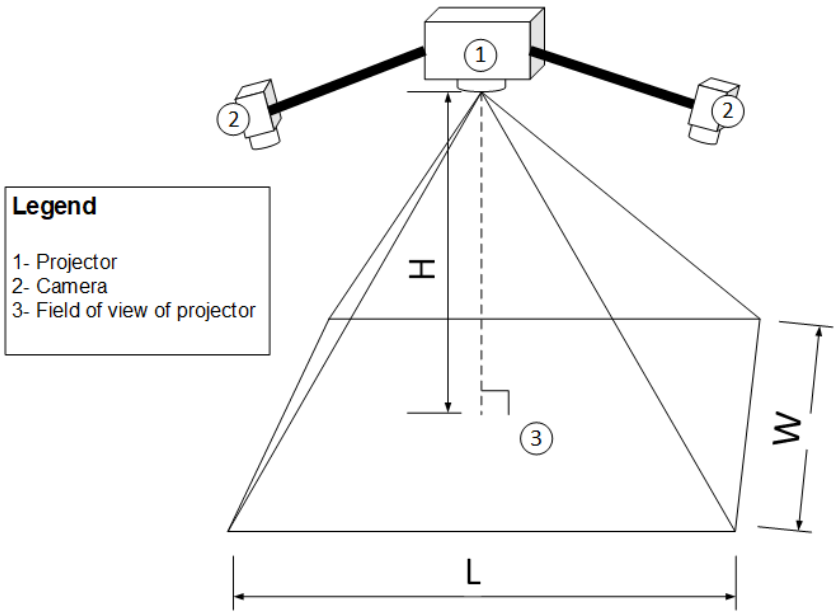


Figure 15: Schematic diagram of projector-based workspace monitoring system

The modeling effort for this sensor focused on the parameters relevant for the calculation of the minimum required safety distance, as well as the physical relationships regarding the viewing area of the sensor (see Section 3.1.3) and the validity of the sensor positioning in the environment.

The C-value of the sensor is calculated according to Equation 4.10, and the reaction time of the sensor is currently a fixed value and exhibits no dependencies due to the sensor configuration.

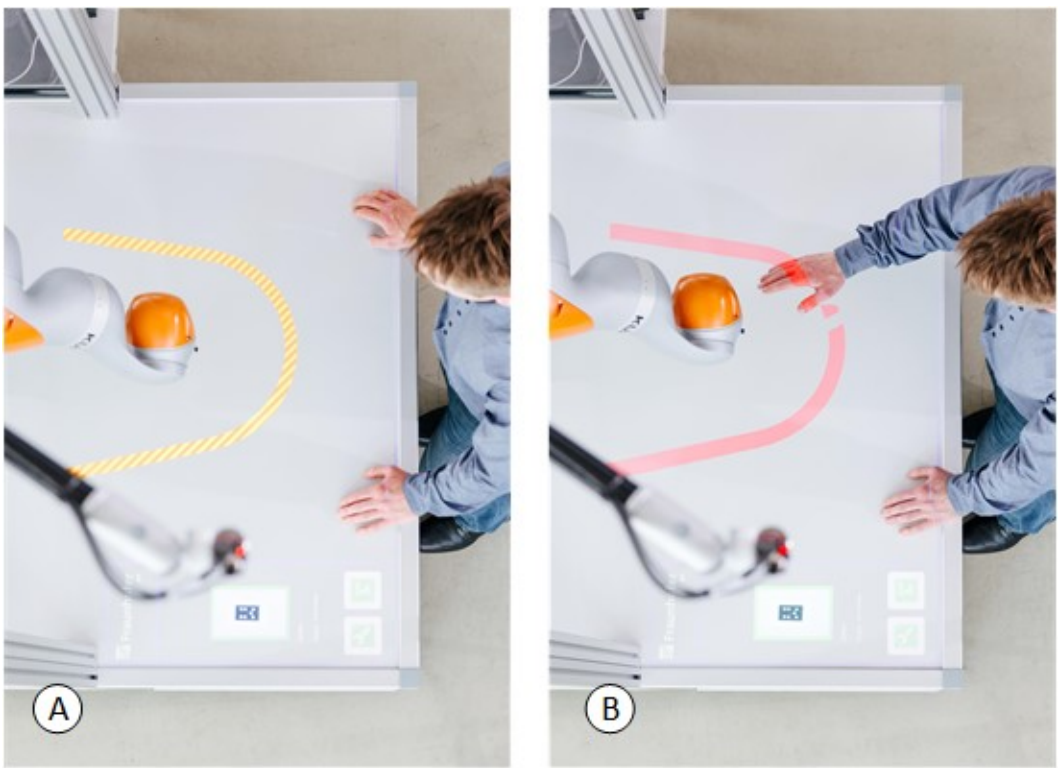


Figure 16: Example of the projection based workspace monitoring system: A) no intrusion into safety zone by the operator; B) the intruding hand is detected by the system. © Fraunhofer IFF

For the physical modeling of the sensor, a 3D description of the volume of the projected light (in the shape of a rectangular pyramid) is sufficient for the designer to review whether the sensor is able to sufficiently monitor the required areas. This 3D volume of projected light can can be described by the height of the pyramid from the apex to the base, as well as the length and width of the base (Figure 15). The position and orientation of the sensor in the environment is also required and sourced from the CAD/simulation tool.

Combining this information, the following attributes in Table 14 are required to model the safety related aspects of the projection based workspace monitoring system.

Table 14: Overview of attributes for generic projection based workspace monitoring system

Projection based workspace monitoring system attribute	Description	Units
x_p	Position of the sensor in the x-direction	mm
y_p	Position of the sensor in the y-direction	mm
z_p	Position of the sensor in the z-direction	mm
α_p	Angle of the sensor around x-axis	deg
β_p	Angle of the sensor around y-axis	deg
γ_p	Angle of the sensor around z-axis	deg
H_p	Height of the pyramid of the projected light (base to apex)	mm
W_p	Width of the base of the pyramid of projected light	mm
L_p	Length of the base of the pyramid of projected light	mm
θ_{LS}	Angle of sensor relative to the plane of the ground	deg
T_r	Reaction time	ms
d_{LS}	Resolution	mm
Z_r	Sensor uncertainty	mm

4.7 Recommendation for sensors to enable HRC

The generic sensor models are sufficient for an initial analysis of the safety related aspects of HRC applications featuring SSM. However, in order to support the later work of implementation and validation as described in Figure 8, the specific sensor models should be used. The determination of specific sensor configuration parameters in the design phase essentially front-loads some of the overall engineering work, since the settings for a safety laser scanner are not usually set until the commissioning phase of a system. Using this methodology, the scanner configuration necessarily needs to be determined during the design phase in order to fully understand the effect that configuration has on the safety related parameters, and in many cases will require the designer to make decisions earlier in the overall process than now. This methodology nevertheless has the advantage that the configuration only needs to be determined one time during the design phase, and the used values can be exported to the sensor later during the commissioning phase. Indeed, this method would not only streamline the overall workflow, but it would be necessary to ensure that the real system as built conforms to the assumptions made in the design phase of the system.

One of the main weaknesses of the sensors for SSM is their current inability to measure direction and speed of approaching humans in a safety rated manner [10]. The portion of the minimum required separation distance due to the approach speed of humans, S_h, is usually a sizeable portion of the overall separation distance, as we will see in Section 5. The tactile floor mat from the Fraunhofer IFF is able to deliver this information, but has the disadvantage of having the largest C-value of all the sensors

modeled in this work, at 1200 mm. Therefore using the tactile floor mat for SSM leads to small separation distances in situations when operators are not near the robot or when their direction of motion is parallel or away from the robot's direction of motion. However, should operators move towards the robot safeguarded by tactile floor mats, the size of the minimum required separation distance can jump quickly once a person has stepped on the mat and moves in the direction of the robot The tactile floor mat still needs to cover a large area in order to be used. Further analyses of the frequency of passing operators are necessary to determine whether the ability to measure the speed and direction of approaching operators is indeed an advantage.

The C-value for the overreach often makes a large contribution to the overall minimum separation distance. As we will see in Sections 5 and 6, it can often be advantageous to use sensors with a lower C-value to reduce the size of the minimum separation distance. Such solutions can have the disadvantage that they require fencing over a portion of the collaborative workspace (e.g. a combination of light curtains and fencing), and the overall applications requirements here play a large role in the final choice of sensor.

The reaction time of the sensor necessarily includes other safety-related computing components such as external sensor controllers or a safety PLC. These components necessarily add quite a bit of latency to the system (see example of generic safety floor mat in Section 4.5) and can easily be overlooked in the initial phase of a design. Including these in the model supports the novice designer who lacks the experience to also consider these hidden sources of latency.

Résumé Chapitre 5 – Introduction d'applications exemplaires

Deux applications industrielles exemplaires qui devraient être réalisées avec des robots collaboratifs sont décrites dans cette section. Elles seront utilisées comme exemples pour comparer l'approche actuelle et l'approche proposée pour la prise en compte des exigences de sécurité lors de la conception d'applications HRC. Les applications ont été dérivées de cas d'utilisation réels dans le secteur automobile.

Les informations suivantes sur l'application sont structurées en fonction du point de vue de l'utilisateur final. Bien qu'il existe des normes pour la description d'une tâche, par exemple pour les projets informatiques, elles se sont avérées insuffisantes dans la pratique pour une description complète des applications HRC. Les descriptions des applications figurant dans cette section sont le résultat d'une communication intense entre l'utilisateur final, avec son expertise en matière de domaine et de processus, et moi-même, y compris des visites d'atelier en direct.

La première application propose l'utilisation d'un robot à charge utile moyenne pour retirer des pièces empilées sur une palette et les placer sur une table pour d'autres opérations d'assemblage manuel. L'objectif principal de la conception est d'intégrer l'application collaborative dans l'espace de travail existant. L'espace de travail complet comprend trois tables où les opérateurs travaillent sur un sous-ensemble, et la palette, où ils retirent les pièces lourdes. Actuellement, les opérateurs sont responsables de l'ensemble du processus de déchargement de leur support de la palette, puis de l'achèvement du sous-assemblage à leurs postes respectifs.

Les supports sont livrés à la station dans une palette en bois, où ils sont empilés sur trois couches de profondeur. L'opérateur passe de sa table de montage à la palette chaque fois qu'un support est nécessaire. Il retire les supports de la palette, retourne à sa table de montage, puis pose quelques boulons pour la fixation à l'étape suivante. Lorsque le pré-montage du support est terminé, il place la pièce finie dans un autre support, où elle est ensuite transportée vers une autre station pour un nouveau montage. Les supports pèsent de 7 à 12 kg chacun et sont en acier et recouverts de peinture. Les palettes mesurent 1,2 m x 0,80 m et ont une profondeur de 0,90 m. Cette application est physiquement éprouvante pour les opérateurs, surtout lorsqu'il s'agit d'atteindre la profondeur de la palette pour obtenir la deuxième et la troisième couche de pièces.

Outre le processus et la spécification des pièces, le flux de matériaux est décrit. Dans ce cas, les palettes sont amenées dans l'espace de travail avec un chariot élévateur à fourche, et il y a suffisamment de pièces dans deux palettes (placées côte à côte) pour une équipe complète de 8 heures.

Le processus de conception type défini dans la section 2 est exécuté pour ce cas d'utilisation afin d'arriver à une conception pour l'utilisation d'un robot collaboratif. Les tâches individuelles du processus pour le fonctionnement standard et pour le moment où l'opérateur retire le matériel d'emballage sont spécifiées et les tâches sont analysées pour identifier les types possibles de fonctionnement collaboratif. L'opération standard du cas d'utilisation est divisée en 9 sous-tâches, et l'enlèvement du matériel d'emballage par l'opérateur peut être décrit avec 5 sous-tâches. Pour chaque sous-tâche, les dangers possibles sont identifiés et des méthodes d'atténuation des risques sont proposées. Les dangers les plus typiques sont la collision, le serrage et l'écrasement de différentes parties du corps, qui sont possibles lorsque le robot se déplace, prend des pièces ou les place sur la table. Dans ce cas, deux scanners laser sont choisis pour arrêter le robot lorsqu'un opérateur s'approche de la distance de séparation minimale.

Les paramètres du robot et du capteur permettant de calculer la distance de séparation requise sont choisis et expliqués, et la distance de séparation résultante

est de 2,98 m. Avec cette configuration, le robot n'atteindra pas l'objectif de conception requis pour s'adapter à la zone actuellement disponible. La grande taille des zones de sécurité requises s'étend jusqu'à la zone où les chariots élévateurs à fourche passent dans l'espace de travail et, par conséquent, le mouvement du robot serait trop souvent interrompu pour un fonctionnement normal.

La deuxième application propose l'utilisation de robots légers pour effectuer des opérations de nettoyage dans une armoire (pour capturer et réutiliser les fluides de nettoyage). Les pièces à nettoyer sont livrées à la station sur une palette et sont introduites par une seule porte (pour les pièces qui entrent et celles qui sortent). L'opérateur utilise un transpalette pour amener les nouvelles pièces dans la pièce et les monter manuellement sur le bras de la flèche d'une armoire spécifique. Le bras avec la pièce à nettoyer est inséré manuellement dans l'armoire. L'opérateur retire ensuite le transpalette et la palette de l'avant de l'armoire et les gare dans la pièce. L'opérateur retourne alors dans l'armoire et utilise un tuyau avec une buse pour nettoyer les pièces. Le support du bras de la flèche est équipé d'un axe de rotation pour faire tourner les pièces pendant le nettoyage, ce qui permet à l'opérateur d'accéder à toutes les surfaces. Lorsque la pièce a été complètement nettoyée, l'opérateur retire le bras de la rampe avec la pièce de l'armoire, charge à nouveau la pièce sur la palette et retire la palette de la pièce. Il y a trois armoires individuelles dans l'espace de travail.

Cette tâche est physiquement pénible pour les opérateurs, car ils doivent se tenir debout dans des postures inconfortables, se déplaçant à faible vitesse pendant de longues périodes - un seul cycle de nettoyage pour les pièces les plus grandes peut prendre plus de 120 minutes par pièce. Les opérateurs ont des difficultés à suivre manuellement exactement la même trajectoire sur toutes les surfaces et à s'assurer que toutes les surfaces ont un débit volumique similaire, ce qui est important pour la qualité du nettoyage.

Les pièces à nettoyer sont des pièces usinées de différents groupes motopropulseurs, comme les arbres à cames et les blocs moteurs. Elles pèsent entre 15 kg et 300 kg et sont maintenues sur une palette avec un dispositif de fixation.

La solution HRC doit répondre aux objectifs et exigences de conception suivants :

- Les solutions HRC doivent s'intégrer dans l'espace existant avec les trois armoires
- Permettre à l'opérateur de monter et d'insérer les pièces dans les différentes armoires, en tenant compte des chemins généralement utilisés pour accéder aux différentes machines depuis l'entrée de la pièce.
- Le travail dans une armoire ne doit pas être forcé de s'arrêter en raison du travail dans une autre armoire. Les différentes pièces ont des temps de nettoyage globaux très différents, ce qui signifie qu'il est difficile de planifier à l'avance quelles pièces devront être nettoyées dans quelle armoire.
- Le travail manuel dans les armoires doit également être possible.

Comme expliqué précédemment, la principale motivation derrière une solution HRC est l'amélioration de l'ergonomie pour l'opérateur, la suppression des tâches fastidieuses et répétitives, et la réduction de l'exposition aux aérosols. Le temps de cycle de la solution robotisée doit être équivalent à celui de l'opérateur humain, la solution doit s'intégrer dans l'espace existant au sol,

La tâche est analysée selon la même méthode que le cas d'utilisation précédent, et seule la tâche principale de l'opération standard est prise en compte. Cette tâche est divisée en 12 sous-tâches, et les dangers pour chaque étape sont identifiés. Alors

que le robot seul pourrait être protégé par une limitation de la puissance et de la force, le fluide à haute pression provenant de la buse est suffisamment dangereux pour que le mode de protection de la surveillance de la vitesse et de la séparation soit préférable. Une analyse utilisant des scanners laser comme capteur de sécurité à différentes vitesses du robot montre que la distance de séparation minimale requise serait de 2665 mm pour une vitesse maximale du robot de 100 %, de 2159 mm lorsque le robot se déplace à 66 % de sa vitesse maximale et de 1021 mm lorsque la vitesse du robot est réduite à seulement 33 % de sa vitesse maximale.

Une comparaison de la taille de la distance de séparation minimale avec les voies d'accès aux différentes armoires de nettoyage montre que dans le pire des cas, les robots en fonctionnement devront ralentir à 33% de leur vitesse maximale de fonctionnement pendant les périodes où les opérateurs se trouvent dans les zones de sécurité lors du chargement et du déchargement des armoires voisines. L'application d'une surveillance de la vitesse et de la séparation, par laquelle les humains qui s'approchent sont détectés et la vitesse du robot est ajustée, semble être la meilleure option dans ce cas. Bien que le concept de sécurité soit réalisable et puisse être considéré comme raisonnable, le concepteur estime qu'une analyse plus approfondie est nécessaire pour déterminer le temps de cycle global dans les trois armoires.

Il convient de souligner que de nombreux facteurs influent sur la conception d'une application robotique collaborative et qu'il est souvent difficile pour un utilisateur final potentiel et/ou un intégrateur de systèmes d'identifier ceux qui sont les plus importants pour une solution HRC. En utilisant les outils existants, la conception peut rapidement ne pas atteindre les objectifs de conception tels que l'installation dans un espace au sol spécifié, ou le temps de cycle réalisable peut sembler trop faible.

5 Introduction of exemplary applications

In order to present our approach for considering safety requirements when designing HRC applications, we will introduce two exemplary industrial applications that are currently performed manually, and which should be carried out with collaborative robots. The applications has been derived from real use-cases from the automotive sector. Specific information regarding the geometry or specific part descriptions have been changed to protect confidential, proprietary information. These changes however are only cosmetic and do not affect the relevant properties of the application or the HRC solutions.

The following information about the application is structured according to the end-user's perspective. While there are standards for description of a task [19], these methods have proven in practice to be insufficient for a complete description of HRC applications. The descriptions of the applications here are the result of intense communication between the end-user, with their domain and process expertise, and myself, including live shop floor visits.

One issue to highlight is that there are many factors which affect the design of a collaborative robotics application, and it is often difficult for a potential end-user and/or systems integrator to identify which are the most important for an HRC solution. Therefore the final discussion will also describe how the proposed approach also supports the stakeholders in early identification of the factors most relevant to the needs of their application.

5.1 Example: de-palletizing robot

The first exemplary use-case proposes the use of a collaborative robot for unloading a pallet filled with heavy brackets that weigh between 7-12 kg. After a brief description of the task and the general requirements, the standard design process with special focus on the safety-related issues will be executed. These results will later be compared against designing the proposed approach with the new approach.

5.1.1 Task description

This task focuses on the manual assembly of brackets, as part of a pre-assembly station (not a part of the moving line). The operators are responsible for the entire process of unloading their bracket from the pallet and then completing the sub-assembly at their respective stations.

The brackets are delivered to the station in a wooden pallet, where they are stacked three layers deep. The operator walks from their assembly table to the pallet whenever a bracket is needed. They remove the brackets from the pallet, walk back to their assembly tables, and then mount a few bolts for fastening to next step. When their pre-assembly of the bracket is finished, they place the finished part in another carrier, where it is then taken to another station for further assembly. This application is physically strenuous for operators, especially when reaching deep into the pallet to get the second and third layer of parts.

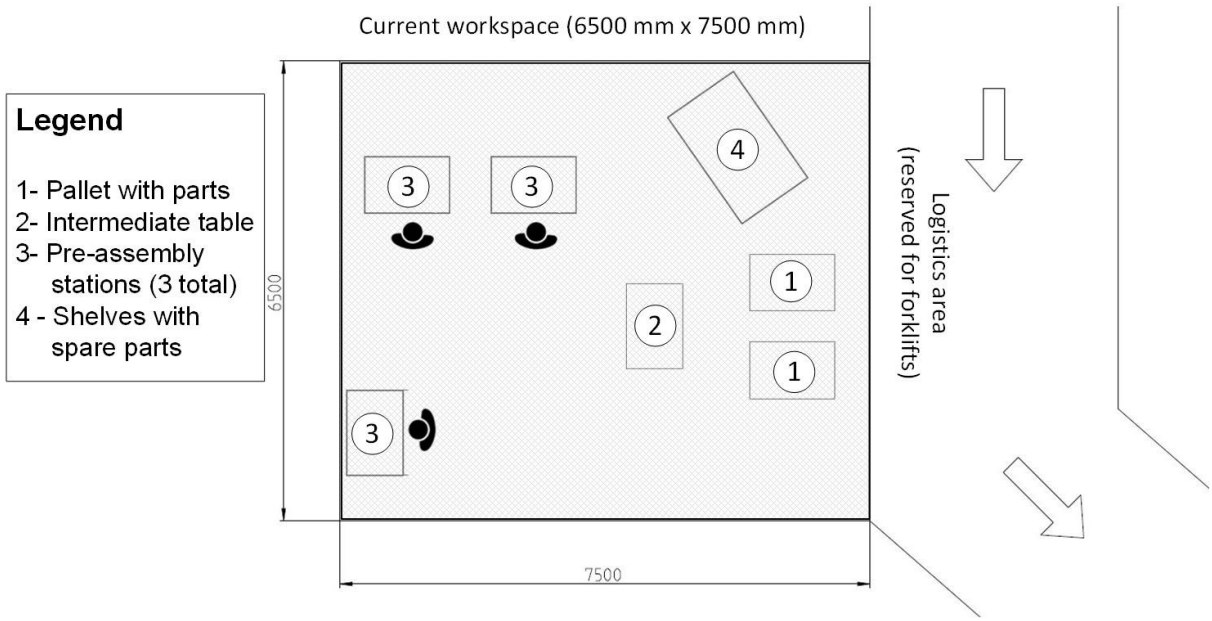


Figure 17: General layout of the depalletizing workspace.

In a first instance, manufacturing engineers would like to determine whether collaborative robots could be used for this application. The general idea is for a robot to take over the task of removing the parts from the pallet and either directly handing them to the operator or placing them on the table where the operators can take them and perform the next assembly tasks.

5.1.1.1 Part specifications

The brackets weigh 7-12 kg each and are made of steel and coated in paint. The pallets are 1.2 m x 0.80 m and are 0.90 m deep. The parts within a layer are separated by rectangular foam pieces, which are laid between the parts, and a composite plastic and foam sheet separates the layers. This separation layer is currently removed manually and is leaned next to the wooden pallet by the operator. When the pallet is empty, they are both manually inserted back into the pallet by the logistics team before exchange.

5.1.1.2 Process and layout specifications

Logistics operators via forklift bring the pallets into the workspace. In normal operation, 2 pallets are sufficient for a single 8 hour shift. Therefore, the pallets are brought into the workspace at the beginning of each shift and the empty ones removed at the end.

There are three separate workstations where operators can assembly the brackets. These are all equidistant from the two pallets that are positioned next to each other (see layout in Figure 17). The pallets are approached from one side by the operators. The rear side is the area for forklifts and logistics processes. The pallets are removed by forklift from this side.

5.1.1.3 Design targets

The most important design target is that the application fit within the existing floor space, especially without impeding on the flow of materials in the logistics area. There are other working stations surrounding the existing working area and the use of collaborative robots should not affect these other stations or their material flows.

As previously mentioned, the desired effect of the introduction of HRC to this task is to improve the worker ergonomics. The cycle time of the applications is quite long (approx. 3 minutes for the completion of the sub-assembly). It is therefore not expected that the cycle time of the robot to pick a part from the pallet and place it on the table should be critical.

5.1.2 Design of de-palletizing HRC application

In the following section, the design process as described in Section 2 will be applied to the exemplary application to highlight the current outcomes and to create a baseline for a comparison to the approach proposed in this thesis.

5.1.2.1 Starting point: General idea of collaborative application

As a starting point, the designer has decided to use a standard industrial robot with a maximum payload of 22 kg and a maximum reach of at least 1.6 m (KUKA KR22-1612). The payload and reach were chosen based on the task specifications.

The designer starts with a CAD layout of the application and simply places the parts in the layout according to the task specification.

The designer creates an excel spreadsheet of the robot speeds for the individual movements, to get a general idea of the cycle time. Based on previous experience, the planner reviews the cycle times to see if they seem reasonable, and uses this first estimate to determine whether it is economical to continue with the design of the application.

5.1.2.2 Safety oriented design

The principles of safety oriented design were applied as far as possible in this early stage. In particular, the designer understands that the robot is particularly dangerous for humans and that either an inherently safe design or one with protective measures will be necessary for the application.

5.1.2.3 General and essential requirements

Here the designer uses an internal checklist, which is derived from the Machinery Directive 2006/42/EC and the ISO 10218-1 like the one in the Appendix. The most important general requirement regards the performance level of the robot used. Either the control system of the robot (and the corresponding safety functions such as motion, speed, and/or force control) is safety rated and certified to have performance level “d”, category 3, or the risk assessment needs to prove that a lower category is possible. As a general rule of thumb, the designer normally starts all designs with a robot that has a PL d, Category 3 safety rating.

5.1.2.4 Model process and assign tasks

In this step, the designer creates a simple listing of the tasks (Table 15) that are required for the process. In this case, individual tasks include movement to specific places in the workspace (e.g. table, pallet), as well as physical manipulation (e.g. picking and placing parts). Ideally, the designer considers different tasks along the entire system lifecycle, from commissioning to productive operation and maintenance tasks. For the purposes of this paper, we have limited ourselves to two separate tasks within the lifecycle of productive operation.

Table 15: Tabular listing of process tasks for two exemplary processes: standard operation and w operator removal of packing material from the pallet.

ID Nr	*Process tasks*	*Task assignment*		*Task characteristics*		
	Standard operation	**Opera-tor**	**Robot**	**Shared work-space**	**Simul-tan-eous co-work**	**Physical contact**
S1	identify part in pallet		x	y	n	n
S2	move to pallet		x	y	n	n
S3	pick up part from pallet		x	y	n	n
S4	move to table		x	y	n	n
S5	place part on the table		x	y	n	n
S6	move to neutral position		x	y	n	n
S7	enter collaborative workspace (table)	x		y	n	n
S8	pick up part from table	x		y	n	n
S9	leave collaborative workspace		x	y	n	n
	Operator removal of packing material					
P1	identify that packing material needs to be removed, send operator signal		x	y	n	n
P2	move to neutral position and stop		x	y	n	n
P3	enter collaborative workspace near pallet	x		y	n	n
P4	remove packing materials and/or separating layer	x		y	n	n
P5	*leave collaborative workspace*	*x*		y	n	n

Using the methodology described by [16], we can also identify the type of HRC that we would like to use. As described in that work, this decision is based on the available and meaningful types of collaboration that suit the task. In this case, the designer chooses Speed and Separation Monitoring (SSM).

5.1.2.5 System limits and requirements

Based upon the floor plan and the task models previously defined, the designer specifies the system application limits and requirements. The limits include:

- the workspace of the robot;
- the configuration of the robot to reach all the critical positions within the workspace;
- the speeds to be reached;
- the payloads carried.

At the conclusion of this step, the designer checks that the system limits and requirements (e.g. for the specific safeguarding mode) are fulfilled. If not, the designer needs to change specific aspects of the design. Any changes made during this phase due to non-conformity to a specific requirement means that the designer has to start the process again from the beginning before continuing.

5.1.2.6 Hazard identification and risk evaluation

The application designer works together with a safety expert to identify hazards. They use the individual steps from the task model, and identify hazards per task step. Table 16 shows the specific risks identified.

The safety expert carries out the risk evaluation. In general, risk is the product of the severity of the damage and the probability of occurrence, and there are several

methods for a quantitative evaluation of the risk [78]. In this example, the risk evaluation shows that there is a risk of collision and clamping during a large number of individual steps in the overall process and that risk mitigation methods need to be applied to them. Following the hierarchy of risk mitigation measures, technical safeguards were chosen since an inherently safe construction for the application is not feasible.

Table 16: Tabular listing of process tasks and associated hazards.

ID Nr	*Process Tasks*	*Hazards*
	Standard operation	
S1	identify part in pallet	*none*
S2	move to pallet	*collision*
S3	pick up part from pallet	*collision, clamping, crushing*
S4	move to table	*collision*
S5	place part on the table	*collision, clamping, crushing*
S6	move to neutral position	*collision*
S7	enter collaborative workspace (table)	*collision*
S8	pick up part from table	*collision*
S9	leave collaborative workspace	*none*
	Operator removal of packing material	
P1	identify that packing material needs to be removed, send operator signal	*none*
P2	move to neutral position and stop	*collision*
P3	enter collaborative workspace near pallet	*collision, crushing, clamping*
P4	remove packing materials and/or separating layer	*collision, crushing, clamping*
P5	*leave collaborative workspace*	*collision, crushing, clamping*

5.1.2.7 Hazard elimination and risk mitigation

The risk mitigation process begins with the safety expert analyzing the results of the risk evaluation and specifying where changes need to be made. In this step the safety expert also approves the use of SSM as the safeguarding mode. Following this, the designer uses the assumptions for the robot speed, the payload, and the extension to determine the braking time and the braking distance from manufacturer data sheets. The designer uses these, as well as initial assumptions from the risk mitigation measures (e.g. safety sensors) to define the size of the minimum required safety zone. In this step, there are also quite a few important configuration settings from the safety sensors that need to be defined or assumed in order to estimate the correct safety zones. These configuration settings are discussed with the electrical engineers, as these play a role in the electrical planning (e.g. which type of bus system is used). The separation distance is usually calculated with software outside of the design-tool, for example in a spreadsheet, using values taken from a robot data sheet [42, 33]. The equations for the calculation of the separation distance are described in Equation 4.5. The calculation here represents a worst-case situation for a single, discreet action, and is then applied this over the boundary conditions of the application. A more nuanced study, considering dynamic effects, the changes in trajectory over time, etc, is not carried out due to lack of proper engineering tools.

Table 17: Parameters used to determine size of minimum protective area for a discreet robot position

Parameter	Value	Comments
v_h	1600 mm/s	Standard assumption when not possible to measure the speed of approaching humans
v_r	1000 mm/s	Max. robot speed
T_r	90 ms	Sensor reaction time from manufacturer data sheet [35]
T_s	400 ms	From manufacturer data sheet
S_h	784 mm	Contribution to safety zone due to approaching human
S_r	90 mm	Contribution to safety zone due to robot speed
S_s	1242 mm	Extrapolated from manufacturer data sheet [37] assuming combined axis 2 and 3 movement, 100% arm extension, 66% payload and 100% of max. speed
C	850 mm	Assuming a horizontal orientation of the laser scanner
Z_d	10 mm	Measurement uncertainty from sensor system
Z_r	5 mm	Positioning uncertainty of robot
S_p	**2981 mm**	**Necessary protective distance for SSM**

The protective zone is represented by a combination of circles with the radius, $r = S_p$, which are centered at the extreme outer positions that the robot reaches. In this case, the designer chose three extreme positions, namely the furthest corners of the two pallets and on the table where the parts are laid, as in Figure 18. The safety zone is represented by the combined area within the three circles.

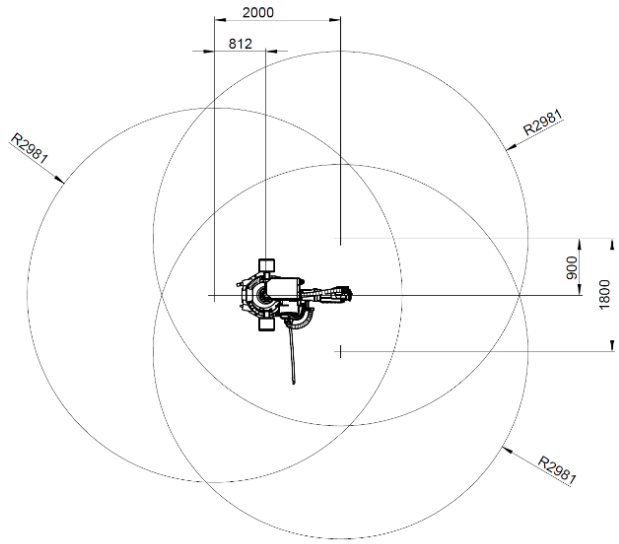


Figure 18: Layout with robot with minimum required safety distance (R=2918mm) at three main positions for picking and placing with a KUKA KR22 robot operated at maximum speed and using two horizontally oriented laser scanners with a reaction time of 90 ms and a C-value of 850mm.

Given the parameters in Table 17, and using a horizontally-oriented laser scanner as the safety sensor, a minimum separation distance of 2.98 m is required around the robot tool at all times. Figure 18 represents the floor space that needs to be safeguarded by the laser scanners. Using the dimensions between the three points that represent the furthest necessary reach of the robot in the three picking and placing positions, the total area to be safeguarded can be calculated to be over 50 m².

5.1.2.8 Review

In this step, the designer reviews the final safety concept and validates the solutions against the requirements on the system. Figure 19 shows the layout with the calculated required minimum separation distance with the initial configuration. The designer quickly sees that this configuration is not acceptable, as the required safety distance is too large for the given workspace. We see a clear overlap of the safety distance and the logistics area why passing forklifts, and operators working at assembly table 2 or accessing shelves will cause the system to stop.

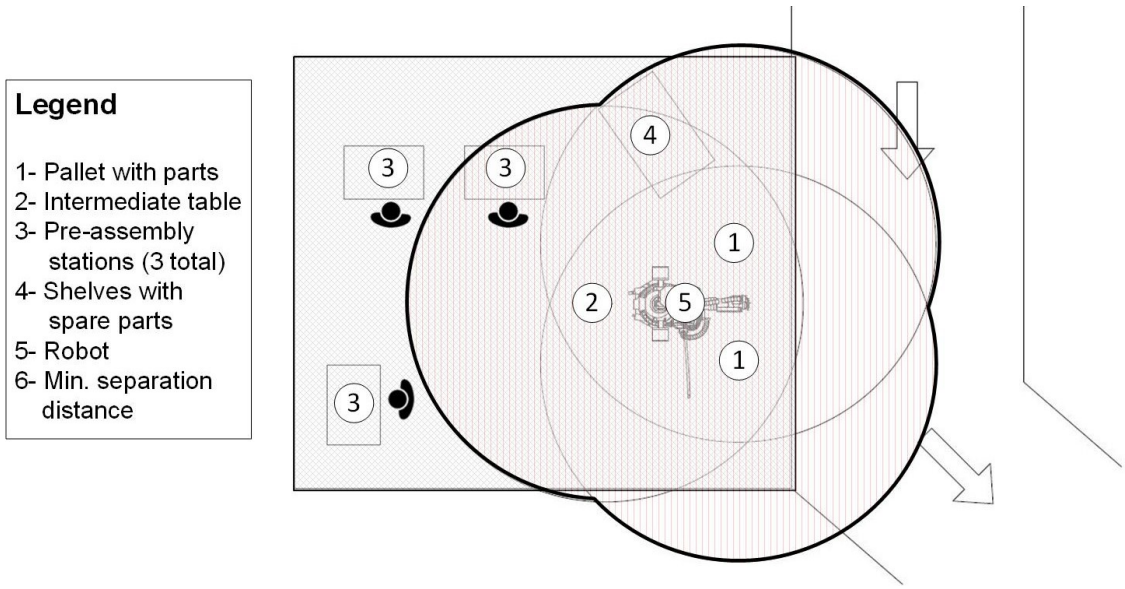


Figure 19: Review of the application with a robot and the calculated minimum separation distances using traditional worst-case calculations. The required safety areas reach well into the logistics area, where forklifts pass by.

5.2 Example: Cleaning machined parts

In contract to the first use-case, which uses a high-payload robot not specifically designed for HRC, this use-case will feature a lightweight robot to support workers during the task of cleaning machined parts. This section will follow the same structure as the first use-case. After an initial description of the task and the requirements, the design process with focus on the safety-related issues will be carried out.

The parts are loaded into the cleaning cabinets manually and currently the operator is responsible for the tasks of cleaning the parts completely. Once a part is completely cleaned, it is then manually removed from the cabinet and sent back to the line for further assembly.

We will use the same structure for describing the application as for the previous example.

5.2.1 General task description

This use-case focuses on the cleaning of various machined parts with a special cleaning fluid in specialized cabinets that filter and reuse the cleaning fluid. The parts to be cleaned are delivered to the station on a pallet and are brought in through a single doorway (for both entering and exiting parts). The operator uses a pallet jack to bring new parts into the room and mount them manually to the boom arm of a specific cabinet. The boom arm with the part to be cleaned is manually inserted into the cabinet. The operator then removes the pallet jack and pallet from in front of the cabinet and parks them in the room. The operator returns to the cabinet and uses a hose with a nozzle to clean the parts. The mount of the boom arm is equipped with a rotational axis to turn the parts during cleaning, providing the operator access all surfaces. When the part has been completely cleaned, the operator removes the boom arm with the part from the cabinet, loads the part back onto the pallet, and removes the pallet from the room.

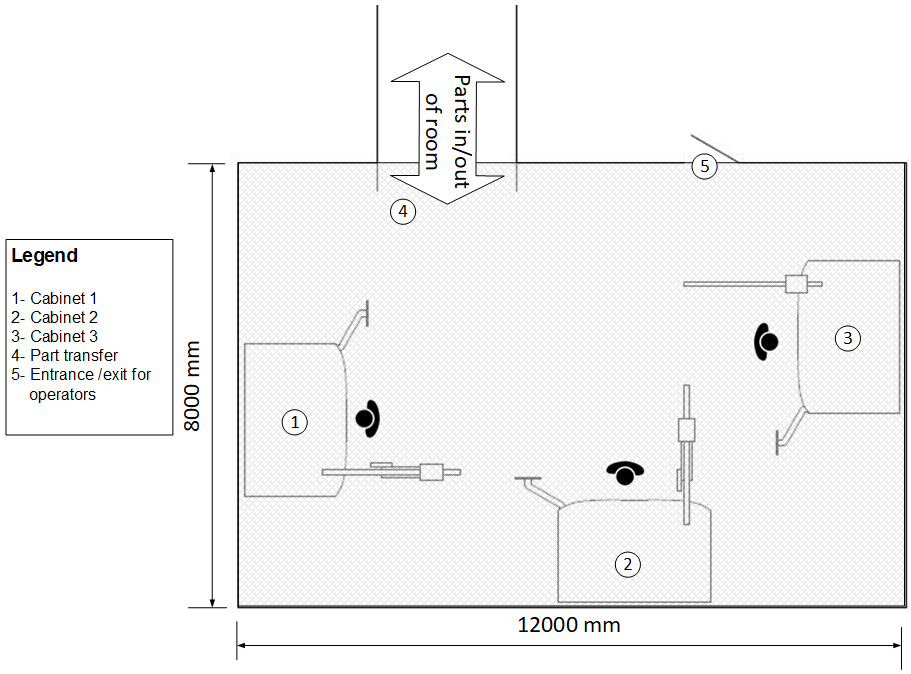


Figure 20: General layout of the part cleaning workspace.

This task is physically strenuous for operators, as they have to stand in uncomfortable body postures, moving at slow speeds for long periods of time – a single cleaning cycle for the largest parts can take in excess of 120 minutes per part. Operators have difficulty manually following exactly the same trajectory over all surfaces and ensuring all surfaces have a similar volume flow, which is important for the cleaning quality.

5.2.1.1 Part specifications

The parts to be cleaned are different drive-train machined parts such as camshafts and engine blocks. They weigh between 15 kg-300 kg and are held on a pallet with a fixture.

5.2.1.2 Process and layout specifications

The parts to be cleaned are brought on pallets by logistics operators (with forklifts) to the sluice of the room and left there. There are three separate cabinets where operators can take the parts to be cleaned.

The process of cleaning the parts can take up to two hours for a single part (with many surfaces and intermediate position changes.

5.2.1.3 Design targets

The HRC solution needs to fulfil the following design targets and requirements:

- The HRC solutions should fit in the existing space with all three cabinets
- Allow access for the operator to mount and insert the parts into the individual cabinets. The pathways for operators to access the individual cabinets are shown in Figure 21.
- Work in one cabinet should not be forced to stop due to work in another cabinet. The various parts have very different overall cleaning times, meaning that it is difficult to plan beforehand which parts will need to be cleaned at which cabinet.
- Manual work in cabinets should also be possible.

As previously explained, the main motivation behind a HRC solution is the improved ergonomics for the operator, the removal of tedious, repetitive tasks, and reduced exposure to aerosols. The cycle time of the robotic solution should be equivalent to the human operator, the solution should fit in the existing floor space,

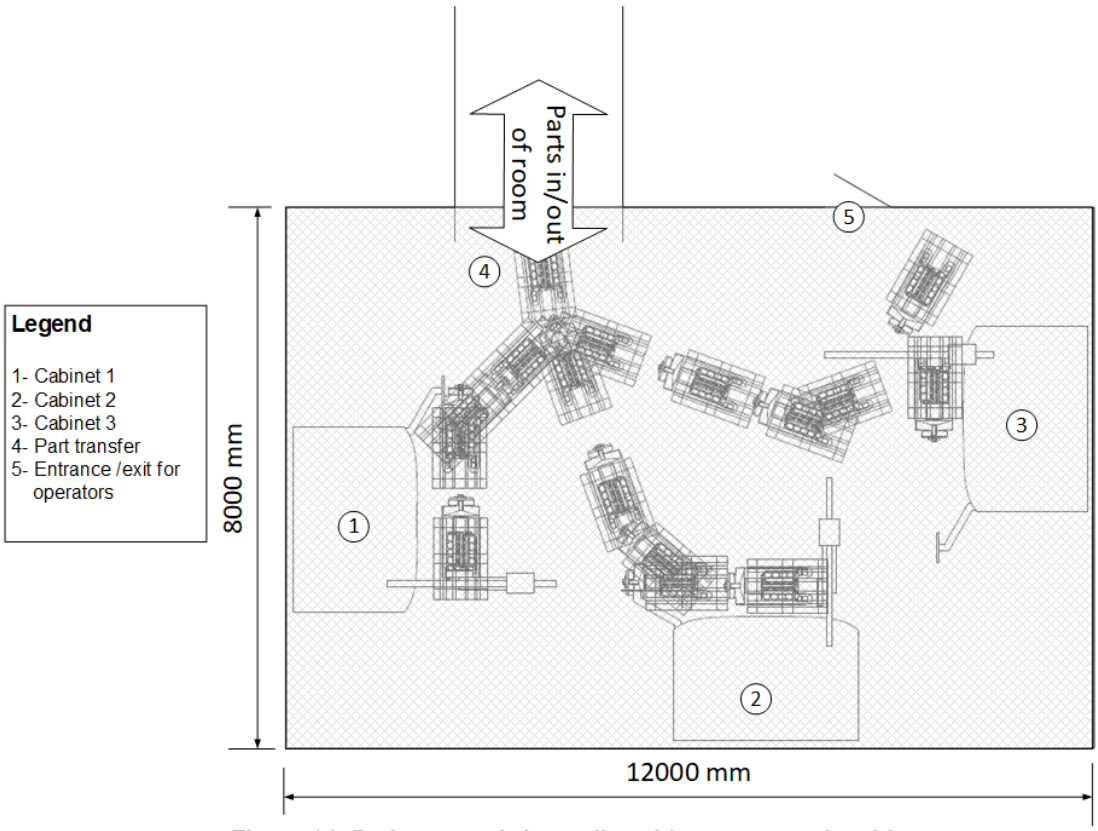


Figure 21: Pathways to bring pallet with parts to each cabinet

5.2.2 Design of parts cleaning HRC application with traditional methods

5.2.2.1 Starting point: General idea of collaborative application

As a starting point, the designer is looking into a lightweight robotic solution that can be manually moved from one position to another. This would leave the area in front of the cabinets free for insertion and removal of the parts into the cabinet.

The designer starts with a CAD design and layout of the application. The designer initially thinks that the use of a lightweight collaborative robot is advantageous, as it principally allows for use of all four types of safeguarding modes. Since the cycle time of the process is less the focus, the designer decides that it is not necessary to carry out an initial analysis or estimate at this early phase. The productivity gains are expected by the reduction in the number of operators needed for all three cleaning cabinets from three to one, and strong ergonomic gains are expected for the operator, as they no longer have the tedious and physically uncomfortable task of cleaning parts for long periods of time.

5.2.2.2 Safety oriented design

As with the previous use-case, the designer adheres to the principles of safety oriented design as far as possible in this early stage. In this case, the designer identifies that the tool and the process (i.e. cleaning fluids under pressure) represent the largest hazard to humans in the collaborative workspace. At this stage protective measures will most likely be necessary for the application.

5.2.2.3 General and essential requirements met

Again, the designer uses their internal checklist, derived from the Machinery Directive 2006/42/EC and the ISO 10218-1 to see that general and essential health and safety requirements are met. Many of these need to be revisited, as they are very concrete (e.g. requirements on labels and signs to indicate the robot's mode of operation) and cannot be fulfilled early in the design process. The most important general requirement regards the performance level of the robot used. Either the control system of the robot (and the corresponding safety functions such as motion, speed, and/or force control) is safety rated and certified to have performance level "d", category 3, or the risk assessment needs to prove that a lower category is possible. As a general rule of thumb, the designer normally starts all designs with a robot that has a PL d, Category 3 safety rating. Should this become an issue, the designer can revisit these assumptions in the later tasks of risk evaluation and mitigation.

5.2.2.4 Model process and assign tasks

The list of tasks (Table 18) required for the process are defined, and the designer defines who is responsible for each task (either the operator or the robot). In this case, individual tasks include physically moving the parts in the workspace from the sluice area to the cleaning cabinets, moving the robot from their parking position to the cabinets, and any eventual maintenance tasks during the process, such as changing the nozzle on the robot. The designer has to consider all tasks along the along the entire system lifecycle, from commissioning to productive operation and maintenance tasks. For the purposes of this thesis, the focus is limited to the tasks within the lifecycle of productive operation.

Table 18: Tabular listing of process tasks standard operation of parts cleaning for a single cabinet

ID Nr	*Process tasks*	*Task assignment*		*Task characteristics*		
	Standard operation	**Opera-tor**	**Robot**	**Shared work-space**	**Simul-tan-eous co-work**	**Physical contact**
1	Use pallet jack to bring pallet with the work piece from material lock into cleaning room.	x		y	y	n
2	Move to appropriate cleaning cabinet.	x		y	y	n
3	Mount work piece to boom arm on crane next to cabinet.	x		y	y	n
4	Move the work piece with the crane into the cleaning cabinet.	x		y	y	n
5	Move and anchor/fix the robot in front of the cleaning cabinet.	x		y	y	n
6	Mount the required nozzle/cleaning tool to the robot.	x		y	y	n
7	Rinse the work piece in a certain pattern and for a certain amount of time. The boom arm rotates work piece inside the cabinet during rinsing.		x	y	y	n
8	Perform maintenance work including changing nozzle.	x		y	y	n
9	Move robot away from in front of cabinet.	x		y	y	n
10	Remove work piece out of the cleaning cabinet.	x		y	y	n
11	Mount the work piece back onto pallet.	x		y	y	n
12	Transport the work piece out of the cleaning room.	x		y	y	n

Using the methodology described by [16], the designer can also identify the type of HRC that they would like to use. As described in that work, this decision is based on the available and meaningful types of collaboration that suit the task. In this case, the designer initially chooses Power and Force Limiting (PFL), as it provides the most flexibility for the overall process and would allow humans to get close (and even in contact) with the robot during operation. This would support the requirement of allowing operators to access cleaning cabinets to change parts while robots are working automatically in neighboring stations.

5.2.2.5 System limits and requirements

Based upon the floor plan and the task models previously defined, the designer specifies the system application limits and requirements. The limits include:

- the reach of the robot;
- the configuration of the robot to reach all the critical positions within the workspace;
- the speeds to be reached;
- the payloads carried.

At the conclusion of this step, the designer checks that the system limits and requirements (e.g. for the specific safeguarding mode) are fulfilled. If not, the designer needs to change specific aspects of the design. Any changes made during this phase

due to non-conformity to a specific requirement means that the designer has to start the process again from the beginning before continuing.

5.2.2.6 Hazard identification and risk evaluation

Table 19 shows the specific risks identified. In this example, the risk evaluation shows that there is a risk of collision and clamping during the tasks 7 and 8 that require risk mitigation measures. In particular, the safety expert decides that the hazards due to the contact with the cleaning fluid can have serious consequences for an operator. Further, when an operator is near the robot during cleaning (e.g. to carry out a maintenance task nearby), it is reasonable to expect the operator to look into the cabinet to see the status of the process, especially should the operator hear anything out of the ordinary or just out of curiosity. Following the hierarchy of risk mitigation measures, technical safeguards were chosen since an inherently safe construction for the application is not feasible. The safety expert does not consider the option of user information (e.g. placing a sign on the machine that the operator should not look into the cabinet during robotic cleaning) sufficient.

Table 19: Tabular listing of process tasks and associated hazards.

ID Nr	*Process Tasks*	*Hazards due to HRC*
1	Use pallet jack to bring pallet with the work piece from material lock into cleaning room.	none
2	Move to appropriate cleaning cabinet.	none
3	Mount work piece to boom arm on crane next to cabinet.	none
4	Move the work piece with the crane into the cleaning cabinet.	none
5	Move and anchor/fix the robot in front of the cleaning cabinet.	none
6	Mount the required nozzle/cleaning tool to the robot.	none
7	Rinse the work piece in a certain pattern and for a certain amount of time. The boom arm rotates work piece inside the cabinet during rinsing.	collision, clamping between robot and tooling and cabinet, contact with pressurized cleaning fluid
8	Perform maintenance work including changing nozzle.	collision, clamping between robot and tooling and cabinet, contact with pressurized cleaning fluid
9	Move robot away from in front of cabinet.	none
10	Remove work piece out of the cleaning cabinet.	none
11	Mount the work piece back onto pallet.	none
12	Transport the work piece out of the cleaning room.	none

5.2.2.7 Hazard elimination and risk mitigation

The risk mitigation process begins with the safety expert analyzing the results of the risk evaluation and specifying where changes need to be made. Whereas the designer initially considered the safeguarding mode of PFL, given that the robot can sense collisions and react accordingly to limit the collision forces and pressures, the safety expert prefers SSM. This is because the risk of operator injury due to the pressurized cleaning fluid is sufficiently high that something needs to be done. Therefore, the process and tooling are in this case the deciding factor with relation to the safeguarding mode. This has many consequences for the designer, as the required separation distance for one robot could overlap with the pathways the operators need to access the other machines.

As in the previous example, the designer uses the assumptions for the robot speed, the payload, and the extension to determine the braking time and the braking distance

from manufacturer data sheets. The designer uses these, as well as initial assumptions from the chosen safety sensors to calculate the size of the minimum required safety distances. The first suggestion from the safety expert is to use laser scanners. Their configuration settings are discussed with the electrical engineers. The separation distance is calculated with a spreadsheet like in the previous example, using values taken from a robot and sensor data sheets [35, 38]. Using the equations for the calculation of the separation distance from Section 3.1.2 and the values listed in

Table 20, the minimum required safety distance for the iiwa is 2665 mm. The calculation here represents a worst-case situation for a single, discreet action, and is then applied this over the boundary conditions of the application.

Table 20: Parameters used to determine size of minimum protective area for a discreet robot position

Parameter	Value	Comments
v_h	1600 mm/s	Standard assumption when not possible to measure the speed of approaching humans
v_r	1000 mm/s	Max. robot speed
T_r	90 ms	Laser scanner reaction time from manufacturer data sheet
T_s	800 ms	From manufacturer data sheet
S_h	1424 mm	Contribution to safety zone due to approaching human
S_r	90 mm	Contribution to safety zone due to robot speed
S_s	286 mm	Extrapolated from manufacturer data sheet assuming axis 1 movement, 100% arm extension, 66% payload and 100% of max. speed
C	850 mm	Assuming a horizontal orientation of the laser scanner
Z_d	10 mm	Measurement uncertainty from sensor system
Z_r	5 mm	Positioning uncertainty of robot
S_p	**2665 mm**	**Necessary protective distance for SSM**

The protective zone is represented by a combination of circles with the radius, $r = S_p$, which are centered at the extreme outer positions that the robot reaches. In this case, the designer chose three extreme positions, namely the two edges where parts should be cleaned, and the home position, spaced out as in Figure 22. As part of a what-if study, the designer also considers changes in the robot's speed could affect the size of the necessary protective distance. Table 21 shows the parameters that change with respect to the robot speed (otherwise using values from the original calculation in Table 20) and the resulting minimum protective distances.

Table 21: Comparison of minimum protective distance for different robot speeds

Parameter	Value (POV 100%)	Value (POV 66%)	Value (POV 33%)	Comments
T_s	800 ms	600 ms	430 ms	From manufacturer data sheet
S_s	286 mm	100 mm	36 mm	From manufacturer data sheet
S_p	**2665 mm**	**2159 mm**	**1021 mm**	**Necessary protective distance for SSM**

Figure 22 shows the size of the minimum protective distance for the three discreet robot speeds, 100%, 66% and 33%. There is therefore a tradeoff between the robot speed and the size of the minimum protective distance. The designer will need to

consider this when finalizing the layout. Since the safeguarding areas around the robot as shown in Figure 22 do not reflect the fact that the cabinets are directly in front of the robot, and the area behind the cabinet no longer needs to be safeguarded, the areas shown here are not meaningful for a comparison of the floor space needed, as for the previous example. In this case, the most important factor is the overlap between the safety zone for one robot and the pathways for loading and unloading parts in to the neighboring cabinets.

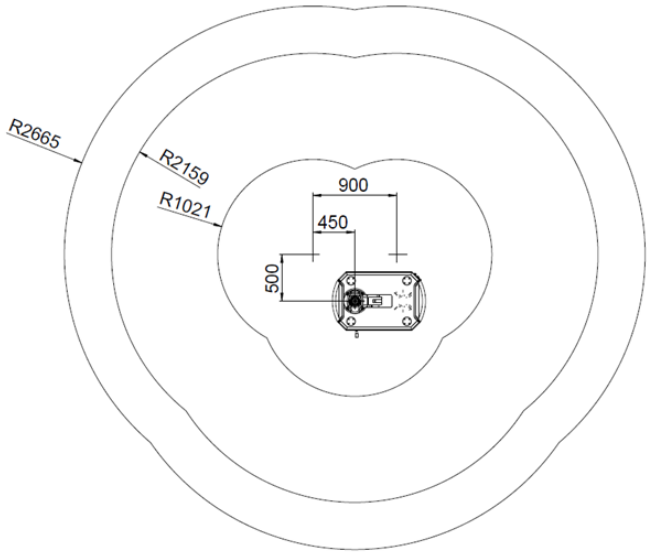


Figure 22: Size of minimum required protective distance for three discreet robot speeds (100% POV, 66% POV, and 33%POV) for a KUKA iiwa 14 robot using two horizontally oriented laser scanners as the safeguarding sensors.

5.2.2.8 Review

In the review of the safety concept, the designer validates the solutions against the requirements on the system. Figure 23, Figure 24, and Figure 25 show the layout with the calculated required minimum separation distance for three robot speeds and for the three cases when operators are loading and unloading parts into each of the three cabinets.

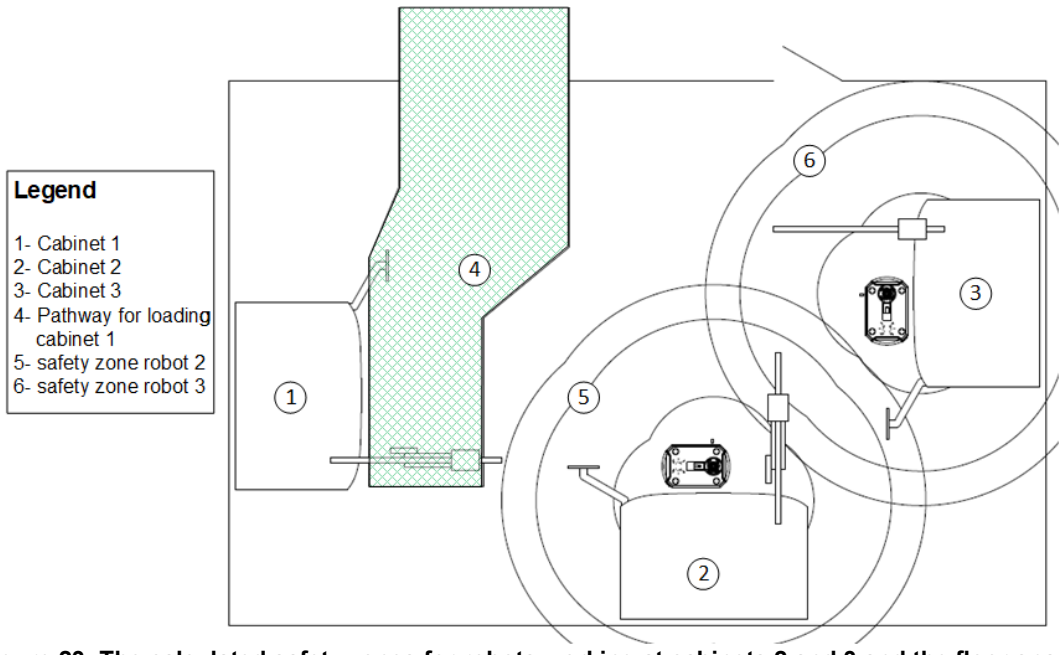


Figure 23: The calculated safety zones for robots working at cabinets 2 and 3 and the floor space needed to load / unload cabinet 1.

Figure 23 shows that there are no overlaps between areas 4, 5 and 6, indicating that operator activities at cabinet 1 will not affect robot operation in cabinets 2 and 3.

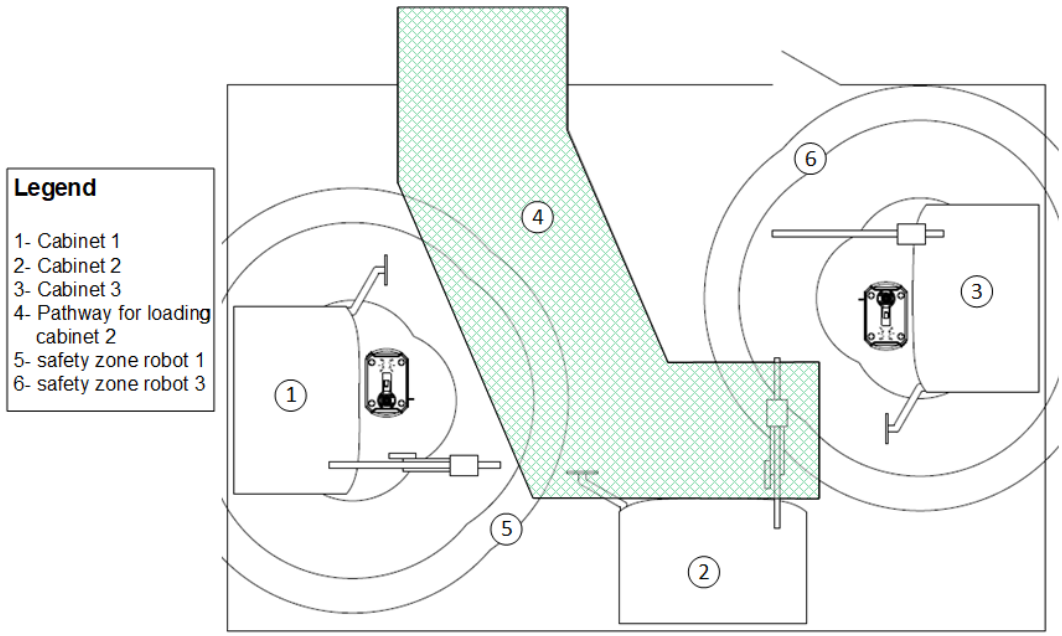


Figure 24: The calculated safety zones for robots working in cabinets 1 and 3 and the floor space needed to load / unload cabinet 2.

Figure 24 shows that there is a strong chance that the operator will enter the safety zones for robots working at cabinets 1 and 3 when loading cabinet 2. In the worst case, the robots at cabinets 1 and 3 will need to slow down to 33% of their maximum operating speed during the times that operators are in the safety zones.

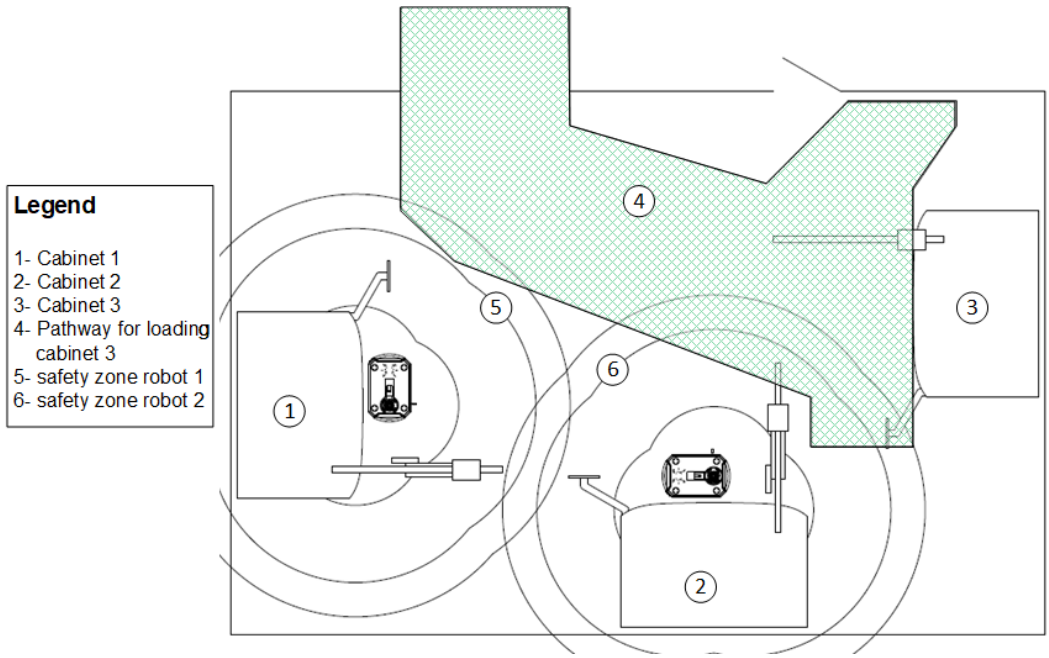


Figure 25: The calculated safety zones for robots working in cabinets 1 and 2 and the floor space needed to load / unload cabinet 3.

Figure 25 shows the situation when loading and unloading cabinet 3. Similar to the previous case when loading and unloading cabinet 2, operator loading at cabinet 3 will affect the speed of the other robots in cabinet 1 and 2. While it may be feasible to adjust the operator's pathway when passing by cabinet 1 to avoid the robot's safety zone (here a simple floor marking would be helpful), this is not possible for cabinet 2.

In summary, the robot speed will need to be adapted to adjust for situations when operators are loading cabinet 2 and 3. The application of speed and separation monitoring, whereby approaching humans are detected and the robot's speed is adjusted seems to be the best option here. In these cases, the designer quickly can see that a robot speed of 33% POV will be necessary for many instances in order to avoid situations whereby the neighboring robots are required to completely stop their movement. While the safety concept is feasible and can be considered reasonable, the designer sees that further analysis is needed to determine the overall cycle time across all three cabinets.

Résumé Chapitre 6 – Etudes de cas d'utilisation avec des applications exemplaires

Dans ce chapitre, les cas d'utilisation décrits précédemment sont analysés à l'aide d'un logiciel qui a été développé selon les spécifications des sections précédentes. Le logiciel a été conçu comme un plug-in pour le logiciel de simulation robotique Visual Components et a été développé par le Fraunhofer IFF. Le logiciel a été implémenté selon l'architecture globale décrite précédemment et avec une spécification claire de l'origine des données des modèles de composants (par exemple, de la fiche technique du composant, de l'outil de simulation ou de l'utilisateur). Les travaux de cette thèse se concentrent sur l'approche et la méthodologie de prise en compte des aspects liés à la sécurité lors de la conception des applications HRC et non sur l'implémentation du logiciel. Le logiciel utilisé dans la section 6 a été mis en œuvre par des collègues de la Business Unit Robotic Systems du Fraunhofer IFF conformément aux concepts et à l'architecture développés dans cette thèse.

Toutes les informations mécaniques et robotiques ont été extraites de la simulation et utilisées dans les modèles de composants de l'outil CAS. Cela comprend les vitesses des articulations du robot, les configurations des articulations, les spécifications cinématiques et les informations sur la charge utile. Elle comprend également des caractéristiques physiques telles que la position de tous les composants. Dans de nombreux cas, les attributs requis pour le calcul de la distance minimale de séparation n'étaient pas disponibles dans la simulation. Cela inclut des informations telles que la configuration des capteurs et d'autres paramètres concernant le robot, comme la charge utile maximale ou la portée maximale du robot. Dans ces situations, il a été demandé à l'utilisateur de remplir initialement les attributs manquants dans les modèles. Ces données ont ensuite été stockées dans une base de données interne avec les informations des modèles de composants, de sorte que cela n'a dû être fait qu'une seule fois.

Les applications décrites dans la section 5 ont été simulées dans Visual Components et le CAS-Tool comme décrit dans les sections 3 et 4. Le logiciel calcule et visualise la distance de séparation requise pour les robots et la configuration des capteurs choisis et permet au concepteur d'effectuer des analyses pour étudier comment les changements apportés au robot, à la configuration des capteurs, au processus et à la disposition affectent les cibles de conception.

Pour le cas d'utilisation de la dépalettisation, la taille de la distance minimale de séparation requise lors de l'utilisation d'un scanner laser est 44% inférieure à celle calculée manuellement. Cependant, la distance de séparation requise est encore trop grande, et avec l'outil CAS, le concepteur peut rapidement essayer d'autres variantes. Les options que le concepteur peut envisager sont les suivantes

- Modifications du robot :
    - le type de robot (avec différents paramètres de freinage, charge utile, portée, etc.)
    - programme du robot
        - la rapidité (soit pour l'ensemble de la procédure, soit pour des requêtes spécifiques)
        - des modifications visant à limiter l'extension maximale lors de mouvements spécifiques
- Changer le capteur
    - type (avec temps de réaction associé et valeurs de dépassement)
    - les paramètres d'utilisation (tels que la résolution)
- Modification du processus et de la présentation, notamment :

- l'ajout ou la suppression d'éléments de clôture
- la position des composants dans l'espace de travail
- des modifications de la tâche elle-même (par exemple, l'ordre des tâches, la responsabilité d'une tâche)

En utilisant des méthodes traditionnelles, le concepteur devrait conserver un certain nombre de documents séparés, notamment une mise en page CAD, des feuilles de calcul des distances de sécurité et une bibliothèque de fichiers PDF contenant des données sur les robots et les capteurs. Grâce à notre approche, le concepteur peut essayer toutes les différentes options énumérées ci-dessus dans l'environnement de simulation du robot et peut immédiatement voir les effets de tout changement spécifique. Seules les configurations de système valides peuvent être utilisées grâce aux modèles internes, et les données des composants sont enregistrées dans une base de données et facilement disponibles pour une utilisation ultérieure.

Afin d'illustrer cela, une série d'analyses avec quatre configurations totales différentes a été réalisée. Deux configurations ont utilisé un scanner laser pour protéger le robot, et la vitesse du robot a été spécifiée pour être à 100% de sa vitesse maximale dans un cas, et réduite à 33% de la vitesse maximale dans le second. Les troisième et quatrième configurations ont toutes deux utilisé une combinaison de barrières et de rideaux lumineux pour protéger le robot se déplaçant à sa vitesse maximale, les différences se situant au niveau de la configuration du capteur. Une configuration comportait un rideau lumineux réglé à une résolution de 40 mm, et l'autre utilisait un rideau lumineux réglé à une résolution de 20 mm. La configuration avec le rideau lumineux réglé sur une résolution de 20 mm était la plus petite sur le site de l'usine, ne nécessitant que 9,5 m² d'espace. Cette solution pouvait facilement s'adapter à l'espace disponible au sol et le mouvement du robot n'était pas interrompu par le passage des chariots élévateurs. En comparaison, la solution initiale de la section 5 utilisant des méthodes traditionnelles a permis d'obtenir une surface au sol de 50,7 m².

Il convient de noter que les quatre configurations retenues pour cette analyse ne représentent qu'un petit nombre des options dont dispose le concepteur. Une autre option de conception que le concepteur peut rapidement tester est de savoir si les modifications apportées à la configuration pour diminuer l'extension du robot et améliorer indirectement le temps et la distance de freinage.

Pour le cas d'utilisation du nettoyage, trois types de capteurs différents, à savoir les scanners laser, les tapis de sol et les systèmes de surveillance de l'espace de travail par projection sont testés en simulation. La surface totale requise pour la distance de séparation minimale n'est pas une mesure utile pour ce cas d'utilisation, car cette surface contient également l'armoire de nettoyage et n'a pas besoin d'être protégée. Une métrique préférée pour comparer la taille de la distance de séparation requise est la distance horizontale entre le centre du robot TCP et le bord le plus éloigné de la zone de sécurité. L'utilisation de la méthode proposée permet de réduire cette distance de 20 % par rapport aux méthodes traditionnelles.

L'outil CAS a également été utilisé pour tester les effets de différents types de capteurs de sécurité sur l'application, notamment avec des systèmes de surveillance de l'espace de travail par projecteur et des tapis de sol tactiles. Une comparaison des trois capteurs a été effectuée, en utilisant le même indicateur de performance que pour la comparaison précédente (la distance horizontale). La comparaison a montré que les trois options sont viables, le scanner laser et le tapis de sol tactile nécessitant tous deux une distance de 2036 mm, tandis que le système de surveillance de l'espace de travail par projecteur ne nécessite qu'un espace de 1226 mm. Pour mieux comprendre

les différences entre les options, il est possible d'inclure également d'autres critères tels que le coût et la facilité d'utilisation. Pour évaluer les coûts, le nombre de capteurs et les dimensions du tapis de sol nécessaires pour protéger un seul robot ont également été pris en compte. Cela pourrait être fait dans la simulation par la modélisation du champ de vision des capteurs et la visualisation de la taille des distances de séparation requises. Deux scanners laser étaient nécessaires par robot, et trois systèmes de surveillance de l'espace de travail par projecteur étaient nécessaires. Le tapis de sol tactile a nécessité un espace de 2600 mm x 5800 mm, ce qui fait que l'espace total au sol pour les trois robots est plus petit en raison du chevauchement de ces zones pour les trois armoires.

Dans cet exemple, le concepteur a choisi des tapis de sol de sécurité, car ils répondent aux exigences globales et sont résistants aux conditions environnementales en raison de la possibilité de projection de liquides de nettoyage hors des armoires.

Les deux cas d'utilisation exemplaires décrits dans cette thèse ont été choisis pour mettre en évidence un certain nombre de problèmes. Le premier problème est la différence entre les calculs du pire cas effectués sur une feuille de calcul avec des estimations très approximatives et l'approche plus granulaire fournie par l'outil CAS intégré dans la simulation. Dans le cas du roobt dépalettiseur, la taille requise de la distance minimale de séparation requise autour du robot était inférieure de plus de 66 % en cas d'utilisation de barrières immatérielles et de plus de 55 % en cas d'utilisation d'un scanner laser en supposant que la vitesse du robot soit de 100 % POV. Les écarts dans ce cas proviennent de l'hypothèse que le robot se déplace en permanence à 100 % de sa vitesse. Alors que le programme du robot dans la simulation était réglé à 100 % pour tous les mouvements, en raison des rampes d'accélération et de décélération et des distances pour les différentes articulations, les différentes articulations ont rarement pu atteindre une vitesse aussi élevée. Cela devient évident lorsque l'on suppose une vitesse de robot plus lente de 33% POV comme dans la 4ème configuration testée. Dans ce cas, la surface calculée par l'outil CAS n'est inférieure que de 12,9 % au calcul le plus défavorable, ce qui indique une convergence entre la vitesse supposée et la vitesse simulée du robot.

Le deuxième problème mis en évidence par le cas d'utilisation dépalettisant est la capacité à effectuer des analyses de simulation avec des variations de différentes configurations physiques dans la disposition. Alors que les configurations 3 et 5 nécessitent la même surface au sol, les dimensions individuelles de la longueur et de la largeur peuvent être modifiées en changeant le positionnement des palettes par rapport au robot. Dans ce cas, les deux variantes s'adaptent, mais la différence de 30-40 cm par dimension pourrait offrir à un concepteur d'une autre application l'espace nécessaire pour que tout s'intègre dans son espace donné. Par ailleurs, si l'utilisation d'un autre type de robot n'offre aucun avantage, la simplicité avec laquelle le robot peut être échangé permet au concepteur d'essayer différents modèles pour trouver un optimum.

Dans les deux cas d'utilisation, les paramètres des capteurs étaient les mêmes. Cela masque l'un des avantages de l'approche proposée, à savoir que le concepteur dispose d'une bibliothèque de composants avec des configurations plausibles. Les fiches techniques des capteurs peuvent être confuses à comprendre et une valeur trop optimiste ou pessimiste pourrait être utilisée pour un paramètre spécifique du capteur (comme la portée requise ou la résolution). Cette approche réduit les erreurs et augmente la fidélité des enquêtes par rapport à leur mise en œuvre réelle.

Un autre avantage de cette approche est la concentration des informations numériques dans une zone, de sorte qu'elles soient réutilisables, non seulement pour

d'autres conceptions, mais aussi pour la phase de mise en œuvre. En effet, un grand avantage de cette approche apparaîtra lorsque le modèle numérique sera d'une fidélité suffisante pour que l'on puisse prouver que les résultats de la simulation coïncident avec les mesures du système mis en œuvre. Dans ce cas, il serait possible de remplir les exigences relatives à la validation du système pour obtenir la marque CE.

Enfin, les résultats de la thèse sont comparés aux exigences spécifiées dans la section 3. Les sept principales exigences sont les suivantes :

- Suivi des exigences dans le processus de conception pour vérifier si les composants utilisés dans la conception répondent aux exigences spécifiées et explicitement formulées selon les normes ISO 10218-1, -2 et ISO-TS 15066
- Soutien aux analyses de simulation couvrant tous les aspects d'une application HRC pour améliorer/optimiser les conceptions.
- Quelles sont les mesures d'atténuation des risques possibles et valables ?
- Quelle est la distance minimale de séparation requise pour un capteur de sécurité spécifique et pour la configuration de système choisie ? Comment les paramètres de processus (par exemple, la charge utile, le programme du robot, les vitesses) que le concepteur peut modifier influencent-ils leur taille ?
- Tous les paramètres du capteur et du robot choisis (par exemple, la résolution du capteur, le temps de réaction du capteur, etc.) répondent-ils aux exigences de l'application spécifique ?
- Soutenir l'intégration avec les outils d'ingénierie existants tels que les logiciels de CAD et de simulation afin que les informations numériques sur le système soient réunies en un seul endroit (par exemple pour soutenir la documentation, la mise en œuvre dans le monde réel),
- Permettre des résultats vérifiables et certifiables.

Toutes les exigences relatives à la méthodologie proposée ont été remplies. Une réserve subsiste, notamment en ce qui concerne l'exigence selon laquelle la méthode doit montrer quelles mesures d'atténuation des risques sont possibles et valables. L'outil CAS tel que développé ne contient qu'une bibliothèque initiale de capteurs de sécurité génériques et spécifiques qui peuvent être utilisés. La bibliothèque n'est pas exhaustive et peut être mise à jour dans le cadre de travaux futurs. La possibilité de visualiser le champ de vision de chaque capteur et de le comparer à la distance minimale de séparation requise à surveiller simplifie la tâche du concepteur. Cette comparaison est actuellement effectuée par le concepteur à l'aide de l'outil CAS. Les futures versions du CAS Tool pourront intégrer des algorithmes permettant de vérifier automatiquement cette comparaison.

6 Use-case studies with exemplary applications

The approach presented in this work has been implemented for use as a plug-in to be used with the commercially available robot simulation software, Visual Components. The software was implemented according to the overall architecture described previously and with clear specification of where the data for the component models comes from (e.g. from the component data sheet, the simulation tool, or the user). The work in this thesis is focused on the approach and methodology for better considering the safety-related aspects when designing HRC applications and not on the software implementation. The software used in Section 6 was implemented by colleagues from the Business Unit Robotic Systems at the Fraunhofer IFF and in accordance with the concepts and architecture developed in this thesis.

All mechanical and robot information was extracted from the simulation environment and fed into the component models in the CAS Tool. This includes robot joint speeds, joint configurations, kinematic specification, and payload information. It also includes physical characteristics such as the position of all components. In many cases, the attributes required for calculation of minimum separation distance were not available from the simulation. This includes information such as sensor configuration and other parameters about the robot such as maximum payload or maximum robot reach. In these situations, the user was asked to initially populate the missing attributes in the models. This data was then stored in an internal database with component model information, so that this only needed to be done once.

The applications described in Section 5 were simulated in Visual Components and the CAS-Tool as described in Section 3 and 4. The CAS-Tool was used for determining the size of the minimum separation distance and for determining other safety-related information.

The author would like to note that the focus of this work is on the methodology for designing HRC applications. This includes the architecture of the model-based systems engineering approach and the specification of the component models. The actual software implementation was not a part of my own work. It was developed in close collaboration with colleagues from the Fraunhofer IFF. The comparison of the results of the approach with traditional methods is nevertheless an important aspect of this work and will be used here to show the advantages of the approach described here.

6.1 Use-case Depalletizing robot

Figure 26 shows three screenshots from the simulation of the robot and sensor system with the aim of supporting the designer with supplemental information about the safety. The application with a robot, 2 pallets (on the right of the robot) and the table (on the left of the robot) is shown for three discreet positions during the robot's program. The red polygon indicates the instantaneous size of the minimum required separation distance, and the yellow polygon (which is the same in all three images) represents the total separation distance over the entire program. In the simulation, the user can see how the red area moves with the robot for each step of the simulation.

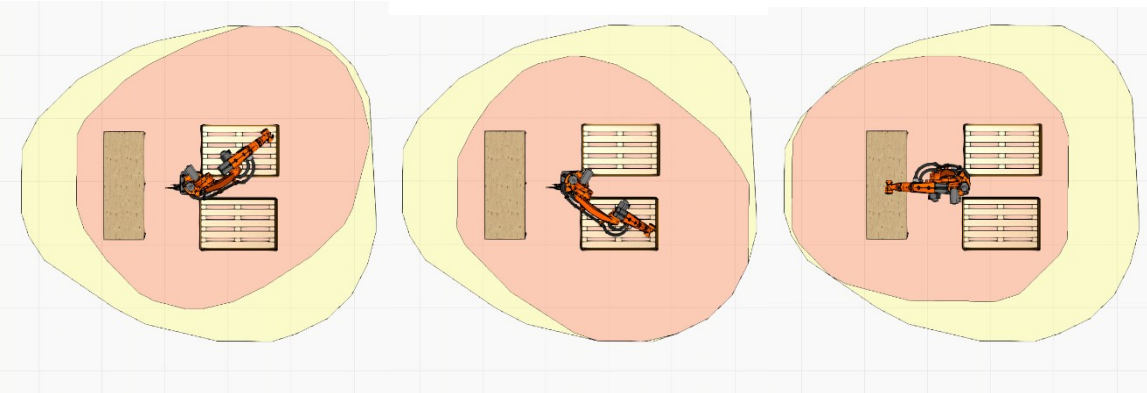


Figure 26: Screenshots from simulation of robot in application. The yellow polygon represents the minimum required separation distance over the entire programmed path. The red polygon represents the instantaneous required separation distance based on the robot's current speed at that moment along the programmed path.

A first comparison relates to the separation distance calculated by a spreadsheet (using worst-case assumptions as described in Section 5.1.2.7) and the distance as calculated using the robot's programmed speeds. Here we see that the minimum separation distance calculated using our approach is significantly smaller than with traditional methods based on simple worst-case assumptions.

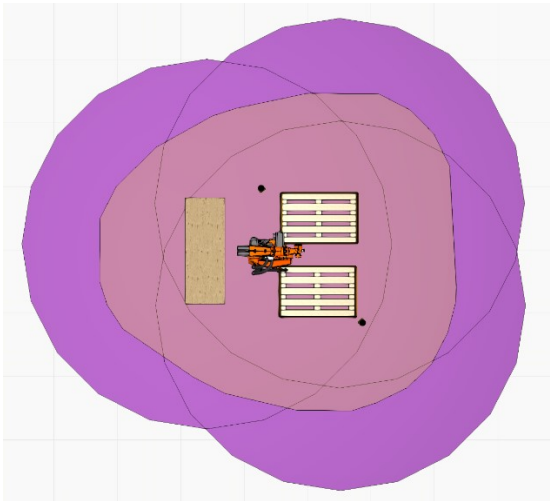


Figure 27: Comparison on separation distance calculated with worst-case assumptions (purple) as three circles at furthest reaching positions and with our approach (yellow polygon) for KR 22 robot with 100% POV and with laser scanner as safety sensor.

While the required separation area around the robot for the given program is much smaller than the worst-case calculation, the designer can easily see (Figure 28) that it still does not fulfil their design goals of fitting within the existing work cell without intruding into the logistic areas (where forklifts drive). This means the designer needs to make more changes in order to reach their design goals of getting the system to fit in the existing floor space.

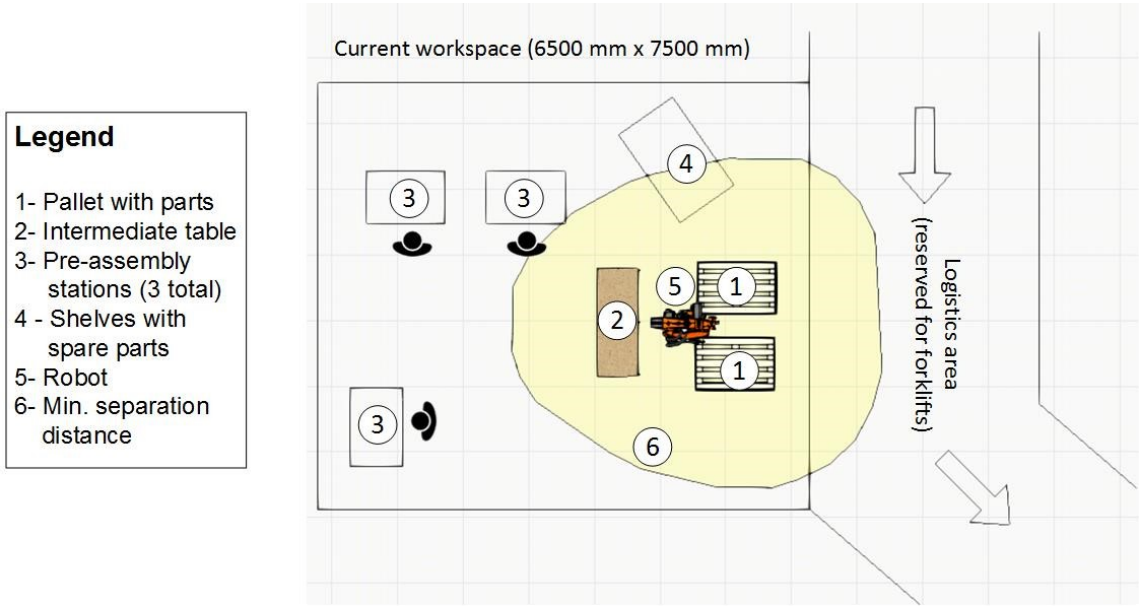


Figure 28: Required separation distance (yellow polygon) around the robot based on a simple program with robot moving to three extreme positions to empty the two pallets and place the parts on the table.

As we stated earlier, the entire application can be viewed as a single system with various parameters available for change in order to reach specific design targets. In this case, the designer would like for a solution to fit into the existing workspace, so as not to disrupt other processes such as the forklift driving areas and the work of the operators on assembly tables 1, 2, and 3. Options the designer can consider include:

- Changes to the robot:
  - type (with different braking parameters, payload, reach, etc.)
  - program
    - speed (either for the entire process, or for specific motions)
    - changes to limit the max extension
- Changing the sensor
  - type (with associated reaction time and overreach values)
  - parameters for use (such as resolution)
- Changing the process and layout
  - This includes adding fencing elements, the position of components in the workspace, or changes to the task itself (e.g the order of tasks, the responsibility for a task)

Using traditional methods, the designer would need to maintain a number of separate documents including a CAD layout, spreadsheets of safety distances and a library of pdf files with robot and sensor data. Using our approach, the designer can try all of the various options listed above from within the robot simulation environment and can immediately see the effects of any specific changes. Only valid system configurations can be used due to the internal models, and the component data is saved in a database and readily available for further use. In order to illustrate this, we have carried out a series analyses with four configurations (Table 22).

Table 22: System parameters for four test configurations

System configuration parameters	Configuration 1	Configuration 2	Configuration 3	Configuration 4
Robot Parameters				
POV [%]	100 %	100 %	100 %	33 %
Payload [%]	66 %	66 %	66 %	66 %
Sensor Parameters				
Type	Laser scanner	Light curtain	Light curtain	Laser scanner
Resolution [mm]	70 mm	40 mm	20 mm	70 mm
Reaction time [ms]	90 ms	21 ms	30 ms	90 ms
C-value [mm]	850 mm	210 mm	50 mm	850 mm
Z-value [mm]	0 mm	0 mm	0 mm	0 mm

The analysis results in terms of cycle time and required floor space are shown in Table 23.

Table 23: Cycle time and size of separation distance around robot for four tested configurations

	Cycle time [s]	Required separation area (worst-case calculation) [m²]	Required separation area (calculated by CAS Tool) [m²]
Configuration 1	11 s	50.72 m²	22.74 m²
Configuration 2	11 s	35.67 m²	11.22 m²
Configuration 3	11 s	28.54 m²	9.52 m²
Configuration 4	19 s	25.99 m²	22.65 m²

A final solution favored by the designer could look like the layout shown in Figure 29. Two light curtains are used to stop the robot when operators approach from the left or forklifts come in from the right. The complete system fits within the existing workspace. Interestingly, changes to the robot speed in this configuration have little effect on the overall size of the workspace. The choice of sensor and its configuration (high resolution, lower C-Value) play the largest role in reducing the size of the required separation distance.

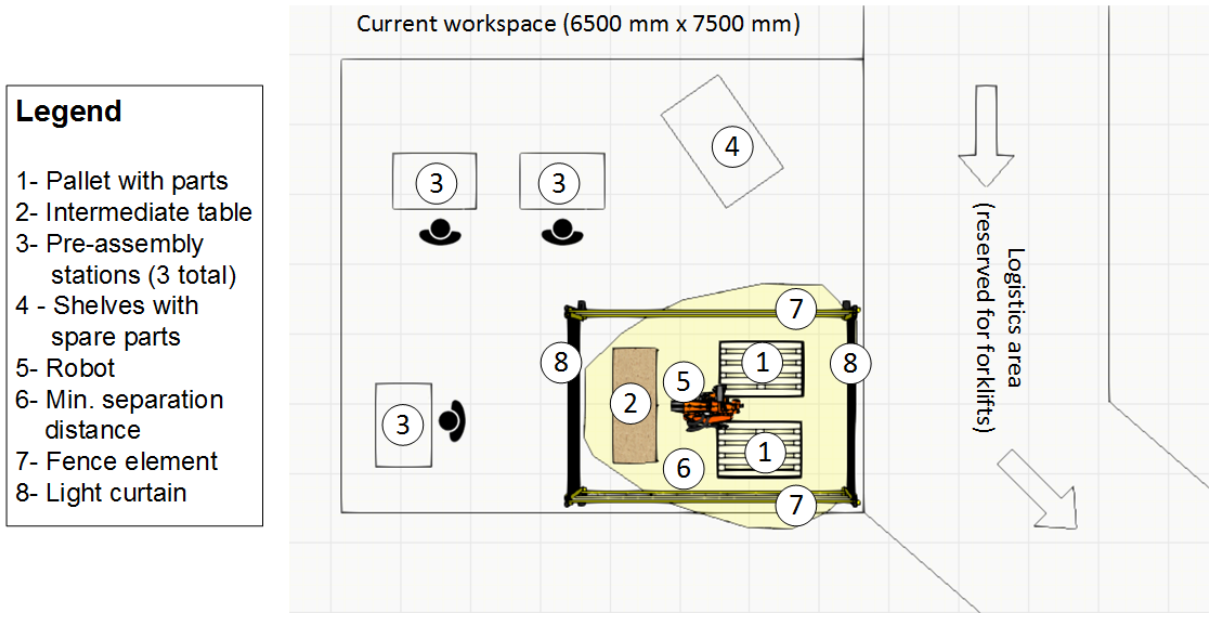


Figure 29: Final selected configuration with a KR22 robot, 2 light curtains, and 2 fences to limit access to the pallets and the table.

It should be noted that the four configurations for this analysis only represent a small number of the options available to the designer. Another design option that the designer can quickly test are whether changes to the layout to decrease the robot extension and indirectly improve braking time and distance would make a difference to the time and floor space needed (Figure 30).

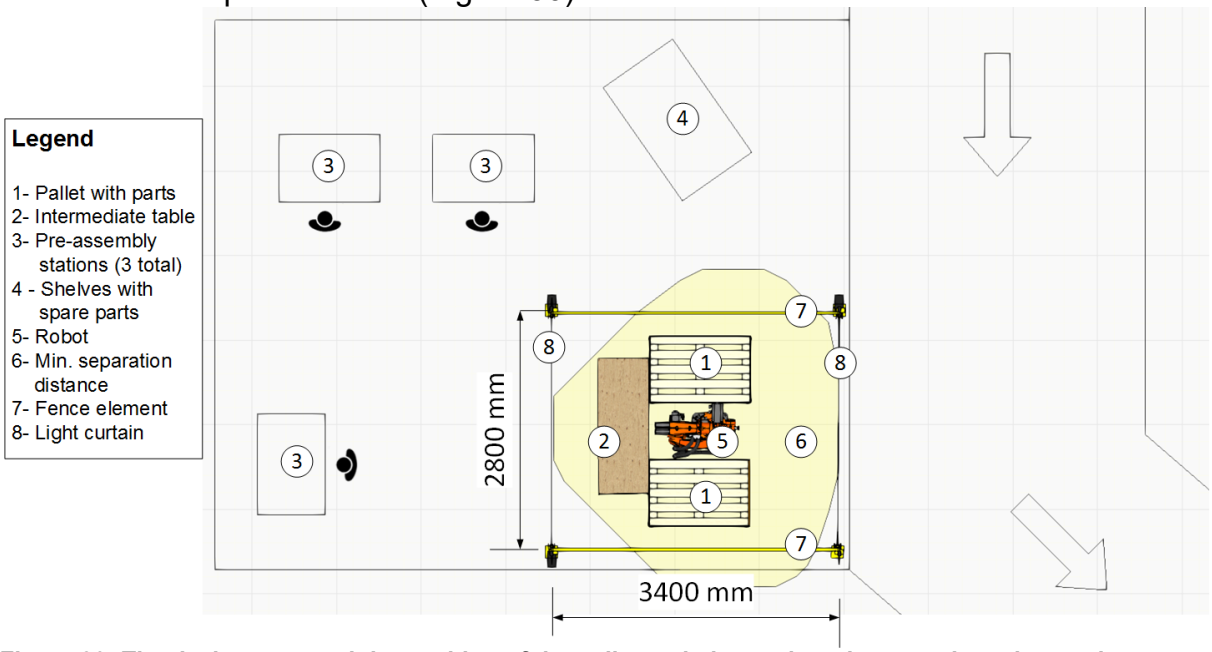


Figure 30: The designer moved the position of the pallets relative to the robot to reduce the maximum required robot extension.

The designer can also quickly replace the robot type with an aim towards using one with better braking parameters. The library of components contains all the relevant information for different robots and all calculations can be executed with different models.

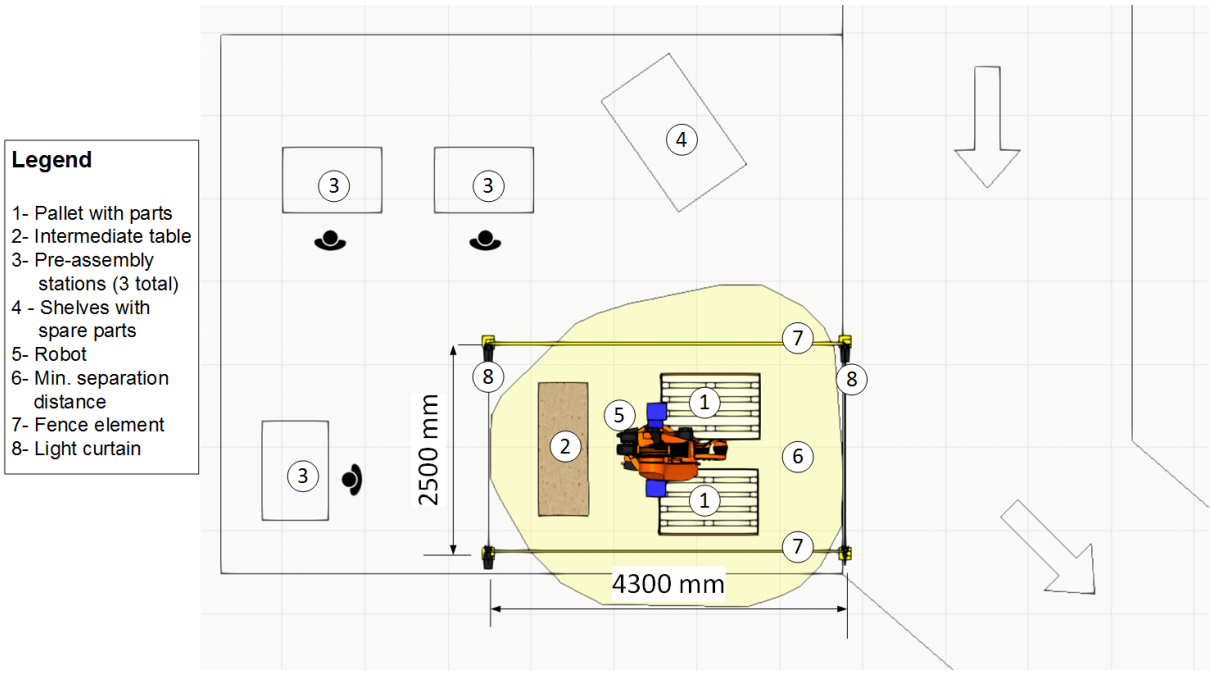


Figure 31: The designer used the layout from Figure 29 with a KUKA KR 60 robot instead of the KUKA KR22.

Table 24 shows the results of the comparison for cycle time and required floor space for the additional what-if analyses. The change to the layout does not result in a change in the cycle time or in the size of the required floor space. The main outer dimensions are slightly changed and this can be of use to the designer when trying to meet other design constraints. The use of a larger robot for the application did not offer any significant improvements. On the contrary, the larger robot is slower (even though it was also programmed to move at 100% POV) and requires slightly larger separation distance.

Table 24: Comparison of three what-if analyses for depalletizing application

	Cycle time [s]	**Cell width [m]**	**Cell length [m]**	**Required floor space (based on CAS Tool calculations) [m²]**
Configuration 3 (from Table 22)	11 s			
Configuration 5: KR22 layout changed	11 s			
Configuration 6: KR60 robot instead of KR22 from Configuration 3	24 s			

It should also be noted that the cycle times presented in these results are not indicative of the complete process, as the program simply contains the robot moving to the three furthers positions. However, they do show a trend for comparison between different configurations, layouts, and robot types.

6.2 Use-case cleaning machined parts

Figure 32 shows a screenshot from the simulation of the cleaning machined parts use-case described Section 0. The application features three cleaning stations, each with its own passively mobile lightweight robot that can be rolled into position by the operator. For simplification, only one robot is shown in the figure. The simulation used simple robot program, defined using two discreet positions that represent the furthest reach within the cabinets and a third position (robot parked). This program corresponds to the analysis for the size of the minimum separation distance using traditional methods. The yellow polygon in the figure represents the total separation distance over the entire program. In this analysis, we compare the results from traditional methods to the CAS Tool.

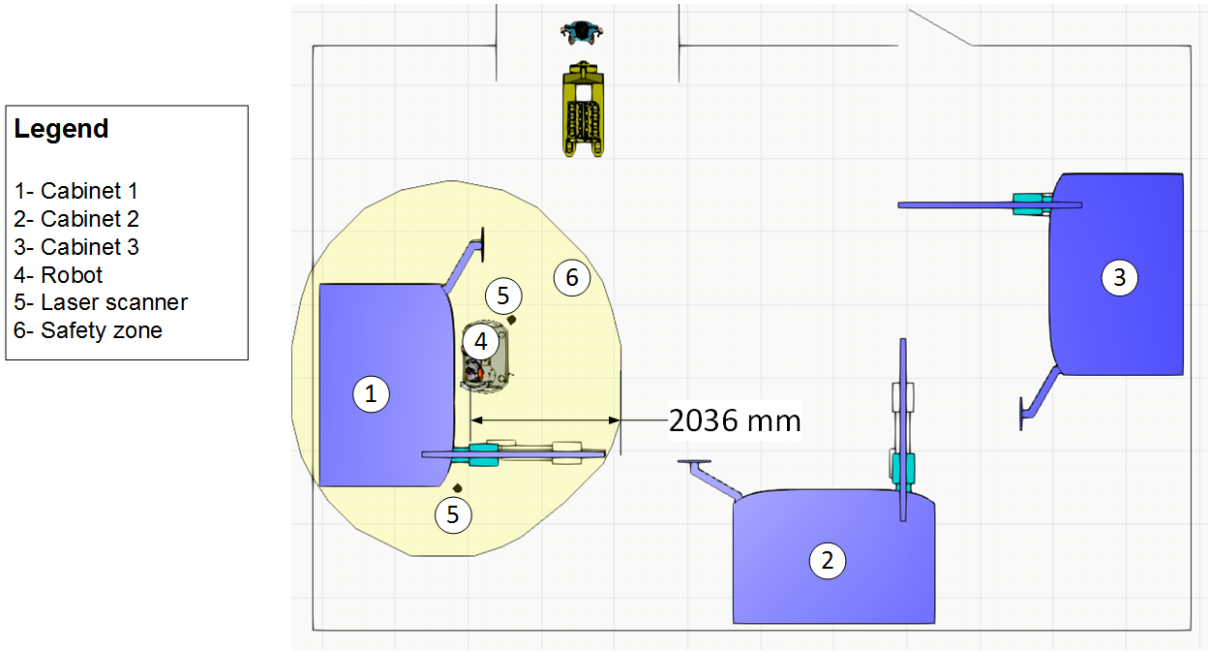


Figure 32: Required separation distance (yellow polygon) around the robot based on a simple program with robot moving to two extreme positions in the cabinet and ending in its parked position for a laser scanner and 100% POV robot speed.

As was already stated in the traditional analysis of this use-case, the overall area required for the minimum separation distance is not a useful metric, since this area is partially covered by the cleaning cabinet and does not need to be safeguarded. A preferred metric for comparing the size of the required separation distance is the horizontal distance from the center of the robot TCP to the furthest edge of the safety zone.

Table 25: System parameters for comparison of traditional methods and CAS Tool

System configuration parameters	**Configuration**
Robot Parameters	
POV [%]	100 %
Payload [%]	66 %
Sensor Parameters	
Type	Laser scanner
Resolution [mm]	70 mm
Reaction time [ms]	90 ms
C-value [mm]	850 mm
Z-value [mm]	0 mm

The analysis results in terms of horizontal distance between the robot and the furthest edge of the minimum required separation distance are shown in Table 26. This distance as calculated with the CAS Tool is over 20% less than with traditional methods.

Table 26: Horizontal distance between robot and furthest edge of minimum required separation distance for the tested configuration with traditional method and CAS Tool

	Horizontal distance (worst-case calculation) [mm]	**Horizontal distance (calculated by CAS Tool) [mm]**
Configuration 1	2665 mm	2036 mm

The results from this analysis were then duplicated for the robots working at the other two stations and the pathways necessary to access the individual cabinets were overlaid on this layout. The overlap between the pathways to access a cabinet and the minimum required separation distance for the neighboring cabinets is quite low. Either the operator can make an effort to adjust their path or the robot can move slower (through application of SSM principles), effectively reducing the size of the minimum required separation distance.

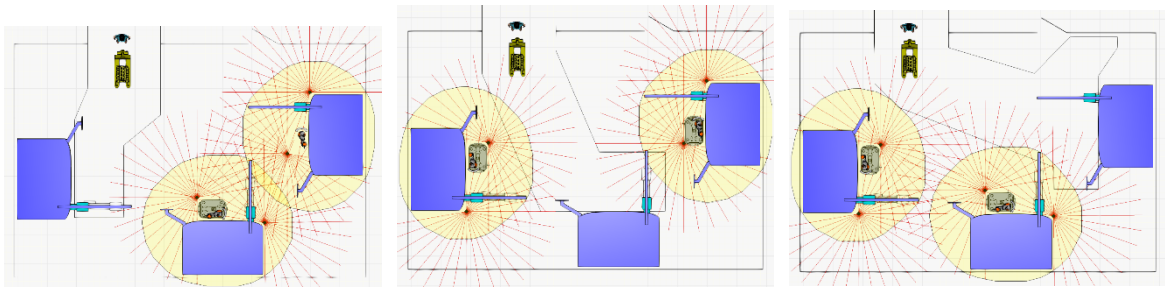


Figure 33: Pathway to access station 1, 2 and 3 with laser scanners as the safeguarding sensor and 100% POV robot speed

The CAS Tool was also used to test the effects of different types of safety sensors on the application. Figure 34 and Figure 35 show the same use-case with projector-based workspace monitoring systems and tactile floor mats. Using the same performance indicator as for the previous comparison, it is possible to compare the usage of different types of sensor.

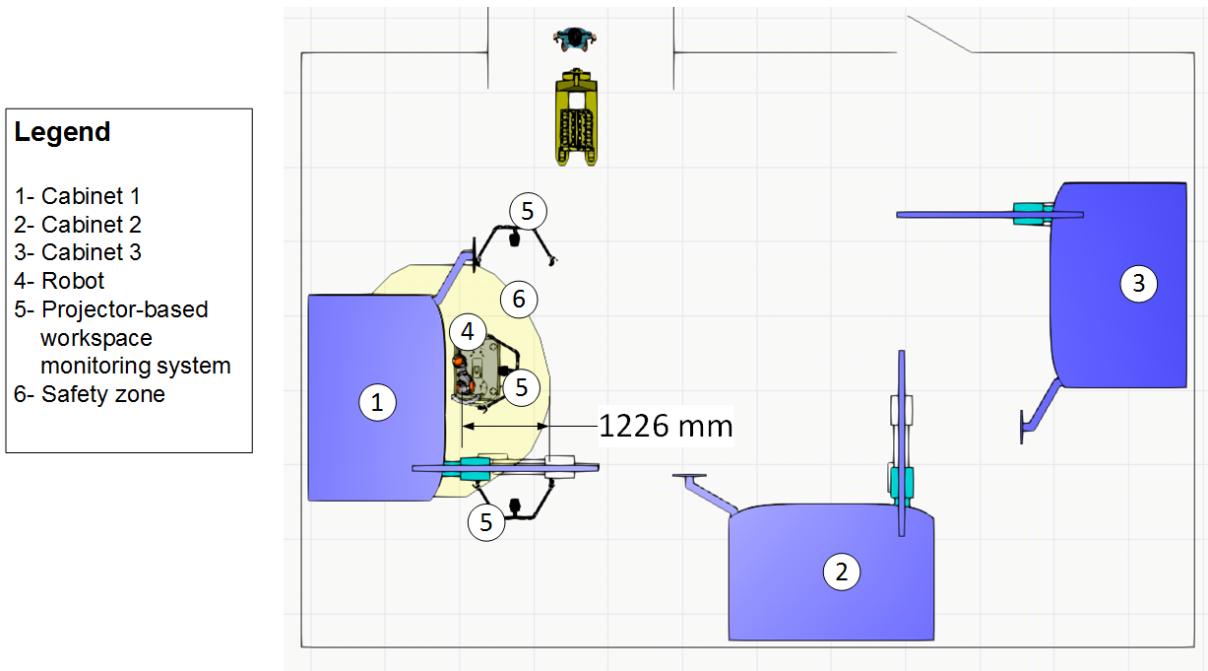


Figure 34: Required separation distance (yellow polygon) around the robot based on a simple program with robot moving to two extreme positions in the cabinet and ending in its parked position for the IFF projector based workspace monitoring system and 100% POV robot speed.

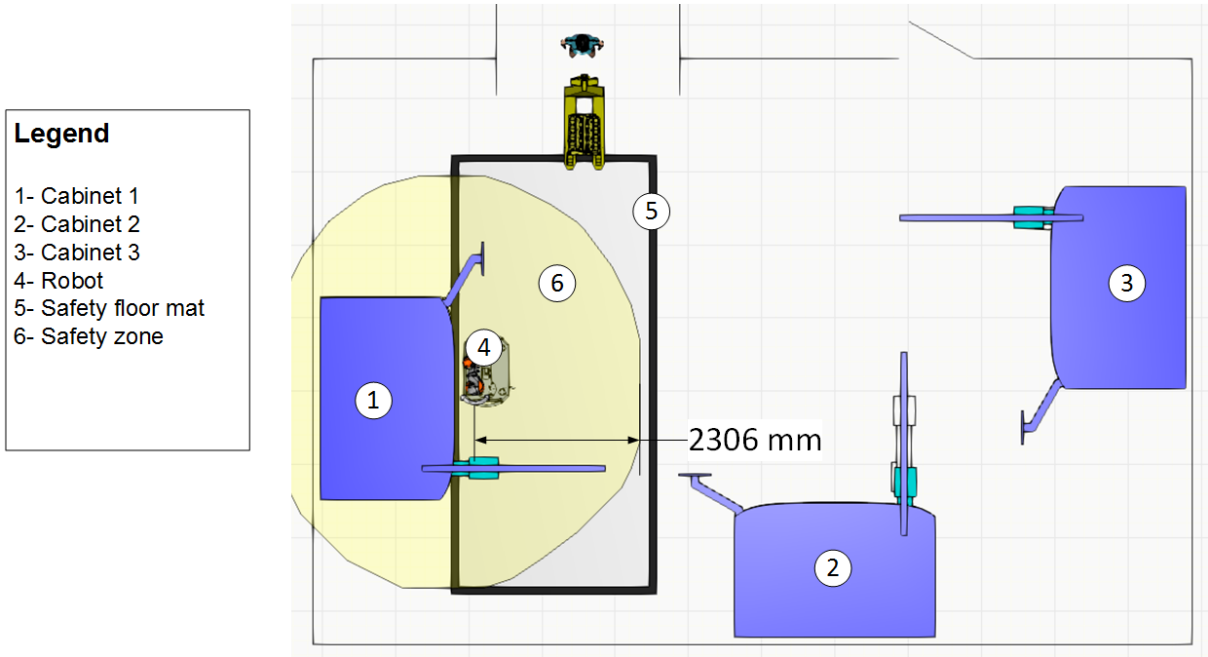


Figure 35: Required separation distance (yellow polygon) around the robot based on a simple program with robot moving to two extreme positions in the cabinet and ending in its parked position for the IFF tactile floor mat and 100% POV robot speed

While the size of the minimum separation distance in Figure 34 is much smaller than with the other two sensor systems, the designer needs to use three projection systems in order to oversee the entire area. Although this sensor system is not yet a product and there is no binding information regarding costs, the designer can estimate that the cost for the solution in Figure 29 will be higher than for the other options.

Table 27: Comparison of three different safety sensors calculated with CAS Tool for cleaning machined parts use-case

	Horizontal distance (calculated by CAS Tool) [mm	**Number of sensors (or dimensions of floor mat) needed per robot to monitor separation distance**
Laser scanner	2036 mm	2 sensors
Projector-based workspace monitoring system	1226 mm	3 sensors
Tactile floor mat	2306 mm	2600 mm x 5800 mm

Figure 36 and Figure 37 show the pathways to access the three stations for each of the three sensors evaluated. The designer can quickly see that all three options are viable, and then other criteria such as cost and ease of use should be considered.

Figure 36: Pathway to access station 1, 2 and 3 with projector-based workspace monitoring systems as the safeguarding sensor and 100% POV robot speed

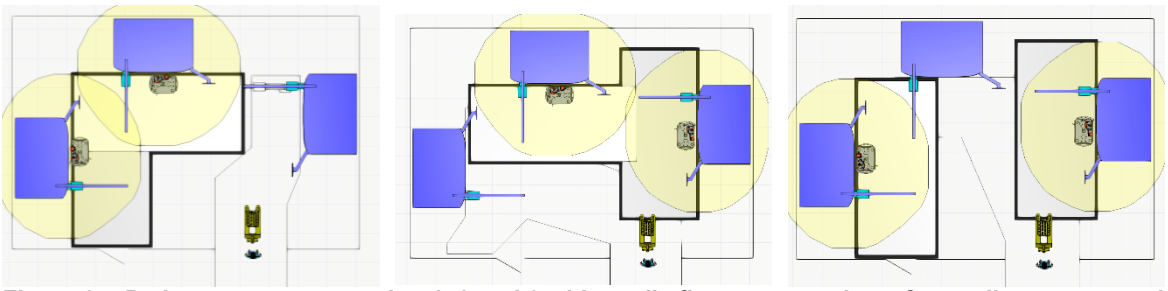


Figure 37: Pathway to access station 1, 2 and 3 with tactile floor mats as the safeguarding sensor and 100% POV robot speed

Given the environment and the process (cleaning fluids, aerosols), the designer initially chooses the tactile floor variation (Figure 38) due to its robustness against these environmental factors. The ability to detect the speed and direction of approaching humans will support the proper implementation of SSM, so that the robot will only need to slow down when the approaching human is moving towards the robot and not parallel to it (i.e. to get to the neighboring station).

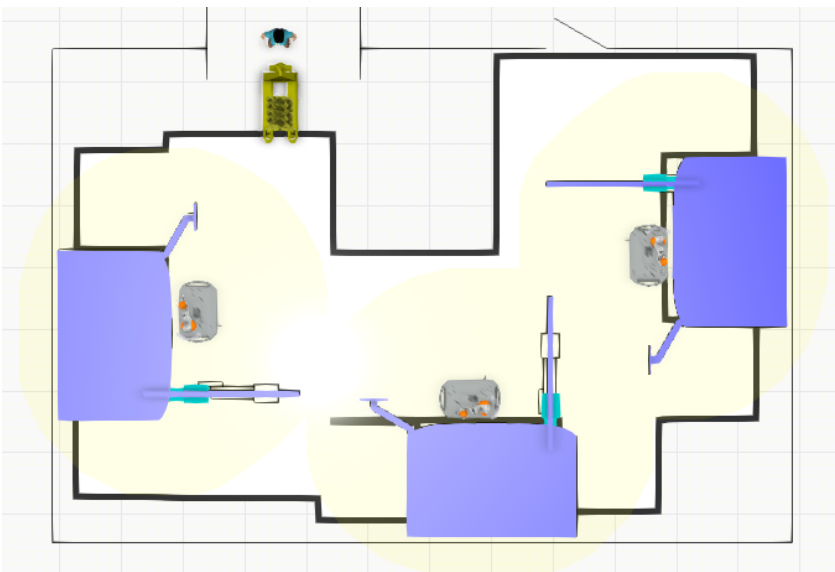


Figure 38: Suggested layout with tactile floor mats safeguarding all robots.

6.3 Discussion of proposed method

The method for designing safe HRC applications described in this thesis aims to support the designer of new and existing applications by making information about safety available prior to the final commissioning. The system consisting of a collaborative robot with tooling and parts, a process, working in a specific environment is very complex and the interdependencies of all the system components are not always understandable or readily traceable. Such a complex system offers many opportunities for adaption of individual or multiple parameters and components to meet specific design goals. Therefore, an important advantage of the proposed method is the possibility to perform “what-if” analyses of the application in a simulation, considering safety-related aspects such as the required size of the minimum separation distance. This was possible through the use of models that were developed to relate to the safety decisions that are made during the normal design process. Parameters that contribute to this include sensor choice, sensor configuration and parameterization, the robot program (speeds and trajectories), and other layout considerations.

The two exemplary use-cases described in in this thesis were chosen to highlight a number of issues. The first issue is the difference between the worst-case calculations carried out on a spreadsheet with very rough estimates versus the more granular approach provided through the CAS tool integrated in the simulation. In the case of the depalletizing robot, the required size of the minimum required separation distance around the robot was over 66% less when using light curtains and over 55% less when using a laser scanner when assuming that the robot speed is at 100% POV. The discrepancies in this case come through the assumption that the robot will move all the time at 100% POV. While the robot program in the simulation was set to 100% for all movements, due to the acceleration and deceleration ramps and the distances for the individual joints, the individual joints were seldom able to reach such a high speed. This becomes evident when assuming a slower robot speed of 33% POV as in the 4th configuration tested and listed in Table 22. In this case, the area as calculated by the CAS tool is only 12.9% less than the worst-case calculation, indicating a convergence between the assumed and simulated speed of the robot.

The second issue highlighted by the depalletizing use-case is the ability to carry out what-if analyses with variations of different physical configurations in the layout. While both configuration 3 and 5 required the same floor space, the individual dimensions for the length and width could be varied through changes to the positioning of the pallets relative to the robot. In this case both variations fit, but the difference of 30-40 cm per dimension could offer a designer of another application the necessary space to make everything fit into their given space. In addition, while the use of another robot type did not offer any advantages, the simplicity with which the robot can be exchanged allows the designer to try various models to find an optimum.

The cleaning cabinet use-case highlights on one hand the issue that designers often need to iterate with regards to the safeguarding mode due to assumptions made during the risk analysis. The lightweight robot was initially chosen with the idea that it would be safeguarded through power and force limiting. This would allow operators to pass close to the robot when loading and unloading neighboring station without requiring the robot to stop or slow down. However, the safety expert identified that an operator could come in very close to the nozzle and the robot with their head to monitor cleaning progress or investigate any non-nominal situation (e.g. the operator hears a strange sound or sees fluid spraying uncontrolled due to some kind of error) and that this situation needs to be avoided. Given the spatial constraints (the room cannot be

made larger) and the process constraints (e.g. 3 machines in the room), it is an extreme challenge to find a cost effective solution that allows for parallel operation of all robots and manual loading/unloading tasks.

Another issue highlighted by the cleaning cabinet use-case is how the designer can flexibly choose their design KPI and trace it over multiple variations rather than just looking at one calculation (e.g. the area of the size of the safety zone). For the cleaning cabinet application, the relevant metrics were the relation of the size of the safety zone to the pathways to access the neighboring stations, as well as the size of the safety zone in the direction behind the robot. The calculation of this distance was 25% less with the CAS Tool than with the worst-case manual calculation. Furthermore, the designer could not only try alternative sensors by drag and drop, but the number of sensors required to safeguard the areas could be easily determined.

In both use-cases, the parameters for the sensors were the same. This masks one of the advantages of the proposed approach, namely that the designer has a library of components *with plausible configurations* at their disposal. Sensor data sheets can be confusing to understand and an overly optimistic or pessimistic value could be used for a specific sensor parameter (such as range required or resolution). This approach reduces errors and increases the fidelity of the investigations with respect to their real implementation.

A further advantage of the approach is how digital information is concentrated in one area so that it is reusable, not only for other designs, but also for the implementation phase. Indeed, a great advantage to this approach will become apparent when the digital model is of sufficient fidelity that simulation results can be proven to coincide with measurements of the implemented system. In this case it would be possible fulfil the requirements on the system validation to obtain the CE Mark. The currently running EU project COVR [43] is currently putting lots of effort into supporting the robotics community in their validation measurements. Rendering these measurements obsolete through the proposed approach would be a great support to roboticists worldwide. Table 28 shows the software that different responsible persons use along the design process (as described in Section 3 and Section 2.2). This represents a contrast to the software used traditionally and listed in Table 3, specifically for steps 3, 5, 7 and 8. After step 3, the designer is almost exclusively using the CAD/simulation with CAS Tool for all tasks and calculations. The other software is focused on documentation (e.g. for CE Mark purposes) and not for the design and development process. According to the current methods for obtaining the CE Mark (Figure 4), the phases of hazard identification and risk evaluation still require documentation and expert evaluations that safety experts are held responsible for. These tasks were for this reason outside the scope of this thesis. Nevertheless, as the research of the state of the art has shown, there is already other research looking at methods to automate the risk analysis, and it is plausible to incorporate their work into this overall workflow in the future.

Table 28: Overview of software used by responsible role during the design phases of an HRC application in manufacturing according to the proposed methodology

Step n°	Design phase according to MD 2006/42/EC	Responsible role	Software used
1	Starting point: General idea of collaborative application	Designer	Document, CAD/Simulation
2	Safety oriented design	Designer	Document (checklist)
3	General and essential requirements	Designer	**CAD/simulation with CAS Tool** / Document (checklist)
4	Model process, assign tasks	Designer Safety Expert	Document/Spreadsheet
5	Define system limits and requirements	Designer	**CAD/Simulation with CAS Tool**
6	Hazard identification and risk evaluation	Designer Safety Expert	Document / Spreadsheet/ other safety evaluation tool CAD/Simulation
7	Hazard elimination and risk mitigation	Safety Expert Designer	Document / other safety evaluation tool **CAD/Simulation with CAS Tool**
8	Review	Designer	**CAD/Simulation with CAS Tool**

In accordance with the overall approach of this thesis as described in Figure 7, the developed models and approach have been validated against the requirements specified in Section 3.1. Table 29 shows the results of the validation.

Table 29: Validation of final thesis results against the requirements on the methodology for design of safe HRC applications

n°	Requirement	Achieved? How?
1	Requirements tracking in the design process to check whether the components used in the design fulfill the specified and explicitly formulated requirements according to the standards ISO 10218-1, -2 and ISO-TS 15066	**Achieved.** Requirements tracking for the stereotypes “performanceRequirement”, “physical-Requirement” and “designConstraint” are possible through transposition of ISO/TS 15066 into SysML requirements diagram and their subsequent mapping to specific component attributes. Requirements tracking was **not achieved** for requirements of type “functional-Requirements” that are satisfied by behaviors and that are currently not modeled in CAD/simulation. This can be the focus of future work.
2	Support for what-if analyses covering all aspects of an HRC application to improve/ optimize designs.	**Achieved.** The combination of CAD/simulation with the CAS Tool allows for designers to investigate changes to the environment, the parts, the robot, the sensors, and all of their configurations, as demonstrated in the examples in Sections 6.1 and 6.2.
3	What risk mitigation measures are possible and valid?	**Achieved – to a certain extent.** The CAS Tool contains an initial library of generic and specific safety sensors that can be used as risk mitigation measures. The library is not exhaustive and can be updated in future work. The ability to visualize the field of view of the individual sensors and to cross-reference this against the required minimum separation distance that should be monitored simplifies this task for the designer. Future versions of the CAS Tool can integrate algorithms to automatically verify this.
4	What is the required minimum separation distance for a specific safety sensor and for the chosen system configuration? How do	**Achieved**. The CAS Tool calculates and visualizes the minimum separation distance based on the configurations of the robot (e.g. its program, payload, extension) and the

	process parameters (e.g. payload, robot program, speeds) that the designer can change influence their size?	sensors. The designers can easily vary individual parameters within allowable, meaningful limits to calculate the influence of specific changes on the overall size of the safety zone.
5	Do all chosen sensor and robot parameters (e.g. sensor resolution, sensor reaction time, etc.) fulfill the requirements of the specific application?	**Achieved.** The internal models of the specific, non-generic sensors validate that only meaningful set of configuration parameters can be chosen. Furthermore, the designer can
6	Support integration with existing engineering tools such as CAD and simulation software so that digital information about the system is in one place (e.g. to support documentation, real world implementation),	**Achieved.** The software implementation from the Fraunhofer IFF based on the design specifications from this thesis was successfully integrated with the simulation software “Visual Components”. The test results were shown in Sections 6.1 and 6.2.
7	Allow for verifiable and certifiable results.	**Achieved.** While the results of the current CAS Tool implementation are not yet certified, this would be possible after a validation of the results with a real world application. In particular, the models for determining the braking distance and time of the robot system need to be validated against a real robot system. In this case, the results of the CAS Tool would only be valid for specific robot models that have undergone this testing.

Résumé Chapitre 7 – Conclusion

Dans cette thèse, une méthode permettant de prendre en compte les aspects de sécurité des applications HRC pendant la phase de conception a été développée et présentée. Ces applications HRC se caractérisent par leur complexité due aux interdépendances entre les différents paramètres du système (choix du matériel, paramétrage du système comme la portée des capteurs ou la vitesse du robot) et peuvent être considérées comme des systèmes de systèmes. Un examen de la pratique actuelle en matière de planification et de conception des applications HRC, en mettant l'accent sur les aspects liés à la sécurité, a été réalisé afin de déterminer où se situent les défis actuels du point de vue du concepteur. Un examen supplémentaire de l'état de l'art de l'utilisation des principes et des méthodes d'ingénierie des systèmes pour la robotique et la sécurité a révélé un grand nombre d'efforts passés axés sur le génie logiciel, l'analyse des dangers et des risques et le développement ontologique (qui sous-tend tout ce travail). Néanmoins, il n'y a eu jusqu'à présent aucun effort axé sur les problèmes auxquels le concepteur est confronté en ce qui concerne le choix et la paramétrisation des composants pour les applications HRC comportant un contrôle de la vitesse et de la séparation.

Des exigences relatives à une nouvelle approche de la prise en compte de la sécurité basée sur les besoins du concepteur et de l'expert en sécurité ont été formulées et ont guidé les processus de spécification de l'architecture et de modélisation des composants qui sont au centre de cette thèse. En particulier, des modèles réutilisables de capteurs et de robots sont nécessaires afin de disposer d'informations numériques sur l'application en un seul endroit. Cela permet de faciliter l'utilisation et l'exécution des analyses de simulation qui font partie du processus de conception. Une architecture qui utilise les outils de CAD/simulation existants que les concepteurs utilisent actuellement a été spécifiée. L'approche a été mise en œuvre sous la forme d'un outil de sécurité assisté par ordinateur (CAS Tool) et les résultats de la conception ont été comparés aux méthodes traditionnelles avec deux cas d'utilisation exemplaires. Les cas d'utilisation sont adaptés de cas d'utilisation réels de l'industrie automobile.

Grâce au CAS Tool, il a été possible d'apporter rapidement des modifications à la conception, en particulier aux composants tels que le robot et les capteurs de sécurité. L'emplacement des composants et le processus lui-même (c'est-à-dire la vitesse du robot) peuvent être facilement adaptés pour atteindre des objectifs de conception spécifiques tels que le temps de cycle minimum ou l'encombrement minimum de l'application. Le résultat final a été validé par rapport à l'ensemble des exigences de conception initiales, qui ont été respectées. Dans les applications exemplaires décrites dans cette thèse, la surface de la zone de sécurité requise autour d'un robot spécifique pourrait être réduite jusqu'à 66% en utilisant l'outil CAS basé sur cette méthodologie par rapport aux méthodes actuelles de calcul du pire cas, grâce à une base de données plus granulaire pour les calculs.

La méthodologie globale de modélisation des attributs de sécurité du système complet peut avoir de vastes conséquences pour les futures applications robotiques qui sont flexibles et qui prévoient des modifications en ligne du programme pendant l'exécution. Actuellement, une validation du système complet est nécessaire après sa mise en service afin de s'assurer que les effets de sécurité souhaités (par exemple, le maintien d'une distance de séparation sûre) sont effectivement obtenus par le système. Si la simulation des performances du robot en ce qui concerne le temps et la distance de freinage est correcte, la simulation serait suffisante pour valider le système. Cette approche pourrait donc contribuer à la validation de la performance globale du système. En outre, dans le cas d'un nouveau programme de robot, il

serait également possible de calculer en ligne la distance de protection minimale requise pour la sécurité. En supposant que les paramètres du système soient connus, cela pourrait offrir un moyen simple de valider que la sécurité est respectée pour les mouvements robotiques qui ne sont pas complètement préplanifiés, mais plutôt générés en temps réel. En ce sens, peu importe qui a créé les nouvelles commandes de mouvement, qu'il s'agisse d'un agent basé sur l'IA ou d'un programmeur humain. Le système vérifie simplement quelle est la distance minimale de séparation requise en fonction des mouvements planifiés et des paramètres des capteurs, puis vérifie que ces zones de sécurité ne sont pas violées.

Les travaux futurs se concentreront sur l'amélioration de la mise en œuvre des outils logiciels et sur le support à utiliser avec d'autres logiciels de CAD/simulation.

7 Conclusion

In this thesis, a method for considering the safety aspects of HRC applications during the design phase was developed and presented. Such HRC applications are characterized by their complexity due to the interdependencies between different system parameters (choice of hardware, parameterization of the system such as sensor range or robot speed) and can be considered system of systems. A review of the current practice for planning and designing HRC applications with a focus on the safety-related aspects was carried out to identify where the current challenges are from the designer's point of view. An additional review of the state of the art of using systems engineering principles and methods for robotics and safety revealed a large number of past efforts focusing on software engineering, hazard and risk analysis, and ontological development (which underlies all that work). Nevertheless, there were no efforts up to now that focused on the issues that the designer faces with regard to the choice and parameterization of components for HRC applications featuring speed and separation monitoring.

Requirements on a new approach to the consideration of safety based upon the needs of the designer and safety expert were formulated and guided the processes of architecture specification and component modeling that are the focus of this thesis. In particular, reusable models of sensors and robots are necessary in order to have digital information about the application in one place. This allows for easier use and execution of "what-if" analyses that are a part of the design process. An architecture that utilizes existing CAD/simulation tools that designers currently use was specified. The approach was implemented as a computer-aided safety tool (CAS Tool)[2] and the design results were compared with traditional methods with two exemplary use-cases. The use-cases are adapted from real use-cases from the automotive industry.

With the CAS Tool it was possible to quickly make changes to the design, particularly the components such as the robot and safety sensors. The placement of the components and the process itself (i.e. the robot speed) can be easily adapted to achieve specific design goals such as minimum cycle time or minimal footprint of the application. The final outcome was validated against the original set of design requirements, which have been fulfilled. In the exemplary applications described in this thesis, the area of the required safety zone around a specific robot could be reduced by up to 66% by using the CAS Tool based upon this methodology vs current worst-case calculation methods through a more granular data basis for the calculations.

The overall methodology of modeling safety-related attributes of the complete system can have wide-reaching consequences for future robotics applications that are flexible and feature online changes to the program during run-time. Currently, a validation of the complete system is required after it has been commissioned in order to ensure that the desired safety effects (e.g. maintaining a safe separation distance) are indeed achieved by the system. Should the simulation of the robot performance with respect to braking time and distance be correct, the simulation would be sufficient to validate the system. Therefore this approach could offer support in the validation of the overall system performance. Furthermore, given a new robot program, it would also be possible to calculate the safety required minimal protective distance online. Assuming that the system parameters are known, this could offer a simple means for validating that the safety is respected for robotic movements that are not completely pre-planned, but rather generated in real-time. In this sense, it also does not matter

[2] The software implementation of the CAS Tool was not within the scope of this thesis.

who created the new motion commands, either an AI-based agent or a human programmer. The system simply checks what minimum separation distance is required based on the planned motions and sensor parameters, and then checks to ensure that these safety zones are not violated.

Future work will focus on improved implementation of the software tools and support for use with other CAD/simulation software.

8 References

1. 2015: IEEE Standard Ontologies for Robotics and Automation.
2. International Standard Organisation: ISO 10218-1:2011: Robots and robotic devices -- Safety requirements for industrial robots -- Part 1: Robots
3. International Standard Organisation: ISO 10218-2:2011: Robots and robotic devices -- Safety requirements for industrial robots -- Part 2: Robot systems and integration
4. International Standard Organisation: ISO/TS 15066:2016: Robots and robotic devices -- Collaborative robots
5. International Standard Organisation: ISO12100: 2010: Safety of machinery - General principles for design - Risk assessment and risk reduction
6. International Standard Organisation: ISO 13855:2010: Safety of machinery -- Positioning of safeguards with respect to the approach speeds of parts of the human body
7. International Standard Organisation: ISO 13849:2015: Safety of machinery – Safety related parts of control Systems – Part 1: General principles for design
8. International Electrotechnical Commission: IEC 61508-1:2010 Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 1: General requirements
9. Bitkom e.V. Bunderverband Bitkom e.V. Bunderverband Informationswirtschaft, Telekommunikation und neue Medien e.V. (2015): Plattform Industrie 4.0.
10. Byner, C., Matthias, B., & Ding, H. (2019). Dynamic speed and separation monitoring for collaborative robot applications – Concepts and performance. Robotics and Computer-Integrated Manufacturing, 58, 239–252. https://doi.org/10.1016/j.rcim.2018.11.002
11. Directive 2006/42/EC of the European Parliament and of the Council of 17 May 2006 on machinery, and amending Directive 95/16/EC.
12. Abdalla, G.; Damasceno, C. D. N.; Guessi, M.; Oquendo, F.; Nakagawa, E. Y. (2015): A Systematic Literature Review on Knowledge Representation Approaches for Systems-of-Systems. In: Components, Architectures and Reuse Software (SBCARS), 2015 IX Brazilian Symposium on, S. 70–79.
13. Askarpour, M,; Mandrioli, D.; Rossi, M.; and Vicentini, F. “SAFER-HRC: Safety Analysis Through Formal vERification in Human-Robot Collaboration.” SAFECOMP (2016).
14. Awad, R; Fechter, M.; van Heerden, J.: Integrated Risk Assessment and Safety Consideration during Design of HRC Workplaces. In IEEE Emerging Technologies and Factory Automation (ETFA) 2017. September 12-15, Limassol, Cyprus.
15. Bauer, W; Bender, M: Braun, M; Rally, P; Scholtz, O. Lightweight robots in manual assembly—best to start simply!: Examining companies’ initial experiences with lightweight robots. Fraunhofer-Institut Für Arbeitswirtschaft und Organisation IAO, Stuttgart 2016. Retrieved from: https://www.produktionsmanagement.iao.fraunhofer.de/content/dam/produktionsmanagement/de/documents/LBR/Studie-Leichtbauroboter-Fraunhofer-IAO-2016-EN.pdf.
16. Behrens, R.; Saenz, J.; Vogel, C.; Elkmann. N. "Upcoming technologies and fundamentals for safeguarding all forms of human-robot collaboration", 8th International Conference Safety of Industrial Automated Systems (SIAS 2015), Königswinter, Germany 18-20 November, 2015. ISBN 987-3-86423-163-6, S.18-23.

17. Bikas, C.; Argyrou, A.; Pintzos, G.; Giannoulis, C.; Sipsas, K.; Papakostas, N.; Chryssolouris, G. (2016): An Automated Assembly Process Planning System. In: Procedia CIRP 44, S. 222–227. DOI: 10.1016/j.procir.2016.02.085.
18. Bischoff, R.; Guhl, T.; Prassler, E.; Novak, W.; Kraetschmar, G.; Bruyninckx, H.; Siciliano, B.; Pegman, G.; Hägele, M.; Zimmermann, T.; Agirre, J.; Leroux, C.; Tranchero, B.; Labruto, R.; Knoll, A.; Lafrenz, R. (2010). BRICS - Best practice in robotics. International Symposium on Robotics. Munchen, Germany, 7-9 June.
19. DIN 69901-5 (2009): Projektmanagement – Projektmanagementsysteme – Teil 5: Begriffe; Hrsg.: Deutsches Institut für Normung e.V.; Auflage: 9; Verlag: Beuth Verlag GmbH; Adresse: Berlin, Wien, Zürich; http://www.beuth.de/en/standard/din-69901-5/113428752; 2009.
20. Eastman, R; Schlenoff, C.; Balakirsky, S.; Hong, T. (2013): A Sensor Ontology Literature Review: National Institute of Standards and Technology.
21. Fang, J.; Walsh, M. (2018-04-29). "What is Made in China 2025 and why is the world concerned about it?". ABC News. Retrieved 2019-01-10.
22. Gribov, V.; Voos, H. (2013): Safety oriented software engineering process for autonomous robots. In: 2013 IEEE 18th Conference on Emerging Technologies Factory Automation (ETFA), S. 1–8.
23. Guiochet, J.; Motet, G.; Baron, C.; Boy, G. (2004): Toward a human-centered UML for risk analysis. In: Chris W. Johnson und Philippe Palanque (Hg.): Human Error, Safety and Systems Development, Bd. 152. Boston: Kluwer Academic Publishers (IFIP International Federation for Information Processing), S. 177–191.
24. Guiochet, J. (2016): Hazard analysis of human–robot interactions with HAZOP–UML. In: Safety Science 84, S. 225–237. DOI: 10.1016/j.ssci.2015.12.017.
25. http://www.computer-automation.de/feldebene/robotik/artikel/119127/ Cited on 04.09.2017.
26. https://www.extremnews.com/berichte/wirtschaft/9fb01669a439db8 Cited on 04.09.2017.
27. https://opelpost.com/03/2016/kollege-roboter/ Cited on 04.09.2017.
28. https://www.press.bmwgroup.com/deutschland/article/detail/T0209722DE/neuarti ge-mensch-roboter-zusammenarbeit-in-der-bmw-group-produktion?language=de Cited on 04.09.2017.
29. International Federation of Robotics (2016): Executive Summary World Robotics 2016 Industrial Robots. Online under https://ifr.org/img/uploads/Executive_Summary_WR_Industrial_Robots_20161.pd f, cited on 28.08.2017.
30. International Federation of Robotics. (December 2018). Demystifying Collaborative Industrial Robots [Positioning paper]. Retrieved from: https://ifr.org/downloads/papers/IFR_Demystifying_Collaborative_Robots.pdf
31. https://www.eu-robotics.net/sparc/about/index.html Cited on 25.10.2019.
32. Maier, Mark W. (1998): Architecting principles for systems-of-systems. In: Syst. Engin. 1 (4), S. 267–284. DOI: 10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D.
33. Marvel, J., & Norcross, N. (2017). Implementing speed and separation monitoring in collaborative robot workcells. Robotics and Computer-Integrated Manufacturing, S. 144-155.
34. Petersen, H.; Behrens, R.; Saenz, J.; Schulenburg, E.; Vogel, C.; Elkmann, N.; "Reliable Planning of Human-Robot-Collaboration featuring Speed and Separation Monitoring", 9th International Conference on Safety of Industrial Automated Systems – SIAS 2018. October 10-12 2018 - Nancy, France.

35. Operating Instructions SICK microScan3 Core I/O Safety laser scanner. SICK AG. https://cdn.sick.com/media/docs/7/57/757/Operating_instructions_microScan3_Core_I_O_en_IM0063757.PDF Cited on 04.09.2019.
36. Platbrood, F.; Görnemann, O., "Safe Robotics – Safety In Collaborative Robot Systems", White Paper June 2018. Retrieved from: https://cdn.sick.com/media/docs/6/96/996/Whitepaper_Safe_Robotics_en_IM0072996.PDF
37. Robots KR CYBERTECH-2 Specification. KUKA AG. Revision 4, Issued 06.05.2019. *https://www.kuka.com › media › imported › spez_kr_cybertech2_en* Cited on 04.09.2019.
38. Robots, LBR iiwa, LBR iiwa 7 R800, LBR iiwa 14 R820, Specification. Version 8. KUKA AG. Issued 03.05.2019. https://www.kuka.com › kuka-downloads › imported › spez_lbr_iiwa_en Cited on 04.09.2019.
39. RobotEnomics (2016): The facts about Co-Bot Robot sales. Online under https://robotenomics.com/2016/01/11/the-facts-about-co-bot-robot-sales/, cited on 28.08.2017.
40. http://robmosys.eu/approach/ Cited on 04.09.2017.
41. Saenz, J.; Elkmann, N.; Gibaru, O.; Neto, P. (11-13 December, 2017). Survey of methods for design of collaborative robotics applications - Why safety is a barrier to more widespread robotics uptake. 2nd International Conference on Mechanical Engineering and Robotics Research, ICMERR 2017. Paris, France.
42. Saenz, J., Vogel, C., Penzlin, F., & Elkmann, N. (27-30 June 2017). Safeguarding collaborative mobile manipulators - Evaluation of the VALERI workspace monitoring system. Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017. Modena, Italy.
43. Saenz, J., Aske, L., Bidard, C., Buurke, J.H., Nielsen, K., Schaake, L, Vicentini, F.: "COVR – Towards simplified evaluation and validation of collaborative robotics applications across a wide range of domains using robot safety skills", 9th International Conference on Safety of Industrial Automated Systems – SIAS 2018. October 10-12 2018 - Nancy, France.
44. Salmi, T.; Väätäinen, O.; Malm, T.; Montonen, J.; Marstio, I. "Meeting new challenges and possibilities with modern robot safety technologies," in Enabling Manufacturing Competitiveness and Economic Sustainability, M. F. Zaeh, Ed. Springer International Publishing, 2014, pp. 183–188.
45. Schröter, D.; Kuhlang, P.; Finsterbusch, T.; Kuhrke, B.; Verl, A. (2016): Introducing Process Building Blocks for Designing Human Robot Interaction Work Systems and Calculating Accurate Cycle Times. In: *Procedia CIRP* 44, S. 216–221. DOI: 10.1016/j.procir.2016.02.038.
46. Stampfer, D.; Lotz, A.; Lutz, M.; Schlegel, C. (2016): The SmartMDSD Toolchain: An Integrated MDSD Workflow and Integrated Development Environment (IDE) for Robotics Software 7(1), S. 3–19.
47. Tenorth, M.; Bartels, G.; Beetz, M. (2014): Knowledge-based Specification of Robot Motions. In: Proc. of the European Conference on Artificial Intelligence (ECAI).
48. Tsarouchi, P.; Spiliotopoulos, J.; Michalos, G.; Koukas, S.; Athanasatos, A.; Makris, S.; Chryssolouris, G. (2016): A Decision Making Framework for Human Robot Collaborative Workplace Generation. In: *Procedia CIRP* 44, S. 228–232. DOI: 10.1016/j.procir.2016.02.103.
49. W. K. Vaneman (2016): The system of systems engineering and integration "Vee" model. In: 2016 Annual IEEE Systems Conference (SysCon), S. 1–7.

50. Vicentini, F.; Giussani, M.; Tosatti, L. M. "Trajectory-dependent safe distances in human-robot interaction," Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), Barcelona, 2014, pp. 1-4
51. Vogel, C.; Walter, C.; Elkmann, N.: "Safeguarding and supporting future human-robot cooperative manufacturing processes by a projection- and camera-based technology", Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy.
52. Walden, David D.; Roedler, Garry J.; Forsberg, Kevin; Hamelin, R. Douglas; Shortell, Thomas M.: Systems engineering handbook. A guide for system life cycle processes and activities. 4th edition.
53. Wortmann, U. (2017). Sichere MRK bei Procter & Gamble – Entwicklung und Realisierung des CoPal. Presentation, Magdeburg, Germany.
54. N. Yakymets, S. Dhouib, H. Jaber, A. Lanusse (2013): Model-driven safety assessment of robotic systems. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, S. 1137–1142.
55. Axelsson, J. (2015): A systematic mapping of the research literature on system-of-systems engineering. In: System of Systems Engineering Conference (SoSE), 2015 10th, S. 18–23.
56. Muller, G.; Andersen, J. H. (2015): Factory production line as SoS; a case study in airplane engine component manufacturing. In: System of Systems Engineering Conference (SoSE), 2015 10th, S. 42–46.
57. Clark, J. O.: System of Systems Engineering and Family of Systems Engineering from a standards, V-Model, and Dual-V Model perspective. In: 2009 3rd Annual IEEE Systems Conference. Vancouver, BC, Canada, S. 381–387.
58. Chhaniyara, S.; Saaj, C. M.; Maediger, B.; Althoff-Kotzias, M.; Ahrns, I. (2011): Model based system engineering for space robotic systems. In: Proceedings of 11th Symposium on Advanced Space Technologies in Robotics and Automation.
59. C. Schlegel; T. Hassler; A. Lotz; A. Steck (2009): Robotic software systems: From code-driven to model-driven designs. In: Advanced Robotics, 2009. ICAR 2009. International Conference on, S. 1–8.
60. Mhenni, F.; Nga Nguyen; Kadima, H.; Choley, J.: Safety analysis integration in a SysML-based complex system design process. In: 2013 7th Annual IEEE Systems Conference (SysCon). Orlando, FL, S. 70–75.
61. Martin-Guillerez, D.; Guiochet, J.; Powell, D.; Zanon, C.: A UML-based method for risk analysis of human-robot interactions. In: G. Marzo Di Serugendo und J. S. Fitzgerald (Hg.): the 2nd International Workshop. London, United Kingdom, S. 32–41.
62. Scippacercola, F.; Pietrantuono, R.; Russo, S.; Silva, N. P.: SysML-based and Prolog-supported FMEA. In: 2015 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW). Gaithersburg, MD, USA, S. 174–181.
63. Friedenthal, S.; Moore, A.; Steiner, R. (2008): A practical guide to SysML. Systems Model Language. Burlington, Mass.: Elsevier/Morgan Kaufmann.
64. Weilkiens, T.: Systems Engineering mit SysML/UMS, Modellierung, Analyse, Design. ISBN 978-3898644099
65. Delligatti, L.(2013): SysML distilled. A brief guide to the systems modeling language: Pearson Education, Inc.
66. Biggs, G.; Sakamoto, T.; Fujiwara, K.; Anada, K. (2013): Experiences with model-centred design methods and tools in safe robotics. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, S. 3915–3922.

67. Yakymets, N.; Dhouib, S.; Jaber, H.; Lanusse, A. (2013): Model-driven safety assessment of robotic systems. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, S. 1137–1142.
68. Tenorth, M.; Clifford Perzylo, A.; Lafrenz, R.; Beetz, M.: The RoboEarth language: Representing and exchanging knowledge about actions, objects, and environments. In: 2012 IEEE International Conference on Robotics and Automation (ICRA). St Paul, MN, USA, S. 1284–1289.
69. Corcho, O.; Fernández-López, M.; Gómez-Pérez, A.(2003): Methodologies, tools and languages for building ontologies. Where is their meeting point? In: Data & Knowledge Engineering 46 (1), S. 41–64. DOI: 10.1016/S0169-023X(02)00195-7.
70. Noy, N.F.; McGuinness, D.L. (2001): Ontology Development 101: A Guide to Creating Your First Ontology. Stanford University.
71. Saenz, J. "Workspace Sharing in Mobile Manipulation." In Human–Robot Interaction, 81–89. Chapman and Hall/CRC, 2019. https://doi.org/10.1201/9781315213781-6.
72. Fiorini, S.R.; Chibani, A.; Haidegger, T.; Carbonera, J.L.; Schlenoff, C; Malec, J.; Prestes, E.; Gonçalves, P.; Ragavan, S.V.; Li, H.; Nakawala, H.; Balakirsky, Bouznad, S.; Ayari, N.; Amirat, Y. "Standard Ontologies and HRI" In Human–Robot Interaction, 81–89. Chapman and Hall/CRC, 2019. https://doi.org/10.1201/9781315213781-3.
73. Huelke, M.; Lungfiel, A.: Software-Assistent SISTEMA: Berechnung und Bewertung der Maschinensicherheit. SPS-Magazin 24 (2011) Nr. 6, p. 49-50.
74. Safety Laser Scanner UAM 05LP User's Manual, Document No: C-61-00003-2. Hokuyo Automatic Co., Ltd. https://www.reer.it/reer_dati/FILESSIC/D5/0100790_ING.pdf . Cited on 04.09.2019.
75. Safexpert software for CE Marking https://www.ibf.at/en/ce-software-safexpert/ Cited on 04.09.2019
76. Ibanez, A.; Bocquet, B.; Martinez, C.; Urretavizcaya, I.; Juez, G.; Amparan, E.; Martinez, J. D2.2 eITUS Demonstrator. May 15, 2019. Retrieved from: https://robmosys.eu/e-itus/.
77. Nascimento, A. M. ; Vismari, L. F.; Cugnasca, P. S. ; Camargo Jr., J. B.; Almeida Jr., J. R.; Inam, R.; Fersman, E.; Hata A.; Marquezini, M. V. "Concerns on the Differences Between AI and System Safety Mindsets Impacting Autonomous Vehicles Safety," in Computer Safety, Reliability, and Security. SAFECOMP 2018. Lecture Notes in Computer Science.
78. Mössner, T., & Bundesanstalt Für Arbeitsschutz Und Arbeitsmedizin. (2012). Risikobeurteilung im Maschinenbau. Dortmund Bundesanst. Für Arbeitsschutz Und Arbeitsmedizin. Bochum: Verlag Technik & Information 2009
79. Forsberg, K. and Mooz, H., "The Relationship of Systems Engineering to the Project Cycle" Archived 2009-02-27 at the Wayback Machine, First Annual Symposium of the National Council On Systems Engineering (NCOSE), October 1991.
80. Neches, R.; Fikes, R.E. ; Finin, T.; Gruber, T.R.; Senator, T.; Swartout, W.R. Enabling technology for knowledge sharing, AI Magazine 12 (3) (1991) 36–56.
81. Ssn.owl
82. Albrecht, P. "Standardization Roadmap – The future of work," 6th European Conference on standardization, testing and certification in the field of occupational safety and health (EUROSHNET) June 12-14 2019. Retrieved from https://www.euroshnet.eu/fileadmin/Redaktion/Presentations/2_Albrecht_-_Standardization_roadmap.pdf Cited on 04.10.2019.

83. Behrens, Roland: "Biomechanische Grenzwerte für die sichere Mensch-Roboter-Kollaboration", Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2019.

9 Appendix – Example checklists for use during design process

9.1 Checklist for general and essential requirements

This checklist corresponds to the step “General and essential requirements” from Figure 3.

Ergonomics

The design of the collaborative work-cell considers the variability of the operator's physical dimensions, strength and stamina	☐
The layout provides sufficient space for movements of the operators or its body parts	☐
The process design avoids a machine-determined cycle time and working speed	☐
The process design avoids monitoring action that requires lengthy concentration	☐
The interface is adapted to the foreseeable characteristics of the operator	☐
The position of the operator allows for good working conditions and is protected against any foreseeable hazards	☐
The position of the operator must allow rapid evacuation	☐

Lighting

Integral lighting is used for the intended operations where the absence thereof is likely to cause a risk	☐
The design avoids areas of shadow likely to cause nuisance, irritating dazzle or stroboscopic effects on moving parts due to lighting	☐
Internal parts requiring frequent access must be provided with appropriate lighting	☐

General mechanics

The accessible parts within the collaborative space have no sharp edges, no sharp angles and no rough surfaces likely to cause injury	☐
Precautions are taken to prevent risks from falling or effected objects	☐
Within the collaborative work-cell there are no hazard which may cause slips or falls	☐

Robot system - Control systems

The control system can withstand the intended operating stresses and external influences	☐
A fault in the hardware or software does not lead to hazardous situations	☐
Any actuated component in the collaborative work-cell cannot start unexpectedly	☐
The parameter of the robot system cannot change in an uncontrolled way, where such change may lead to hazardous situations	☐
The robot system cannot be prevented from stopping if the stop signal has already been given	☐
The moving robot arm or a piece held by the robot cannot fall or rejected	☐

Robot system - Control Devices

Any control device is clearly visible and identifiable, using pictogram where appropriate	☐
Any control device is positioned in such a way as to be safely operated without hesitation or loss of time and without ambiguity	☐
The movement of the control device is consistent with its effect	☐
Any control device is positioned in such a way that their use cannot cause additional risk	☐
Any control device is made in such a way as to withstand foreseeable forces	☐

Robot system - Emergency Stop

At least one emergency stop is provided to enable actual or impending danger to be averted	☐
Each emergency stop is clearly identifiable, clearly visible and quickly accessible	☐
Initiating an emergency stop halts the process as quickly as possible, without creating additional risks	☐

The stop command sustains until the engagement of the emergency stop is overridden	☐
Disengaging the emergency stop will not restart the machinery but only permit restarting	☐
The emergency stop function must be available and operational at all times	☐
Emergency stop devices are only a back-up to other safeguarding measures and not a substitute for them	☐

9.2 Checklist for system limits and requirements

The following checklist correspond to the step “System limits and requirements” from Figure 3.

Robot system - Control system

The robot control is safety-rated and certified in PL d category 3*		☐
The safety-rated characteristics of the robot system and all further machines are specified in the operator instructions		☐

*) A lower category can be applied if the result of the risk assessment proves that such a measure is appropriate according to the risk of identified hazards.

Robot system - Stop and restart

The robot system stops if a human enters the safe-guarded space		☐
A restart of the stopped robot system is avoided if a human is still in the safe-guarded space		☐
A restart of the stopped robot system can only be initiated by a human manually		☐

Robot system - Protective and emergency stop

Each input device for setting (initiating or changing) robot motion is provided with an emergency stop		☐
The operator can stop the robot system or exit the co-space at any time by a single action		☐
Any failure in the safety related parts of the robot control initiates a protective stop of category 0 or 1		☐

Collaborative space

The collaborative space is conspicuously highlighted and as such visible		☐
Any person in the collaborative space is individually protected		☐
Safety-rated soft axes and space-limiting functions are applied for limiting robot motions on an appropriate minimum		☐
The minimum clearance distance between the robot system and points**, where whole-body clamping can occur, is at least 500mm		☐

Collaborative operation

The robot system has a label that indicates a shared workspace with humans and appears prominently for any person		☐
The robot system notifies the operator if collaborative mode is active or not		☐
The robot system notifies the operator about each transitions between collaborative and non-collaborative mode or vice versa		☐

METHODOLOGIE POUR LA CONCEPTION D'ESPACES DE TRAVAIL SURS, AVEC LA COLLABORATION DE ROBOT HUMAIN

RESUME :

CETTE THESE PROPOSE LE DEVELOPPEMENT SCIENTIFIQUE D'UNE METHODE PERMETTANT DE PRENDRE EN COMPTE LES ASPECTS DE SECURITE POUR LES APPLICATIONS INDUSTRIELLES FAISANT INTERVENIR LA COLLABORATION HOMME-ROBOT (APPLICATIONS HRC). CETTE METHODOLOGIE PROPOSE DE PRENDRE EN COMPTE CES ENJEUX DE SECURITE DES LA PHASE DE CONCEPTION, SUR LA BASE DES CONCEPTS D'INGENIERIE DES SYSTEMES, D'ONTOLOGIES ET D'INDUSTRIE 4.0. LES EXIGENCES RELATIVES A UNE NOUVELLE APPROCHE DE LA PRISE EN COMPTE DE LA SECURITE BASEE SUR LES BESOINS DU CONCEPTEUR ET DE L'EXPERT EN SECURITE ONT ETE FORMULEES ET ONT GUIDE LES PROCESSUS DE SPECIFICATION D'ARCHITECTURE ET DE MODELISATION DES COMPOSANTS QUI SONT AU CENTRE DE CETTE THESE. UNE ARCHITECTURE UTILISANT LES OUTILS DE CAD / SIMULATION EXISTANTS UTILISES PAR LES CONCEPTEURS A ETE SPECIFIEE. L'APPROCHE A ETE MISE EN ŒUVRE SOUS LA FORME D'UN OUTIL DE SECURITE ASSISTE PAR ORDINATEUR NOMME CAS TOOL. LES RESULTATS DE LA CONCEPTION ONT ETE COMPARES AUX METHODES TRADITIONNELLES SUR DEUX EXEMPLES D'UTILISATION. AVEC NOTRE OUTIL CAS, IL EST POSSIBLE DE MODIFIER RAPIDEMENT LA CONCEPTION, EN PARTICULIER LES COMPOSANTS TELS QUE LE ROBOT ET LES CAPTEURS DE SECURITE. LES ZONES DE SECURITE CALCULEES A L'AIDE DE L'OUTIL CAS SONT REDUITES JUSQU'A 66% PAR RAPPORT AUX METHODES TRADITIONNELLES LES PLUS DEFAVORABLES.

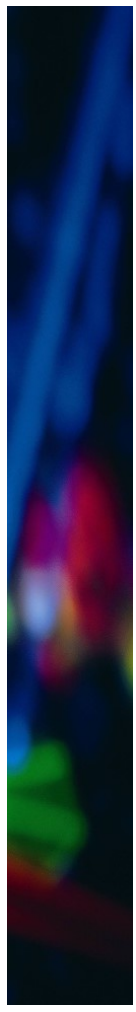
Mots clés : Interaction homme-robot, robotique collaborative, sécurité, ingénierie des systèmes, industrie 4.0

METHODOLOGY FOR DESIGN OF SAFE WORKSPACES FEATURING HUMAN ROBOT COLLABORATION

ABSTRACT :

THIS THESIS FOCUSES ON THE DEVELOPMENT OF A METHOD FOR CONSIDERING THE SAFETY ASPECTS OF INDUSTRIAL APPLICATIONS FEATURING HUMAN-ROBOT COLLABORATION (HRC APPLICATIONS) DURING THE DESIGN PHASE BASED ON SYSTEMS ENGINEERING, ONTOLOGIES, AN INDUSTRY 4.0 CONCEPTS. REQUIREMENTS ON A NEW APPROACH TO THE CONSIDERATION OF SAFETY BASED UPON THE NEEDS OF THE DESIGNER AND SAFETY EXPERT WERE FORMULATED AND GUIDED THE PROCESSES OF ARCHITECTURE SPECIFICATION AND COMPONENT MODELLING THAT ARE THE FOCUS OF THIS THESIS. AN ARCHITECTURE THAT UTILIZES EXISTING CAD/SIMULATION TOOLS THAT DESIGNERS CURRENTLY USE WAS SPECIFIED. THE APPROACH WAS IMPLEMENTED AS A COMPUTER-AIDED SAFETY TOOL (CAS TOOL) AND THE DESIGN RESULTS WERE COMPARED WITH TRADITIONAL METHODS WITH TWO EXEMPLARY USE-CASES. WITH THE CAS TOOL, IT WAS POSSIBLE TO QUICKLY MAKE CHANGES TO THE DESIGN, PARTICULARLY THE COMPONENTS SUCH AS THE ROBOT AND SAFETY SENSORS. THE SAFETY ZONES CALCULATED WITH THE CAS TOOL ARE UP TO 66% SMALLER THAN WITH TRADITIONAL, WORST-CASE METHODS.

Keywords : human-robot interaction, collaborative robots, safety, systems engineering, industry 4.0



Arts et Métiers Sciences et Technologies

HESAM UNIVERSITÉ