Safety of Autonomous Cognitive-oriented Robots

Von der Fakultät für Ingenieurwissenschaften, Abteilung
Maschinenbau der Universität Duisburg-Essen
zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften
Dr.-Ing.

genehmigte Dissertation

von

Philipp Ertle

aus

Reutlingen

Gutachter: Univ.-Prof. Dr.-Ing. Dirk Söffker
Prof. Dr.-Ing. Holger Voos

Tag der mündlichen Prüfung: 05. Dezember 2013

Danksagung

Diese Arbeit ist im Rahmen des Forschungsprojektes ‘ZAFH Autonome Mobile Serviceroboter’ an der Hochschule Ravensburg-Weingarten entstanden. Leiter meines Teilprojekts war Herr Prof. Dr.-Ing. Holger Voos, der später einem Ruf an die Universität Luxembourg folgte. Ihm gilt zunächst mein Dank, da er mir diese Möglichkeit überhaupt erst eröffnete. Stets war er offen für Vorschläge, hörte sich geduldig Probleme an, stand mit guten Ratschlägen hilfreich zur Seite und zeigte sich meinen Veränderungswünschen gegenüber immer verständnisvoll und unterstützend.

Des Weiteren gilt mein Dank Herrn Univ.-Prof. Dr.-Ing. Dirk Söffker von der Universität Duisburg-Essen. Zuerst einmal zeigte er sich hinsichtlich der Promotionswünsche eines Fachhochschulabsolventen sehr offen, was bedauerlicherweise nicht häufig an deutschen Universitäten anzutreffen ist. Außerdem band er mich, auch teilweise über große Entfernung, in das Lehrstuhlgeschehen ein. Wertvolle Korrekturen und Hinweise, ein verständnisvoller, offener und freundlicher Umgang waren immer ein Markenzeichen der Kommunikation. Einen besseren Doktorvater hätte ich mir nicht wünschen können.

Während meiner Zeit im ZAFH hatte ich das Glück sehr liebe Kollegen zu haben, die immer für eine Diskussion, einen Scherz, oder Problemersuchen offen waren. Ebenso in der Hochschule in Weingarten, wie auch bei Besuchen des Lehrstuhls in Duisburg. Insbesondere möchte ich mich dafür bei meinem langjährigen Room-Mate Michel Tokic bedanken, der immer guten Rat, aufbauende Worte und Zeit für Diskussionen übrig hatte. Gleiches gilt für Denis Gamrad, meinem Mitstreiter aus Duisburg, von dem ich viel lernte, insbesondere bei gemeinsamen Arbeiten. Beiden gilt der Dank für die nette Zusammenarbeit bei gemeinsamen Arbeiten und den gemeinsamen Konferenzreisen.

Ohne meine netten Kollegen wäre diese Zeit ungleich schwerer und farbloser gewesen. Daher auch Dank an Achim Feucht, Marco Junglas, Amir Kazeminia, Yvonne Vengels, Richard Cubek, Haitham Bou Ammar, Tobias Fromm, Markus Schneider, Arne Usadel, Benjamin Stähle, Jürgen Behmüller, Karl Glatz, Joachim Fessler, Gregor Flesch, Matthias Marx, Xi Nowak, Xingguang Fu, Quang Khanh Luu und allen anderen im ZAFH und im Lehrstuhl SRS. Im besonderen Maße unterstützt hat mich auch Prof. Dr. Wolfang Ertel, indem er mich in seine Gruppe und auch über große Entfernung in den Hochschulkontext eingebunden hat.

Außerordentlicher Dank gilt meinen Eltern Drs. Annemarie und Andreas Ertle, die überhaupt die Voraussetzungen schufen, die mir ein Studium ermöglichten. Aber auch dafür, dass sie mir in allen Belangen stets mit Rat und Tat beiseite standen.

Ganz spezieller Dank gilt meiner lieben Frau Verena, die ich immer an meiner Seite wähnen konnte, die alle Launen und schwere Zeiten mit mir durchstand und diese mit mir ertrug und auch oft mich ertragen musste. Sie gibt meinem Leben die richtige Richtung, sie ist meine Muse und mein Ansporn.

Biberach, den 09. Dezember 2013
Philipp Ertle

Abstract

Service robots shall very soon autonomously provide services in all spheres of life. They have to execute demanding and complex tasks in a dynamic environment, collaborate with human users in a natural and intuitive way and adapt themselves to varying conditions. It can be assumed that complex robots have to be able to learn, if they shall be able to provide complex tasks in unstructured environments in an autonomous fashion. The capability to 'act autonomously' is often mentioned in conjunction with robots, however, the perception and understanding of the term autonomy varies among the different research fields. Therefore, a closer look is taken at robot autonomy and intelligence, in particular, with regard to current and future robots. From this perspective, implications for safety are derived concerning safe autonomous behavior.

In order to push forward the robot safety in the light of safe behavior in complex environments, a novel classification of robot hazards is provided. Based on this, the so-called object interaction hazards are derived which arise when objects that are, for instance, located in the near environment, interact with objects that are manipulated by a robot. Taking into account the current state-of-the-art, it can be stated that this denotes a novel problem area. This problem area is so far addressed neither in current research work, nor in the relevant standards.

The new type of hazards can be assigned to a group of hazards that originate from the interaction with a complex and unstructured environment. In order to sufficiently consider the environment and operation context, the robot has to be aware of it. In the field of cognitive (technical and biological) systems, this key capability can be called 'situation awareness' (cf. Söffker, 2008). Based on Endsley (1995)'s definition of situation awareness, Wardziński (2008) proposes the 'dynamic risk assessment' approach, which shall enable the robot perceive the risk of current and upcoming situations. In order to realize such a risk-aware planning system for the first time, dynamic risk assessment is integrated within a cognitive architecture in order to utilize cognitive functions, such as anticipation, planning and learning. Here, the so-called action spaces (sets of possible upcoming situations) are dynamically anticipated within the underlying cognitive architecture, and a risk assessment component assesses them with regard to comprised risks. Thus, the generated risk information can be utilized for a risk-aware action planning. The proper operation of this integrating concept is demonstrated via simulations and with a robot experiment.

In order to consider (object interaction) hazards by means of dynamic risk assessment, (initial) knowledge about hazards is required. Thus, a novel procedural model is developed for systematically generating a safety knowledge base. In this connection, the concept is to formalize risk models (risk description rules) in a generalized manner so that they remain valid for future, and so far unknown situations. The structure of so-called 'Safety Principles' can be additionally considered as meta-structure for integrating already available safety-related approaches with the context-aware system (adaptive collision avoidance, adaptive compliance actuation, etc.).

However, it can be assumed that the safety knowledge potentially lacks completeness. The application of AI-based approaches constitutes a noteworthy opportunity (Fox

and Das, 2000), for instance, learning of safety-related knowledge. For this reason, light is shed on strategically influential learning methods in safety-critical contexts: On the one hand, well-known 'reinforcement learning' algorithms are investigated paying special attention to their performance to learn and relearn in the presence of unknown and hazardous situations. On the other hand, a 'learning from demonstration' approach is investigated, constituting interesting potential for improving or simplifying the generation process of the safety knowledge (either in the development phase or even in the operation phase of the system). A gathered risk model in turn is integrated within the dynamic risk assessment approach.

Finally, this work reveals a general perspective on potential hazards of future autonomous robots. It describes the generation, integration, utilization, and maintenance of a system-internal safety knowledge base for dynamic risk assessment. It denotes an overall concept toward solving the advanced safety problem of intelligent autonomous robots (systems). Consequently, feasibility of safe behavior of autonomous and intelligent systems is confirmed in principle.

Kurzfassung

Autonome mobile Serviceroboter sollen zukünftig selbständig Dienstleistungen in allen Lebensbereichen erbringen, auch in direkter Nähe zum Menschen. Hierbei kann angenommen werden, dass komplexe Roboter lernfähig sein müssen, wenn sie komplexe Aufgaben in unstrukturierten Umgebungen autonom erbringen können sollen. Die Fähigkeit autonom zu Handeln ist essentiell für viele Roboteranwendungen, allerdings wird dieser Begriff in der Robotik sehr unterschiedlich aufgefasst. Daher wird der Begriff Autonomie zunächst eingehender beleuchtet, in Hinsicht auf gegenwärtige und vor allem auf zukünftige Roboter. Hierauf basierend wird abgeleitet, welche Konsequenzen dies hinsichtlich der Sicherheit nach sich ziehen kann.

Um das Verständnis für Sicherheit in der Robotik zu erweitern, wird zunächst eine neue Klassifizierung der möglichen Gefahren vorgenommen. Hiervon wird die Klasse der Objektinteraktionsgefahren abgeleitet. Vor allem wenn Objekte vom Roboter aufgenommen, transportiert und abgestellt werden, können Gefahren dadurch provoziert werden, dass diese mit Objekten der Umgebung auf gefährliche Art und Weise interagieren. In Anbetracht des aktuellen Stands der Sicherheitstechnik in der Robotik wird klar, dass sich hier ein neues Problemfeld auftut, welches bisher weder in Forschungsarbeiten, noch in der entsprechenden Normung Berücksichtigung findet.

Die benannten Gefahren entstehen überwiegend durch die Interaktion des Roboters mit einer unstrukturierten Umgebung. Um seine Umgebung und den gegenwärtigen Kontext adäquat berücksichtigen zu können, muss der Roboter Kenntnis desselben haben. Eine solche Schlüsselfähigkeit kann im Bereich der (biologisch und technisch) kognitiven Systeme als 'Situation Awareness' (Situationsbewusstsein) bezeichnet werden (vgl. Söffker, 2008). Basierend auf der Definition von Situationsbewusstsein von Endsley (1995) schlägt Wardziński (2008) den Ansatz der dynamischen Risikobewertung vor. Hierbei soll der Roboter selbst in die Lage versetzt werden, das Risiko einer Situation ermitteln zu können. Um eine solche risikobewusste Handlungsplanung erstmals zu realisieren, wird der dynamische Risikobewertungsansatz in eine kognitive Architektur integriert, um deren kognitiven Funktionen, wie Antizipation, Planen und Lernen zu nutzen. Hierbei werden mögliche Handlungsräume mittels der zugrundeliegenden kognitiven Architektur dynamisch antizipiert und mittels dynamischer Risikoanalyse auf mögliche Gefahren untersucht. Diese zusätzliche Risikoinformation findet dann in der nachfolgenden Handlungsplanung Berücksichtigung, um risikoadäquate Handlungsabläufe zu realisieren. Die Funktionsfähigkeit des Konzepts wird mittels Simulationen und eines Roboterexperimentes gezeigt.

Um (Objektinteraktions-) Gefahren mittels dynamischer Risikountersuchung bestimmen zu können, bedarf es eines (initialen) Wissens über mögliche Gefahren. Aus diesem Grund wird ein Vorgehensmodell zur systematischen Erzeugung einer solchen Sicherheitswissensbasis entwickelt. Das Konzept ist hierbei, dass sogenannte Risikomodelle in generalisierter Weise formalisieren werden, sodass diese ihre Gültigkeit in zukünftigen und unbekannten Situationen behalten. Die Struktur der sogenannten Sicherheitsprinzipien kann ebenfalls als Modell dienen, um vorhandene sicherheitsbezogene An-

sätze (z.B. adaptive Kollisionsvermeidung und adaptive, nachgiebige Roboterantriebe) mit dem dynamischen Umgebungskontext in Verbindung zu bringen.

Das Gefahrenwissen ist jedoch potentiell unvollständig. Daher stellt die Erweiterung und Verfeinerung desselben eine Notwendigkeit dar, da neue Gefahren auftauchen können, bzw. bisheriges Gefahrenwissen detailliert werden muss. Hierbei können die Ansätze aus dem Bereich der künstlichen Intelligenz als nützliche Möglichkeit wahrgenommen werden (Fox and Das, 2000). Aus diesem Grund werden strategisch wichtige Lernmethoden hinsichtlich der Anwendung in einem sicherheitskritischen Kontext untersucht. Einerseits werden einige etablierte Reinforcement Learning Algorithmen untersucht, inwiefern diese Handlungsstrategien in Hinsicht auf unbekannte und gefährliche Situationen erlernen, bzw. auch 'umlernen'. Andererseits wird ein Ansatz, basierend auf dem Lernen durch Demonstration-Paradigmas so integriert, dass der Entwicklungs- oder Erweiterungsprozess des Sicherheitswissens (zur Entwicklungszeit oder zur Betriebszeit des Roboters) erleichtert und verbessert werden kann. Ein Risikomodell, dass in diesem Zusammenhang entsteht, wird anschließend (nach erneuter Untersuchung) wieder für die dynamische Risikoermittlung eingesetzt.

Die vorliegende Arbeit trägt somit schlussendlich zur Untersuchung der potentiellen Gefahren heutiger und zukünftiger Roboter bei. Sie beschreibt die Erzeugung, die Integration, die Verwendung und die Aufrechterhaltung einer systeminternen Sicherheitswissensbasis zum Zwecke der dynamischen Risikountersuchung. Sie stellt hierbei ein Gesamtkonzept dar, dass zur Lösung des erweiterten Sicherheitsproblems von autonomen und intelligenten Robotern (Systemen) beiträgt. Die prinzipielle Realisierbarkeit des sicheren Betriebs von autonomen und intelligenten Systemen ist somit bestätigt.

Contents

1 Introduction and Background 1

1.1 Introduction . . . 1

1.1.1 Motivation . . . 1

1.1.2 Organization and contribution of this work . . . 4

1.2 Background . . . 6

1.2.1 Robot science and robotic systems . . . 6

1.2.2 Cognitive aspects . . . 10

1.2.3 System safety . . . 13

2 State-of-the-Art in Robotic Safety 17

2.1 Safety Standards . . . 17

2.2 Determining the Injury Potential of Collisions . . . 18

2.3 Mechanical Safe Robot Design . . . 18

2.3.1 Robot shape . . . 18

2.3.2 Reduction of colliding masses . . . 18

2.3.3 Passive compliant actuation . . . 19

2.3.4 Summary . . . 20

2.4 Reactive Safety Approaches in Robotics . . . 22

2.4.1 Safety integrated into robot manipulator control . . . 22

2.4.2 Safety integrated into reactive robot platform motion planning . 24

2.4.3 Summary . . . 26

2.5 Safe High-level Reasoning and Planning . . . 26

2.5.1 Risk in Utility Theory . . . 28

2.5.2 The safety bag concept . . . 29

2.5.3 Guardian agents . . . 29

2.5.4 Dynamic risk assessment . . . 31

2.5.5 Safety for the system situation awareness . . . 31

2.5.6 Summary . . . 33

3 Robot Autonomy 35

3.1 Autonomous Systems . . . 35

3.2 Aspects of Autonomy . . . 35

3.3 Present, Desired, and Future Robot Autonomy . . . 36

3.4 Autonomy for Structural Drift . . . 37

3.4.1 Autonomy as control of control . . . 39

3.4.2 Autonomy as adaptability . . . 40

3.4.3 Autonomy for anticipation . . . 41

3.4.4 Autonomy as reciprocal information entropy . . . 41

3.4.5 Informed autonomy . . . 42

3.4.6 Research-related mechanisms generating autonomy 43
3.4.7 Autonomy via nature and nurture 43
3.4.8 Summarizing annotations 44
3.5 Implications for Safety 45
3.5.1 A cognitive perspective 46
3.5.2 The psychodynamic structure model 47

4 A Cognitive-oriented Architecture 51
4.1 Situation-Operator-Model 51
4.1.1 Representing classical control problems 52
4.1.2 Representing algorithms 52
4.2 Cognitive-oriented Robot Architecture 53
4.2.1 Behavior generation hierarchy 53
4.2.2 Learning issues 54
4.2.3 Adaptive and selective perception 55
4.2.4 Mental action space and action planning 55
4.3 Summary 56

5 Identification and Classification of Robotic Hazards 57
5.1 Defining System and Environment 57
5.1.1 The problem of defining the system boundary 57
5.1.2 Transition of responsibility 58
5.2 The Hazard Theory 58
5.3 Classification of Hazard Causes in Robotic Applications 59
5.3.1 Robot-originated and robot-initiated hazards 61
5.3.2 Robot-originated and human-caused hazards 62
5.3.3 Robot-originated and object-caused hazards 63
5.3.4 Human-originated and robot-caused hazards 63
5.3.5 Object-originated and robot-caused hazards 63

6 Conclusions and Problem Identification 65

7 A priori Formalization and Quantification of Hazards 71
7.1 The Concept of Explicit Safety Knowledge 71
7.1.1 Exemplary safety-critical scenarios 72
7.2 Risk-awareness via intrinsic rule-based safety knowledge 75
7.2.1 The abstracted risk modeling approach 76
7.2.2 The generalized risk modeling approach 77
7.2.3 Modeling risks - the evidence-based approach 78
7.2.4 The conservative risk modeling approach 78
7.2.5 Refinement of the safety knowledge 79
7.3 Operating Hazards Analysis 79
7.3.1 A procedure for modeling object-related safety knowledge 81
7.4 Hazard Matrix Analysis 81
7.4.1 Hazard identification matrix 82
7.4.2 Hazard analysis I: Hazard description 83
7.4.3 Hazard analysis II: Preliminary risk estimation 85
7.5 Specifying the Safety Principles 88

7.5.1 Modeling and quantification of likelihoods and consequences . . 91
7.5.2 Required information basis 94
7.5.3 Evaluation of information quality in early development phases . 95
7.5.4 Determining the risk function 98

8 Realization of the Dynamic Risk Assessment Approach 105
8.1 Integrating Hazard Knowledge 105
8.1.1 Risk-sensitive planning and perception 107
8.1.2 The Risk assessment server 109
8.1.3 Safety clearance to critical situations 110
8.2 Simulation Experiments 113
8.2.1 Risk assessment server experiment 113
8.2.2 Risk-sensitive planning experiment 115
8.2.3 Interactive object manipulation simulation 118
8.3 Robotic Experiment 124
8.3.1 Experimental scenario 125
8.3.2 Experimental results 127

9 The residual incompleteness of the safety-related knowledge 131
9.1 Learning in robotics 131
9.1.1 Implications of learning 131
9.1.2 The problem of incomplete safety knowledge 132
9.2 Recent Work 133
9.2.1 An perspective on safety in reinforcement learning 133
9.3 Safety Aspects in a Non-stationary Reinforcement Learning Problem . 135
9.3.1 The non-stationary experiment 136
9.3.2 The hazard potential of the scenario 137
9.3.3 Experimental setup 139
9.3.4 Results 140
9.3.5 Conclusions 143
9.4 Learning Safety from Demonstration for Robot Skills 144
9.4.1 The extended safety procedure 145
9.4.2 Object interaction risk experiment 148
9.4.3 Results of the ironing task 148
9.4.4 Results of the stack-it-safely task 150
9.4.5 Conclusions 152
9.5 Concluding Remarks 154

10 Summary and Future Work 155
10.1 Summary 155
10.2 Future Work 157

Literature 159

Keyword Register 177

List of Figures

1.1 Vacuum cleaning robot 2
1.2 Robot PR2 folding towels 2
1.3 Care-O-bot3 serving a drink 2
1.4 Research robot ‘Kate’ 3
1.5 Deliberative, reactive, and hybrid robotic paradigms 9
1.6 Model of the human decision making process 12
1.7 The general elements of the system safety process 14
1.8 The ALARP model 15

2.1 Different main concepts for safe and compliant actuators 21
2.2 Danger Index 24
2.3 Process model of safety risk assessment and management 26
2.4 Example of a fault tree mapping system states into hazardous events . 27
2.5 A generic safety protocol 30

3.1 Present and future autonomy aspects 37
3.2 Different kinds of autonomy aspects 38
3.3 Freud’s structure model of the human psyche 48

4.1 Graphical representation of the SOM approach 52
4.2 Graphical illustration of the active/passive operator 52
4.3 ILCA - a model for a cognitive architecture 54
4.4 Different types of situations within ILCA 55
4.5 A dynamically generated, simple exemplary action space 56

5.1 Hazard actuation in a robotic example 60
5.2 Hazard actuation in a service robotic context 60
5.3 Illustration of the different categories of hazard actuation 62

6.1 Parameterization of skills in a cognitive framework 68

7.1 Possible hazards with regard to the kitchen stove 72
7.2 Possible hazards with regard to a power plug 74
7.3 Possible hazards - a hairdryer on the basin 74
7.4 Risk quantification procedure for the dynamic risk assessment 78
7.5 The two principle cycles of safety knowledge refinement 80
7.6 Overall procedural model for specifying the safety hazard knowledge . 82

7.7 Procedural model for identifying object interaction hazards and their hazard causal factors 83
7.8 Procedural model for specifying the Safety Principles 90
7.9 Proper and improper representations 95
7.10 Scenario for estimating the risks of approaching a plastic bowl to the cooking plate 102
7.11 Graphs of the expert estimates, the maximums, and the polynomial interpolation 103

8.1 Realization of the dynamic risk assessment approach 106
8.2 SOM-relation applied to denote the principle of the risk assessment process within a SOM-based approach 106
8.3 The situational object interaction risk assessment process denoted in detail in SOM notation 107
8.4 The dynamic risk assessment process based on initial safety knowledge 108
8.5 Sketch of the communication within the dynamic risk assessment architecture 109
8.6 UML extract of the dynamic risk assessment server structure 110
8.7 Safety clearance delivers risk value for risks with a specific distance . . 113
8.8 The Safety Principle simulator: Simulation comprising different hazardously interacting objects 114
8.9 Graphical illustration of the output of the dynamic risk assessment . . 115
8.10 A simulation example - the arcade game 116
8.11 Graphical representation of the mental action space without dynamic element 117
8.12 Graphical representation of the mental action space including dynamic elements 119
8.13 The Safety Principle simulator: A 2D-world, including typical objects and the robot 120
8.14 The Safety Principle simulator: Kitchen stove and salad bowl example 123
8.15 The Safety Principle simulator: Kitchen stove and watering can example123
8.16 The Safety Principle simulator: Watering can and power plug example 124
8.17 The Safety Principle simulator: Coffee cup and human example 124
8.18 The differential drive robotic experiment - setup and components . . . 125
8.19 Snapshots during the experiment, goal approaching, collision avoiding 128
8.20 Snapshots during the experiment, approaching human, and cooking plate128
8.21 Snapshots during the experiment, deliberating the risk between scalding a human, and electro shock potential 129
8.22 Snapshots during the experiment showing robot transporting a gun . 129

9.1 A classification of situations: Normal and abnormal situations 134
9.2 The non-stationary cliff-walking scenario 137

9.3 The transition probabilities in the scenario computed for a number of taken actions 138
9.4 Hazard performance measure for learning algorithms taking into account the experiment scenario characteristics 139
9.5 Hazard performance of the Q-learning algorithm in combination with different exploration-exploitation strategies 141
9.6 Hazard performance of the Sarsa-learning algorithm in combination with different exploration-exploitation strategies 142
9.7 Hazard performance of Q-learning, and Sarsa-learning in combination with different exploration and exploitation strategies averaged with regard to the single phases 143
9.8 System safety process for learning safety features from demonstration 146
9.9 Need for carefully planned demonstrations 147
9.10 Parameterization of the used learning algorithms 149
9.11 Demonstrating safe behavior for the ironing task 149
9.12 Time plot showing the classification results for the ironing task 151
9.13 Learned decision tree for the ironing task 151
9.14 Demonstrating safe behavior for the stack-it-safely task 152
9.15 Time plot showing the classification results of the stack-it-safely task . 153
9.16 The classification performance of learned classifiers 153

List of Tables

5.1 Available combinations of hazard causal factors typical for robots . . . 61

7.1 Hazard identification matrix for systematically examining object-object interactions 84
7.2 Object matrix: Segmentation of large objects lists in categories 84
7.3 A worksheet example for analyzing the hazard matrix 86
7.4 Qualitative preliminary risk estimation measure for the mishap likelihood 87
7.5 Qualitative preliminary risk estimation measure for the mishap consequence 87
7.6 Decision of risk acceptance based on the preliminary risk estimation . 87
7.7 Worksheet for risks analysis, and estimating the related hazard causal factors 92
7.8 Qualitative evaluation of the efforts that have to be potentially spent to realize a specific information channel to express a desired risk factor 96
7.9 Qualitative evaluation of the estimated reliability of a specific information channel representing a desired risk factor 96
7.10 Qualitative evaluation of the estimated amount that an information channel or a set of them cover the considered hazard actuation 97
7.11 Worksheet for determining likelihood-related risk factors and evaluation of respective information and the information quality 99
7.12 Worksheet for determining severity-related risk factors and evaluation of respective information and the information quality 100
7.13 Exemplary functions for realizing the computation part for estimating the likelihood and severity 101
7.14 Exemplary values of possible expert likelihood estimates for a residential fire in case of approaching plastic to heat sources 103

8.1 The object knowledge base of the risk assessment server containing the attributes of the respective objects 114
8.2 Example of Safety Principles as risk computation models for object interaction accident 114
8.3 The Safety Principle simulator: The list of objects comprised in the object knowledge base 121
8.4 List of ordered hazard scenarios 121
8.5 List of the Safety Principles in the safety knowledge base 122
8.6 The object knowledge base comprising objects and related attributes . 126
8.7 The Safety Principles implemented within the safety knowledge base . 127

Nomenclature

$< sc_i, P_i(\phi_i), P_i(X_i) >$	Risk triplet for scenario sc_i, likelihood $P_i(\phi_i)$, severity $P_i(X_i)$
$\boldsymbol{MRISK}_n$	Situation risk matrix with safety clearance of a number n of operators
$\boldsymbol{M}$	Transition matrix (Markov transitions)
$\boldsymbol{RISK}_{SP}$	Risk matrix comprising for each situation, $\mathrm{s_x}$ risk values, $risk_{k,\mathrm{x}}$ with $k = \{1 \ldots n_{scenario}\}$
$\overline{X}$	Mean of variables in X
$\mathrm{c_i}$	Characteristic i [SOM notation]
C	Set of input channels (characteristics), with $\mathrm{C} = \{\mathrm{c} \mid \mathrm{c}_1 \ldots \mathrm{c}_{\mathrm{n_{IC}}}\}$
$\mathrm{eA_i}$	Explicit assumption i for operator or relation [SOM notation]
F	Function i of operator or relation [SOM notation]
$\mathrm{iA_i}$	Implicit assumption i for operator or relation [SOM notation]
$\mathrm{o_i}$	Operator i [SOM notation]
$\mathrm{r_i}$	Relation i [SOM notation]
$\mathrm{s_i}$	Situation i [SOM notation]
S	Set of all situations [SOM notation]
UCL	Upper confidence limit
$\mathrm{VAR}(X)$	Variance of variables in X
$attr_{k,o}$	Object attribute o of object k, with $o \in \{1 \ldots n_{attr}\}$
$f_i^{acc}(\mathrm{C_q}, \mathrm{Z_v})$	Function for scenario i for mapping the set or subset of input channels (characteristics) $\mathrm{C_q} \subseteq \mathrm{C}$, and parameters $Z_v = \{z \mid z_1 \ldots z_v\}$ to the likelihood value of the accident

$f_i^{sev}(\mathrm{C_r}, \mathrm{Z_w})$	Function for scenario i for mapping the set or subset of input channels (characteristics) $\mathrm{C_r} \subseteq \mathrm{C}$, and parameters $Z_w = \{z \mid z_1 \dots z_w\}$ to the severity value of the accident
H	Set of humans in the robot's 'world'
$mrisk_{n,x,y}$	Element of the situation risk matrix, risk for application of a number n of operators after transferring from situation $\mathrm{s_x}$ to $\mathrm{s_y}$
n_{attr}	Finite number of object attributes
n_{env}	Finite number of objects that are contained in the environment
n_{gr}	Finite number of objects that can be grasped by the robot
n_{IC}	Finite number of input channels (characteristics)
$n_{scenario}$	Finite number of considered scenarios
Obj	Set of objects in the robot's 'world'
obj_k	Object k (grasped by the robot), with $k \in \left\{1 \dots n_{gr}\right\}$
obj_l	Object l (in the environment), with $l \in \{1 \dots n_{env}\}$
$risk_i$	Risk value of the scenario i, with $i \in \{1 \dots n_{scenario}\}$
Rt	Set of robots in the robot's 'world'
Sc	Set of all scenarios.
$t_{(1-\alpha;n-1)}$	$100\% \cdot (1-\alpha)$th percentile of the Student's t-distribution

Acronyms

AIKR	Assumption of insufficient knowledge and resources
ALARP	As low as reasonably practicable
CLE	Coverage level estimate
DI	Danger index
DM^2	Distributed macro-mini actuation
DRA	Dynamic risk assessment
DSF	Design safety feature
DT	Decision tree
ELE	Effort level estimate
HCF	Hazard causal factor
HDAU	Hybrid dual actuator unit
HIC	Head injury coefficient
HMA	Hazard matrix analysis
ICS	Inevitable collision state
IS	Information system
MACCEPA	Mechanically adjustable compliance and controllable equilibrium position actuator
MLP	Multilayer perceptron
PCS	Probabilistic collision state
PDAU	Parallel dual action unit
POMDP	Partially observable Markov decision process
PSSH	Physical symbol system hypothesis
RBFN	Radial basis function network
RF	Risk factor
RLE	Reliability level estimate
RW	Real world
SEA	Series elastic actuation
SLAM	Self localization and mapping
SLM	Safe link mechanism
SOAP	Simple object access protocol
SOM	Situation-Operator-Model
TD	Time difference
VDBE	Value difference based exploration
VDM	Vienna development method
VIA	Variable impedance actuation
WAM	Whole arm manipulator

1 Introduction and Background

1.1 Introduction

The amount of sold traditional industrial robots seems to go into saturation, as it is reported in the study *World Robotics* 2010 of the IFR Statistical Department. However, a new type of robot seems to become successful as its sales potential reveals: The service robot. In this connection, the perception and the definition of the term *robot* have changed, and the robotic domain is currently re-classified (Harper and Virk, 2010). An evolution from robotics over advanced robotics, service robotics, and human-friendly robotics toward personal robotics takes place. This faces researches with drastically increased development problems (Dario et al., 2001).

1.1.1 Motivation

A service robot has to be distinguished from an industrial robot by the kind and intention of tasks to perform. A service robot is a robot which performs services useful to humans, society, or organizations, excluding industrial automation (prEN ISO 8373, 2010). A personal care robot is *"[...] a service robot with the purpose of either aiding actions or performing actions that contribute toward improvement of the quality of life of individuals"* (Harper and Virk, 2010). Hence, this kind of robots has to operate in an unstructured open environment, including also non-qualified, inexperienced users or other humans. Henceforth, the term robot or agent relates to a robot or robotic agent, providing services for personal human-centered applications.

A different class of robots

In accordance to the IFR study *World Robotics* 2010 - *Service Robots*[1], over 70.000 service robots were sold in 2009. These are of course simple robots. The *Navibot* vacuum cleaning robot (**Figure 1.1**) denotes a candidate of robots, recently sold successfully. It is autonomously cleaning the floor on pre-programmed times. Therefore, a camera-based self localization and mapping algorithm is applied in order to build a map of its environment and to navigate systematically through it.

Occasionally, it empties the dirt deposit and recharges its battery at the docking station in order to continue its work at the position of interruption. Thus, it provides a real assistance for every day live.[2]

[1] Executive Summary of World Robotics 2010, `http://www.ifr.org/uploads/media/2010_Executive_Summary_rev_01.pdf`

[2] the author's experience

Figure 1.1: Vacuum cleaning robot

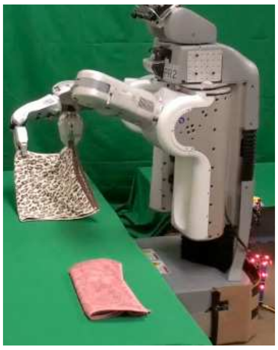


Figure 1.2: Robot PR2 folding towels.[3]

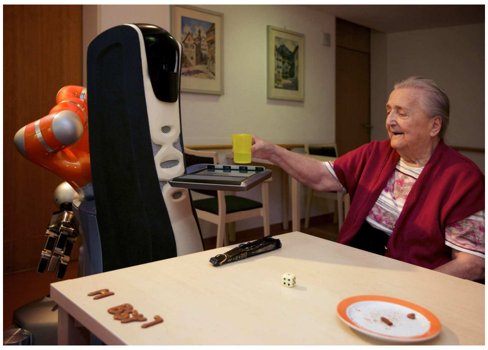


Figure 1.3: Care-O-bot3 serving a drink.[4]

In general, it can be stated that robots are introduced into the human environment. Furthermore, as available robot techniques and approaches are getting more sophisticated, the vision of advanced servant robots seems to be attainable. The *Next Generation Robots* will probably be *"capable of performing such tasks as house cleaning, security, nursing, life-support, and entertainment - all functions to be performed in co-existence with humans in businesses and homes"* (Weng et al., 2009).

In **Figure 1.2,** the *PR2* robot is shown folding towels. Currently, the folding takes a long time - about 3 to 7 minutes (cf. Van Den Berg et al., 2011). Nevertheless, this represents a significant complex task that can be already solved by robots, although, this is based on pre-specified models of the objects that should be folded.

The *Care-O-bot3* serves a drink to an elderly person in **Figure 1.3.** The Care-O-bot3 is based on an omnidirectional platform, enabling it to autonomously plan and follow an optimal, collision free path to a given target by automatically avoiding collisions with dynamic obstacles such as persons. It is equipped with a flexible seven degrees of freedom manipulator as well as with a three-finger hand, making it capable of gripping various different typical household objects.

The research robot *Kate* is based on a differential drive platform and is equipped with a six degrees of freedom manipulator, see **Figure 1.4.** For being able to establish complex tasks, several academic research approaches implemented and encapsulated as skills within the 'SmartSoft' software framework and combined in a meaningful way. Thus, the robot is enabled to navigate through a household environment, recognize several objects and manipulate them for different task purposes. A learning from demonstration approach was integrated into the framework as well, thus, tasks or task sequences, the robot does not know so far, can be taught by a human demonstrator. In consequence, the robot can take a person's order, prepare, and serve respective drinks. Afterwards,

[3] Image: From video `http://www.eecs.berkeley.edu/~pabbeel/personal_robotics.html` [online; accessed 14-October-2013]

[4] Image: Press article Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA, 12-July-2011, `http://idw-online.de/de/news432787` [online; accessed 27-February-2013]

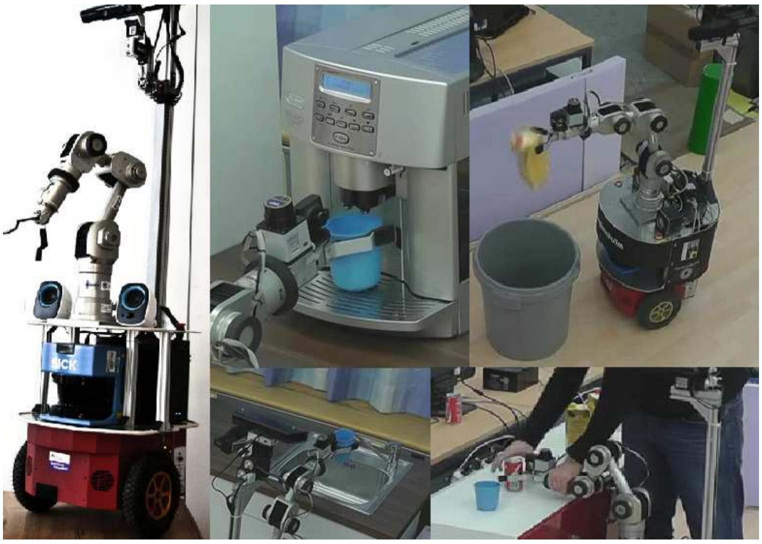


Figure 1.4: Research robot 'Kate'[6]

it cleans up the table and disposes the drinking vessels. The way how a drink is poured into a glass is shown by a user.[5]

From this perspective, it becomes apparent that complex robots in human's everyday live are not mere science fiction. However, this close co-existence of machines with humans provides not only technological challenges, also legal, ethical, and social aspects will become important; Matthias (2008), for instance, addresses arising legal difficulties.

Intelligent machines for unstructured environments

The critical point is that these robots have to operate in an unstructured and complex human environment. This implies that they act as a part of this environment, and in many cases, their tasks also require modifying environment entities. As outlined later on, this implies as well that most of the future (and current research) robots have to be considered to be safety-critical. However, it is quite evident that the robot's designated use excludes the traditional segregation from humans (by Bicchi et al. (2008) referred to as *segregation paradigm*).

As explained later on, it is typical for open environments that the occurrence of new situations can be expected. Thus, researchers in artificial intelligence spend huge efforts enabling systems to adapt to such environments. The systems are intended to show autonomous emergent behavior in such situations at runtime (cf. Russell and Norvig, 2010; Murphy, 2000; Arkin, 1998). In contrast, the principle of safety ideally relies on recognizing and mitigating all undesired hazardous situations before the system is put into operation (Ericson, 2011). From a principal point of view this will lead to contradictions and implies that current safety methods are conceptionally not sufficient for autonomous systems. The development of more and more complex autonomous

[5] Project homepage: `http://www.zafh-servicerobotik.de/`, demonstration video: `http://www.youtube.com/watch?v=nUM3BUCUnpY[online;accessed27-February-2013]`

[6] Images: From video `http://www.youtube.com/watch?v=nUM3BUCUnpY` [online; accessed 27-February-2013]

robots is in principle not fully determined anymore by its design process. This requires carefulness, in particular, with regard to safety requirements. Safety mechanisms of such systems have to keep pace with the achievements of ‘robot-enabling’ - in virtue of the tenor of Jonas: *“Act so that the effects of your action are compatible with the permanence of genuine human life”* (Jonas, 1985).

1.1.2 Organization and contribution of this work

Contribution

This contribution compiles the current state-of-the-art in understanding robotic systems with an emphasis of cognitive-oriented approaches. Furthermore, essential contributions that account for safety issues in this domain are summarized. A general hazard analysis for robots is provided. Due to comparison of already addressed with practical possible hazards, it turns out that there are hazard aspects which are not sufficiently considered so far. For instance, accidents can occur if the robot modifies objects that can hazardously interact with the environment. Such interaction hazards may exist in many operating places, however in particular, they are typical for robots acting in the unstructured human living space.

Dynamic risk assessment (Wardziński, 2008) is considered as the fundamental approach to cope with mentioned problems. Dynamic risk assessment basically is a system monitoring approach by which parts of the hazard identification and assessment are automated in order to provide situational risk information to the monitored system during its operating time. Since cognitive systems are characterized to be equipped with anticipatory and representational capabilities (Strube et al., 2003), the combination of these features with the dynamic risk assessment approach are perceived as a promising step toward the goal to enable the system to be aware of present and forthcoming risks, not pursued so far in other research work for autonomous manipulating robots.

The dynamic risk assessment approach is based on knowledge about hazards. Therefore, besides providing an analysis of possible robot hazards, a procedural model is explicated that aims on systematically formalize hazards, in particular with regard to object interaction hazards. The integration of the safety knowledge in the dynamic risk assessment approach is demonstrated via simulations and robotic experiments. Furthermore, a Markovian approach for realizing a safety clearance within the anticipated action space is contributed.

Learning can be fundamental for achieving autonomy of robotic systems due to lack of sufficient knowledge about its environment (cf. Russell and Norvig, 2010; Fox and Das, 2000; Ridley, 2003); hence, it is one of the essential reasons for the mentioned safety problems. On the other hand, learning capabilities can denote as well an opportunity for safety considerations to cope with so far unknown situations. For this reason, it is taken a closer look at the reinforcement learning and learning from demonstration paradigm in a safety-critical context. Investigations with regard to state-of-the-art reinforcement learning algorithms show that some of them reliably learn (converge) even under worse case parameter settings. In this connection, learning capabilities ‘on board’ can denote a better choice for autonomous systems than depend on (as the case may be) prespecified, mis-designed, 'static' strategies. As a second perspective on learning approaches for safety-critical applications, learning from demonstration is utilized for learning safety knowledge. It is found that learning approaches can also support the safety-related development process.

Organization

Chapter 2 recapitulates fundamental aspects of robotic, cognitive and safety sciences, since this work is located in the intersection of robotics with cognitive sciences and safety.

Chapter 3 outlines the current state-of-the-art of safety approaches in conjunction with robots, and mobile robots. Within that section a variety of approaches is reported considering the collision safety of robots. Hence, it starts with outlining safety measures that are realized via hardware. For taking into account the robot behavior, approaches realized within control theory and planning are outlined. Finally, the contributions in conjunction with abstract high-level safety goals are reported. Here, the concepts of situation awareness and dynamic risk assessment are of special interest for this contribution.

Autonomy is an often and widely used term, also in robotics. With the intention to open up a more comprehensive understanding of the term autonomy, the different perspectives and aspects of the topic autonomy are consolidated in **Chapter 4**. In this connection, autonomy aspects that might be required, and aspects that may become important for future robotic systems and resulting implications for safety are discussed. Specifically, intelligent systems appear to be problematic in conjunction with heteronomous safety measures. However, a 'psychodynamic approach' seems to be well combinable with the dynamic risk assessment concept and, hence, provides an important perspective.

In **Chapter 5**, it is briefly explained the cognitive architecture on which the concept is based on. Here, a conceptual prototype of a cognitive-oriented robot is outlined for which the following concepts and implementations are designed for. Both is realized on basis of the so-called 'SOM approach' which is a meta-modeling technique; hence, it provides a consistent formal description basis.

Since robots are intended to manipulate their environment, far reaching implications with regard to safety may result that are beyond the scope of so far considered collision risks. Therefore, a fundamental investigation of possible robot hazards can be found in **Chapter 6**. Here, the Hazard Theory is applied in order to derive a taxonomic set of hazard classes for robotic systems.

In **Chapter 7**, the observations made so far are summarized. Furthermore, conclusions are drawn in order to define possible safety problems that originate from the utilization of intelligent systems and elements (cf. Fox and Das, 2000; Wardziński, 2008), (Ertle et al., 2010c,a). Furthermore, hazards stemming from interacting objects are assumed to be one of the upcoming problems. To take a step forward, in **Chapter 8** a procedural model is explained that focuses on the development of an initial safety knowledge base considering the object interaction. The systematically gathered safety knowledge shall denote an initial safety measure for autonomous robot systems, which represents a kind of a so-called ALARP[7] approach.

In **Chapter 9**, it is shown how the safety knowledge is applied to a robotic system, and how it is intended to come into effect as safety measure for realizing risk-sensitive planning and perception. This is demonstrated with the help of two simulations and one robotic experiment.

The safety measures of intelligent robots are assumed to be potentially incomplete. Besides being the origin of, or the catalyst for the mentioned safety problems, methods of artificial intelligence can also be seen as chance. Hence, **Chapter 10** presents a perspective on this topic. Here, learning methods with regard to safety-critical applications are investigated.

Finally, in **Chapter 11**, the main contributions are recapitulated within the summary section, while the possibilities are discussed within the outlook section, how the proposed approach can be extended for solving remaining and further problems in future research work.

1.2 Background

1.2.1 Robot science and robotic systems

Robotics is a technology branch and field of applied sciences. Robot science is an interdisciplinary field to which important contributions origin from computer science, engineering, psychology, and biology. Since the 1980s robotics is defined as the science which studies the intelligent connection between perception and action. From that time on, interest were shifted toward robot mobility and manipulability in unstructured environments. This led to the desire to develop robots for improving the quality of life and, therefore, pushed interest toward a realization of service robots. The age of human-centered and life-like robotics begun, in which robots are expected to safely and dependably co-habitat with humans in homes, workplaces, and communities (Siciliano and Khatib, 2008).

A recently new aspect of such robots is the interaction between robot and human. The term ‘human-robot interaction’ (HRI) appears, being subdivided into the areas of cognitive HRI (cHRI) and physical HRI (pHRI). The field of cHRI concerns communication and pHRI the real physical contact between human and robots (Bicchi et al., 2008). In this contribution, the interaction of the robot is more generally considered. Basically, robots interact with their environment. Thus, HRI is considered as a special kind of the robot-environment interaction, because humans are obviously entities of a robot’s environment, besides other entities such as animals and objects. Hence, there are different combinations of interactions, which are systematically examined later on.

[7] As-low-as-reasonably-practicable: A *“given risk can be shown to have been reduced to as low a level as is reasonably practicable, taking into consideration the costs and benefits of reducing it further”* (Ericson, 2011).

Even orderly environments such as the living environment of humans or their working areas appear chaotic for robots, since rules how things are ordered are often abstract and of individual, personal nature. Many robotic applications, thus, were specified for simplified and structured environments. This means that the environment is arranged in such way that the robot is able to reliably fulfill its task. Current technical demands for service and personal robots are vice versa: A robot has to adapt itself on the typical human environment as it is. This is an *unstructured* and *dynamic* environment, which is, in most cases, not fully observable by robot's sensor systems. Hence, truly flexible robots have *"to be able to learn to adapt to [such] partially-known dynamic environments"* (Mahadevan, 1996).

Intelligent robots

An intention to develop truly autonomous, intelligent robots is referred to the science of making machines act intelligently, called 'artificial intelligence' (AI), a branch of computer science. Realization of robots, capable to autonomously operate in unstructured environments, requires for intelligence. An *"intelligent robot is a mechanical creature which can function autonomously"* (Murphy, 2000). With the term 'mechanical creature' it is emphasized that mechanical building blocks are used for realizing robots. Autonomous functioning indicates that robots can operate, self-contained, under reasonable conditions without requiring for help from a human operator. Intelligent means that a robot *"does not do things in a mindless, repetitive way"* (Russell and Norvig, 2010) as it is more typical for factory automation and teleoperation.

There are various definitions of what 'intelligence' is. Russell and Norvig outline that an intelligent agent *"[...] should learn what it can to compensate for partial or incorrect prior knowledge"* (Russell and Norvig, 2010). Another definition that delivers important aspects is: *"Intelligence is the ability to work and adapt to the environment with insufficient knowledge and resources"* (Wang, 2007). This definition considers a system that operates under the *Assumption of Insufficient Knowledge and Resources (AIKR)*. The two components 'adaption' and 'insufficient knowledge and resources' are related to each other: An adaptive system has some insufficiency in knowledge and resources; otherwise, there would be no need for adaptive capabilities, and no need to change. On the other hand, systems without adaptive capabilities, insufficient knowledge and resources would make no attempts to improve its capabilities, hence, it would act as its knowledge and resources were sufficient (Wang, 2006). According to Goertzel and Pennachin (2007); Wang (1995), a system that is acting under the AIKR has to concurrently be

- a finite system, whose computing power, as well as its working and storage space, is limited and constant,
- a real-time system, whose tasks to process, can arrive at any time, and all have deadlines attached with them (including derivation of new knowledge and decision making),
- an ampliative system that not only can retrieve available knowledge and derive sound conclusions from it (deduction), but also can form hypotheses in case of uncertainty (induction),

- an open system, whose old knowledge is not restricted to be extended or replaced by new knowledge, and
- a self-organized system, which can accommodate itself to new knowledge, and adjust its memory structure and mechanism to improve its time and space efficiency, under the assumption that there will be a similarity of past and future situations.

This definition of intelligence considers limited computing power and resources; hence, it is also called *efficient intelligence*, which is *"the ability to achieve intelligence using severely limited resources"* (Goertzel and Pennachin, 2007).

From the given point of view, it can be concluded that one of the core aspects of intelligent robots is to be able to deal with new and, hence, a priori not known situations. In consequence, intelligent robots appear to be knowledge-based systems, whose adequate operation depends on adequate knowledge. Desired functionality, thus, relies on sufficient knowledge as well. Knowledge in turn can not be provided a priori to the system, hence, an intelligent system has to additionally meet the requirement to generate sufficient knowledge in order to be able to perform its task.

Robot architectures

A robotic architecture can be regarded as the abstract design of a class of robots, consisting of a set of structural components with specific functionality and interfaces, it shows the interconnection topology between components (Hayes-Roth, 1995), and can be used to illustrate the essentials of a paradigm (Murphy, 2000). There are three main robotic paradigms, which are the *deliberative/hierarchical*, the *reactive*, and the *hybrid deliberative/reactive* paradigm (Murphy, 2000).

Deliberative/Hierarchical Paradigm First robotic architectures, such as *Shakey* (Nilsson, 1984) were realized in a deliberative/hierarchical manner, which are also assigned to the *Sense-Plan-Act* paradigm, see **Figure 1.5 a)**. In general, the single primitives to process are 'horizontally' decomposed and, hence, processed in a sequential and orderly manner. Sensor observations are usually fused into one global data structure called 'world model'. A world model serves as basis for the planning process. If such a deliberative planner is confronted to model and preplan all eventualities of a dynamic environment, it will suffer from what Arkin calls the *Qualification Problem*: A never ending 'what-if' stream, which would hinder the planner to terminate. Hence, for certain tasks that require rapid response times, a deliberative/hierarchical architecture becomes unsatisfactory (Murphy, 2000).

Reactive Paradigm The drawbacks of systems according to the deliberative/hierarchical paradigm changed the focus of robotisits toward fast adequate reaction times of robots. The concept of reactive or behavior-based robots is generally introduced by Brooks (1986). Brooks coined the perspective on reactive robots by formulating the following key aspects. At first, robots are situated and surrounded by the real world, and they operate on reality itself, on their sensors, rather than on abstract representations. Secondly, robots are embodied, which means that robots have a physical presence (a body).

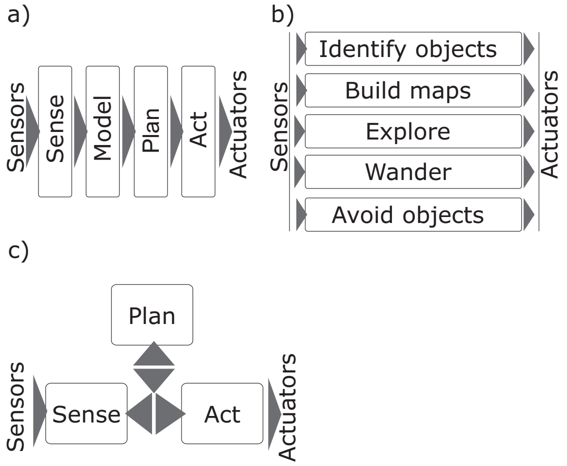


Figure 1.5: a) Sense-plan-act model (deliberative paradigm / horizontal decomposition), b) subsumption architecture (reactive paradigm / vertical decomposition), and c) plan, then sense-act model (hybrid paradigm / vertical and horizontal decomposition) (Arkin, 1998; Brooks, 1986; Murphy, 2000).

This embodiment can not be faithfully simulated with regard to possible consequences of a dynamic interaction with the world. Thirdly, the intelligence or utility of robots emerges in the first place from the interaction of robots with their real environment, including complexity, dynamics, and uncertainty.

These behavior-based systems consist of contextually meaningful units (behaviors) as basic components. Behaviors can be incrementally added to the system in order to increase its utility (Arkin, 1998). In principle, behavior-based systems do not rely on a global world model. Behaviors are lowly coupled because they are independent from each other, and operate concurrently, see **Figure 1.5 b)**. An implementation in hardware, or with algorithms of low computational complexity results, hence, in a tight coupling of sensors and actuators, and limits behaviors to what biologists would call pure stimulus-response reflexes (Murphy, 2000). Two representative reactive approaches are the 'subsumption architecture' of Brooks (1986), and the 'artificial potential field' approaches, introduced into robotics domain by Khatib (1985).

Hybrid Deliberative / Reactive Paradigm On the one hand, there is a consensus that behavioral control is a correct approach for low level control. On the other hand, it is acknowledged as a drawback of the reactive paradigm that the correct assemblage of behaviors for realizing complex tasks, strongly depends on the designer. The concept to enable the robot 'itself' to select behaviors mitigates this design problem, because implemented behaviors can be dynamically combined to complex different tasks in an adaptive manner. However, planning capabilities are required if future aspects shall be taken into account. Architectures which incorporate both, reactive and planning aspects, are assigned to the hybrid deliberative/reactive paradigm (Murphy, 2000). Basically, planning is computational expensive and requires for global knowledge, hence, it is decoupled from the reactive 'act-sense' partition, see **Figure 1.5 c)**. Planning within the hybrid architecture is comprised in a superordinate deliberation layer as an explicit control level.

The meaning of ‘behavior’ in the hybrid paradigm is slightly different in comparison to its pendant in the reactive paradigm, thus, its connotation in the hybrid paradigm is ‘skill’. In general, it is more consistent with the ethological use (behaviors in biology and cognitive science) and, hence, includes reflexive, innate, and learned behaviors. Furthermore, in hybrid architectures are tendentially used assemblages of behaviors sequenced over time, rather than primitive behaviors (Murphy, 2000). A skill can be defined as *“the component in a robot control system that is responsible for the coordinated execution and parameter configuration of the set of available instantaneous robot motion controllers, such that, together, multiple motion controllers let the robot system realize a certain task between a set of modeled objects in the environment”* (Smits, 2010).

For many hybrid architectures, skills have associated events which can be utilized by the deliberative level as explicit control method for bottom-up communication. Vice versa, the global world model, especially for so-called model-oriented hybrid architectures, can support the sensory perception and, thus, serve as a virtual sensor for skills. This has remarkable effect of the performance of robots in terms of reducing sensor errors and uncertainty by sensor fusion (Murphy, 2000).

“The hybrid paradigm has its roots in ethology and provides a framework for exploring cognitive science” (Murphy, 2000).

1.2.2 Cognitive aspects

Cognitive Science

Cognitive science is the interdisciplinary study of the mind, to which many researches are contributing, mainly from domains of psychology, philosophy, neuroscience, biology, linguistic, and computer science (artificial intelligence). Cognitive science originated from problems to explain human capabilities from a behaviorism perspective, which rejected the presence of mental representations. The cognitive paradigm states that cognition is realized via computation, and computation operates on representations. This is based on the assumption that computers and intelligent organisms process and store information in a similar way. Furthermore, the Physical Symbol System Hypothesis (PSSH) (Newell and Simon, 1976), the basis for classical AI and AI-oriented cognitive sciences, states that a physical symbol system is a necessary and sufficient requirement for intelligence.

Considering intelligent organisms and computers as physical symbol systems, the PSSH implies that symbol processing can be considered to be independent from its physical instantiation (substrate) as long as the instantiation can perform the functions required. Consequently, intelligence can be investigated at the level of algorithms and computation processes.

Both, classical AI and cognitive science, consider natural and artificial agents, respectively, as information processing systems. The language of information processing forms a further common denominator for both disciplines (Pfeifer and Scheier, 2001).

Research in cognitive science can be distinguished in two different perspectives, called micro- and macro-cognition. Macro-cognition indicates description of cognitive functions of a cognitive system (e.g. human) under complex real world conditions. Micro-cognition is related to investigation and artificial realization of specific functions that are considered to be the invariant mechanisms forming cognition and determining behavior (Cacciabue and Hollnagel, 1995). Investigation and simulation of cognitive functions and processes, for instance, in artificial agents such as robots, is related to macro-cognition.

A model of the human decision making process

The step-ladder model of Rasmussen (1986) is a well known example of a qualitative information processing model, denoting that there is a sequence of information processing steps when humans perform a problem solving or decision-making task. An important aspect is that 'short-cuts' exist that allow to reduce the amount of cognitive efforts that need to be invested for accomplishing a task (cf. Hollnagel, 1998). Additionally, the step-ladder model is often used to explain human erroneous actions.

In general, the step-ladder model contains different 'states of knowledge', which are provided by 'information processing activities', as illustrated in **Figure 1.6.** As well, it comprises the three different levels of information processing, the skill-based, rule-base, and knowledge-based level. Skill-based behavior represents the sensory-motor level, in which skills are composed by a large repertoire of automated subroutines. Rule-based behavior can be regarded as procedural knowledge, derived from experience, problem-solving or planning. Knowledge-based behavior level is the highest conceptual level. In case when no procedural knowledge (rules) exists, a useful plan can be generated by utilizing a goal description, the mental model of the system itself and its environment.

Cognition and cognitive technical systems

Cognitive systems can be cognitive biological systems or cognitive technical systems. Cognitive systems are characterized by (Strube et al., 2003) to have ability to

- interact in and with their environment (and other cognitive systems as well),
- adapt their acting according to their environment based on internally represented system relevant aspects, and
- feature learning and anticipation capabilities based on information processing.

Furthermore, cognitive systems can be realized by a combination of non-cognitive and cognitive levels. They strongly interact with each other. The underlying non-cognitive layer is formed by basic controls (Strube, 1998).

Cognitive robots

Robots can be considered as technical systems, fulfilling necessary requirements, to be cognitive systems as well. Reiter summarizes the field of cognitive robots: *"The [...] field of cognitive robotics has [...] the provision of a uniform theoretical and implementation framework for autonomous robotic or software agents that reason, act and perceive in changing, incompletely known, unpredictable environments. It differs from 'traditional'*

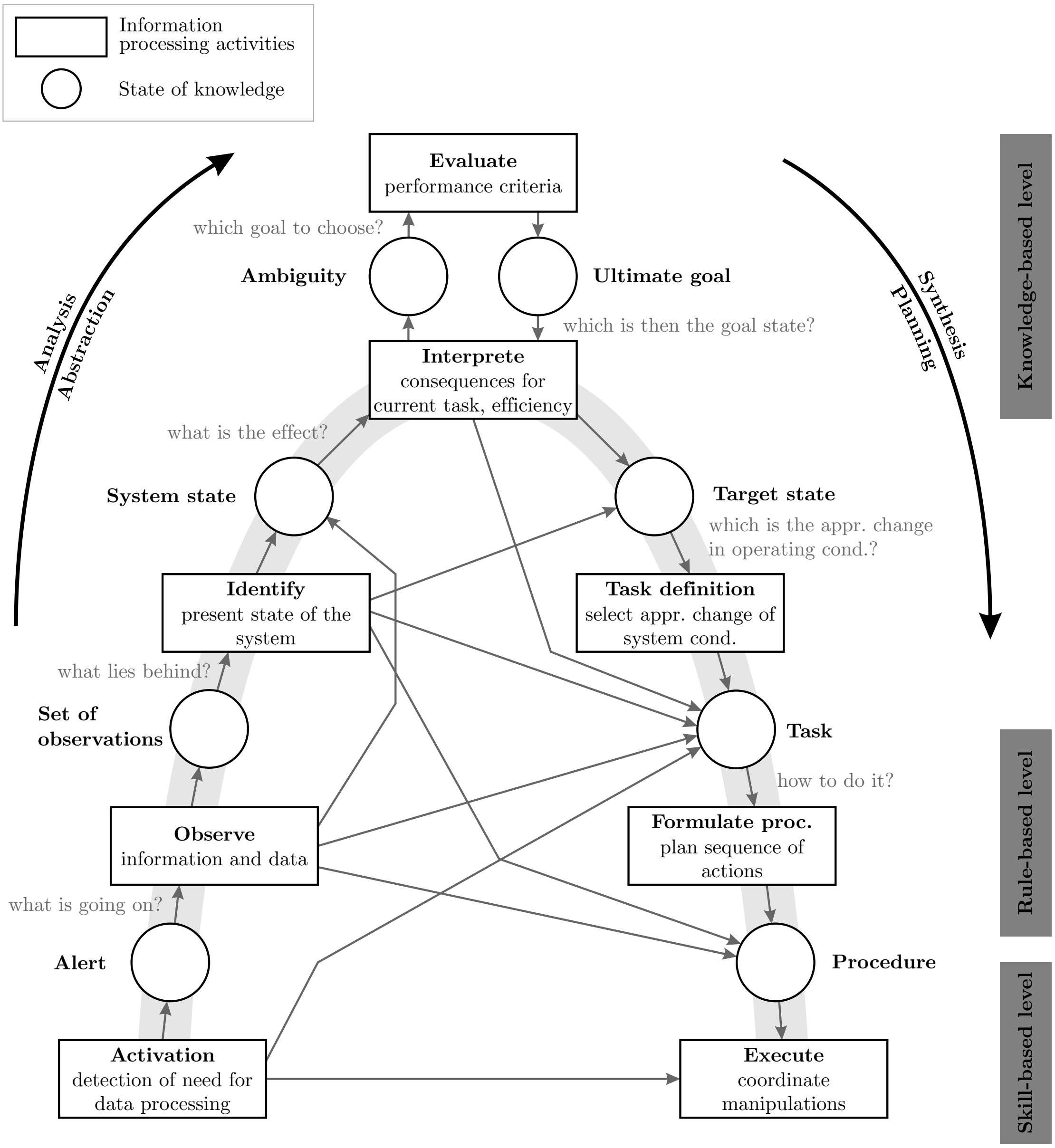


Figure 1.6: Model of the human decision-making process at the three levels of information processing, according to Rasmussen (1986); Hollnagel et al. (1981).

robotics research in emphasizing 'higher level' cognition as a determiner for agent behaviours" (Reiter, 2001).

Cognitive robotics can be regarded 1) as realization of robots with cognitive abilities, 2) as realization of robots based on knowledge and methods of cognitive science, and 3) as robotics, used to inform the field of cognitive science.[8]

Need for cognitive technical systems

Amongst various aspects, the objectives of the AI community are to enable service robots, robots or other systems to learn skills or tasks by environmental or human feedback or by demonstration. Therefore, different systemic functions like perception, vision or learning are needed in order to realize higher functionality by a well-coordinated interplay of these components. They and their interplay have to deal with a huge variety of information with respect to the environment (typical human/domestic environment assumed) and depending on used sensors. As long as processing power of controlling computers is not able to process the full variety of information (if this is ever possible), only a selection of information can be processed. These are the so-called 'relevant aspects' of the full range of information. As cognitive systems are basically characterized by the capability to represent system-relevant aspects of the environment internally (Gamrad and Söffker, 2009a) (in order to process them), it is obvious that robotic systems which are intended to operate in human like (complex) environments are 'somehow' cognitive-oriented systems (Ertle et al., 2010c).

1.2.3 System safety

Safety science is a multidisciplinary field which seeks to ensure rigorousness of theories and methods for research with the aim of understanding and managing unwanted actions or events by developing, experimenting and testing practical methods, tools and models.[9] Safety Science consists of several knowledge areas, such as chemistry, biology, physics, ergonomics, environmental sciences, physiology, business management, economics, sociology, geology, and engineering.[10]

System safety is the engineering discipline (system safety engineering) which aims on developing safe systems and products. It is related to all phases of the system life cycle and covers all system aspects, such as hardware, firmware, software, human operators, and procedures. Generally, it is the process for eliminating or reducing potential accidents (Ericson, 2011). Safety is typically defined as *"[...] relative freedom from danger or the risk of harm [...]"* (Ericson, 2011) (for users, bystanders, environment, etc.).

Stephans reported that the system safety community lacks of standardization or commonality and, therefore, providing 'universally accepted' definitions to even basic terms

[8] according to van Heuveln, Rensselaer Polytechnic Institute, `http://www.cogsci.rpi.edu/~heuveb/teaching/CognitiveRobotics/What%20Is%20Cognitive%20Robotics.html` [online; accessed 23-March-2012]

[9] Call for papers - Special issue on the foundations of safety science. Safety Science, 50(7),2012, I-II

[10] What is safety science: `http://www.asse.org/newsroom/presskit/docs/ASSE_she_career_broch_FNL.pdf` [online; accessed 31-May-2012]

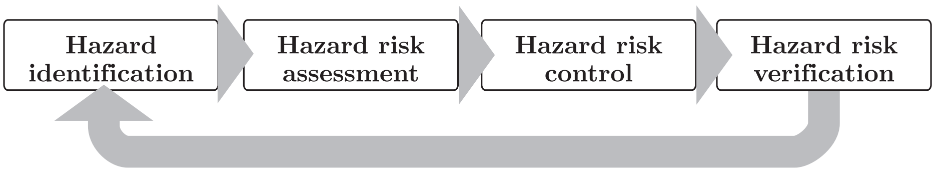


Figure 1.7: The general elements of the system safety process, according to Ericson (2005).

is difficult (Stephans, 2004). System safety is considered as a discipline which is primarily concerned with new systems with the attempt to identify potential hazards while the system is designed. Therefore, safety is reasonably not added onto a completed system design. It has to be considered throughout the life cycle of a system, being already a part of initial concept development (Leveson, 2003). In most cases, safety is in conflict with other design goals, such as operational effectiveness, performance, time, and costs (Ericson, 2005; Leveson, 2003).

Safety is understood as an emergent property of a system that can be determined only in the context of the whole system; safety appears, and must be controlled at the system level. Consequently, system safety is closely coupled with system theory, since it considers the system as a whole rather than single subsystems or components (Leveson, 2012).

System safety focuses on eliminating and preventing of hazards. A hazard is a potential accident, and an accident is an actuated hazard. Actuation of a hazard is the process of a hazard turning into an accident; the hazard as potential condition state transients into the accident as event state. An accident is an actual event that has occurred and has resulted in an undesired outcome (Ericson, 2011).

For system safety, hazards and not failures are the primary concern. In contrast, *reliability engineering* primarily focuses on failures; hence, its contribution to safety considers accidents that are caused by component failures. Thus, reliability engineering contributes to safety, whereas safety has a broader scope: There are accidents that occur despite components that operate exactly as specified. Typically, reliability applies bottom-up approaches in order to anticipate component failure effects on system overall functionality. For safety considerations, top-down approaches are required for evaluating how hazardous states can be provoked by correct and incorrect behavior of components (Leveson, 2012).

The course of action taken to achieve safety is called 'system safety process'. The general *system safety process* consists of consecutive steps which are arranged in a closed-loop manner, during the entire life cycle of a system. As illustrated in **Figure 1.7,** the general elements are 'hazard identification', 'risk assessment', and 'risk control'. These terms are briefly detailed in the sequel, based on the definitions of Ericson (2011).

Hazard identification is the process of recognizing hazards, its consequences, and it is achieved through hazard analysis. Hazard identification denotes the basis for the 'hazard risk assessment'.

In a *hazard risk assessment*, the likelihood of identified hazards is determined in terms of probability, frequency, or qualitative criteria. Both together, likelihood of occurrence

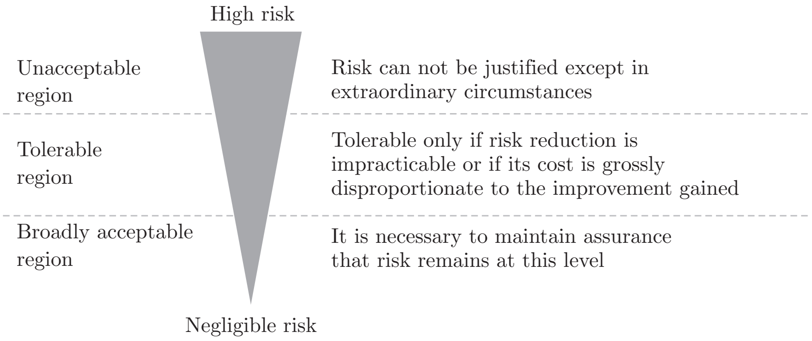

High risk
Unacceptable region
Risk can not be justified except in extraordinary circumstances
Tolerable region
Tolerable only if risk reduction is impracticable or if its cost is grossly disproportionate to the improvement gained
Broadly acceptable region
It is necessary to maintain assurance that risk remains at this level
Negligible risk

Figure 1.8: The ALARP model: The triangle represents the diminishing risk proportion, according to Ericson (2011).

and severity of consequences form the concept of a safety measure, called *hazard risk*. In the following, it is referred to hazard risk as risk. *Risk* is described by a triplet, $< sc_i, P_i(\phi_i), P_i(X_i) >$ (Kaplan and Garrick, 1981). The elements are, at first, the potential future event, sc_i (with $i \in \{1, \ldots, n_{scenario}\}$), secondly, the likelihood, $P_i(\phi_i)$, of its occurring, and last, the potential consequences, $P_i(X_i)$, when it occurs. Each of these aspects involves an element of uncertainty (Ericson, 2011). Hence, risk is a measure of the future event, where the event is an expected accident. From that follows, considering the operation phase of a safety-critical system, that in every situation there exists a list of hazards (with a certain length) with different amounts of risks.

Hazard risk control is the mitigation of risks that were identified to be unacceptable. A common strategy is called the 'safety order of precedence': First it shall be tried to eliminate hazards through design alternatives, if not possible, risks shall be reduced (Ericson, 2011). *Risk mitigation* is the strategy to reduce potential risks including methods that are applied to achieve this. Risk mitigation involves the establishing and implementing of so-called 'design safety features'.

Design safety features are intended to reduce the likelihood and/or severity of hazards. A design safety feature is a synonym and interchangeably usable for safety feature, safety measure, safety mechanism, and hazard countermeasure (Ericson, 2011). It *"is a special and intentional feature in the design of a system or product employed specifically for the purpose of eliminating or mitigating the risk presented by an identified hazard. A design safety feature may not be necessary for system function, but it is necessary for superior safety (i.e., risk reduction). A design safety feature can be any device, technique, method, or procedure incorporated into the design to specifically eliminate or reduce the risk factors comprising a hazard"* (Ericson, 2011).

Risks that are remaining after risk mitigation are called *residual risks*. Residual risks that are acknowledge to persist without further risk reduction for the remaining system life cycle (e.g. operating time) are called *acceptable risks*. Acknowledgment of risk acceptance takes place by the 'risk acceptance process'.

The *risk acceptance* is finally a decision that the potential accident risk presented by a hazard is known, understood, and acceptable. An accepted risk is not necessarily the lowest risk possible, it may also be a risk labeled as unacceptable. In this case it must be accepted for various reasons such as mission needs, for instance, and it must be considered that the system user is consciously exposed to this risk. This is usually explicated with the help of the ‘ALARP’ model. ALARP is an acronym for ‘as low as reasonably practicable’. *“If a given risk can be shown to have been reduced to as low a level as is reasonably practicable, taking into consideration costs and benefits of reducing it further, then it is said to be a tolerable risk”* (Ericson, 2011). The ALARP model is illustrated in **Figure 1.8.**

2 State-of-the-Art in Robotic Safety

2.1 Safety Standards

Norms and standards are given and agreed conventions to ensure that minimal safety level is obtained before products are launched. For the considered kind of industrial robots, the norm DIN EN ISO 10218 (2009) has to be considered. For non-industrial robots currently no analogous norm exists. The ISO/DIS 13482 (2011), currently available as a preliminary draft version, is to become a norm for non-industrial and non-medical personal care robots. Basically, all safety norms and standards are based on risk assessment. The risk assessment and risk reduction process is described in the DIN EN ISO 12100 (2004). Accordingly, all hazards have to be identified, analyzed, and taken into account for each specific application. Corresponding to the DIN EN ISO 12100, a three step approach for mitigating hazards has to be carried out: First of all, inherently safe design measures have to be considered. These protection measures are intended to eliminate hazards. Secondly, hazards that can not be sufficiently mitigated in the first instance require to be considered by the implementation of safeguards. Finally, users have to be warned about the remaining risk that can not be mitigated through the two latter measures.

According to the DIN EN ISO 12100, norms are classified as norms of type A,B, or C. Type-A norms are very general, the specificity increases for type-B to type-C norms. If there are deviations within different applicable norms, the more detailed type-C norms have precedence over type-B norms, and consequently, type-B over type-A norms. The DIN EN ISO 12100 is a type-A norm, the ISO/DIS 13482 and the DIN EN ISO 10218 are type-C norms.

A list of typical hazards is provided in ISO/DIS 13482. These hazards can be classified as mechanical, electrical, thermal, emission-like, environment-related, and controlling-related, or a combination of them (ISO/DIS 13482, 2011). The hazard that is predominantly taken into account in literature is of mechanical nature: Collisions between robots and humans. The field of physical Human-Robot Interaction (pHRI) is driven by applications in which robots cooperate with physical contact to humans, or share the workspace with them, so-called hands-on, or hands-off pHRI (Bicchi et al., 2008). Following this paradigm, several mechanical inherent safe designs for robots are developed. A mechanical safe design is intended to compensate for user failures, and especially, for inadequate control of robot.

2.2 Determining the Injury Potential of Collisions

At first, a quantitative measure is required to determine the hazard potential of collisions. Inspired by investigations in the field of automotive crash tests, the *Head Injury Coefficient* (HIC) is adopted as an adequate measurement. The HIC represents a measure for acceleration of the human head during a collision. It *"[...] has been validated as a predictor for skull fracture and brain injury of certain severities"* (Gao and Wampler, 2009). Based on the HIC, there are various investigations of collisions considering a multitude of different configurations. Blunt impacts are investigated (Haddadin et al., 2008; Park and Song, 2009), singularity poses of robot arms are examined (Haddadin et al., 2010), or pneumatic actuated robot arms are taken into account (Damme et al., 2010). Further investigations are concerned with material properties, shape, acceptable velocity of the robot, and so forth (Wassink and Stramigioli, 2007), to mention only few.

Newsworthy research work considers the injury potential of different object shapes. Injury via different primitive shapes is evaluated by Haddadin et al. (2012), which are a wedge, a small, and a large sphere. Therefore, drop-test experiments with these shapes on pig skin are realized. Afterwards, a set of professionals is evaluating the injuries according to an acknowledged classification. Hence, the injury potential for collisions of different primitive shapes is known in dependence of its respective kinetic energy. The injury potential for closed skin injury, muscle and tendon injury, and neurovascular injury is experimentally determined in this manner.

2.3 Mechanical Safe Robot Design

2.3.1 Robot shape

The contour of the robot as potential colliding geometry plays an important role. Collision energy can be distributed and absorbed by a soft covering (Suita et al., 1995) or chamfering (Ikuta et al., 2003). Investigations with regard to HIC and covering are reported by Zinn et al. (2004).

2.3.2 Reduction of colliding masses

It is acknowledged that the colliding mass is a further important safety-relevant factor. In this connection, there are essentially four basic approaches that contribute to additional safety features. Usually, these are classified as the so-called *post-collision strategies* (Heinzmann and Zelinsky, 2003).

At first, lightweight design can be realized by utilizing modern light-weighted materials.

Secondly, lightweight actuators, such as pneumatic muscles can be preferred. Pneumatic muscles can be designed as inherently compliant actuators. A combination of lightweight design and lightweight actuators is realized in the *Bionic Handling Assistant*.[1] The so-called bionic inspired approaches are realized by copying of organic designs. The *Bionic Handling Assistant* is an ultra-lightweight compliant manipulator.

Thirdly, heavy actuators can be relocated from the manipulator's joints to the robot base. In the *Whole Arm Manipulator WAM*[2] or *Dexter* (Zollo et al., 2002), the actuators are connected via wires to the manipulator's joints. Extending this concept, in the *Distributed Macro-Mini Actuation (DM²)* approach additional small and lightweight actuators are integrated in the manipulator joints for high frequency actuation (Zinn et al., 2004).

Finally, the effective mass that is involved in a collision can be reduced by compliant actuation. The predominant part of the mass of compliant actuators is decoupled from its output shaft. This is effecting on the one hand that the inertia of rotating elements of the actuators, additionally amplified by gear mechanisms, is not directly involved in collisions. Furthermore, serial connected compliant actuators, in a multi-joint robot manipulator, for instance, decouple each, its own and the preceding moved masses (e.g. the rest of the robot) from its output shaft.

2.3.3 Passive compliant actuation

Compliant actuators reduce the collision mass on the one hand, on the other hand, a manipulator with compliant actuators is 'soft' if external forces are applied to it. Consequently, the manipulator can be manually deflected by a user. This feature can as well increase the acceptability in contrast to stiff robot manipulators.

Compliance can be realized actively by control, or passively by compliant elements. Active compliance relies on the control system and its sufficiently low control latency time; passive compliance is inherently available, according to its mechanical properties and reliability.

As passive compliant actuation is realized by integrating elastic elements into actuation chain, it is called ***S**eries **E**lastic **A**ctuation* (SEA). SEA is not suitable for mere precise position control. For precise position control traditionally stiff constructions are preferred. Furthermore, SEA can become instable at its resonance frequency. As the output shaft is decoupled by elastic elements, oscillations are not controllable by the actuator itself. The actuation system shows an open-loop behavior; hence, it can not be considered being safe. Thus, there are six basic strategies how the fundamental SEA actuation can be improved, considering mentioned drawbacks. The latter four are to assign to the so-called *adaptive compliance* or ***V**ariable **I**mpedance **A**ctuation* (VIA) approaches.

Equilibrium-controlled stiffness

This actuation approach is an extension of the SEA principle. Deformation of the spring, being mounted between the joint and the stiff actuation, is measured. Thus, an

[1] `http://www.festo.com/cms/en_corp/9655.htm` [online; accessed 29-June-2012]

[2] `http://www.barrett.com/robot/products-arm.htm` [online; accessed 27-March-2012]

active control of the exerted force is realized; the position control is changed to force control (see **Figure 2.1a**).

DM^2-actuation

On the one hand, heavy actuators are relocated from the robot manipulator joints to the robot base. On the other hand, they are connected as SEA (see **Figure 2.1b**). Lightweight actuators, mounted in the joints, allow controlling the instability of the standard SEA (Zinn et al., 2004).

Force limitation

The *Safe Link Mechanism* (SLM) limits the maximum force that can be transferred via a link. It is a passive redundant element which does not affect the conventional actuation strategy (see **Figure 2.1c**). If the force limit is exceeded, the link becomes compliant (Park et al., 2007).

Structural-controlled stiffness

The variable stiffness of structural-control actuation is realized by taking into account the physical bending properties of a bending beam or spring. The bending beam changes its bending properties with respect to the bending direction or the effective length of beam (or spring). Therefore, the control adjusts the bending orientation or the effective length (see **Figure 2.1d**). The ***H**ybrid **D**ual **A**ctuator **U**nit* (HDAU) varies the length a beam that presses on a spring (Kim and Song, 2010).

Antagonistic-controlled stiffness

In this approach, the stiffness is varied by controlling the preload of two antagonistic coupled (non-linear) SEAs. The higher the preload of both antagonistic springs, the higher is the stiffness of the actuator (see **Figure 2.1e**). The Parallel Dual Action Unit (PDAU) is a compact realization of the antagonistic-controlled stiffness approach, including a force limitation mechanism (Nam et al., 2010).

Mechanically controlled stiffness

The improvement in contrast to the SEA is that, due to the construction, the preload of a spring AND its equilibrium position can be changed. The spring behaves like adjustable torque spring with a variable equilibrium position (see **Figure 2.1f**). This is also called the *MACCEPA* approach (*Mechanically Adjustable Compliance and Controllable Equilibrium Position Actuator*). The *Variable Stiffness* joint (Wolf and Hirzinger, 2008) is a compact realization if the MACCEPA approach.

2.3.4 Summary

Finally, several approaches are available that allow to realized inherently safe actuators, considering the hazard potential of collisions between robots and humans. At this, the

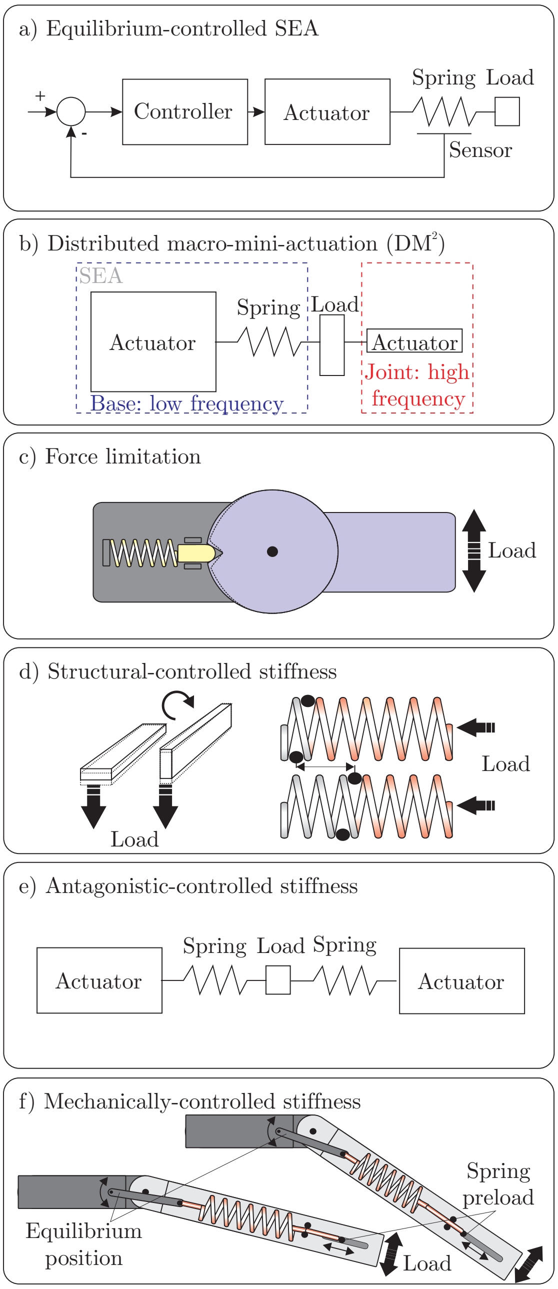


Figure 2.1: Different main concepts for safe and compliant actuators (cf. Ham et al., 2009; Park et al., 2007).

risk of hazardous collisions might be eliminated at least partially if not completely. Certain risk remains if high forces or velocities are required for the task mission because the compliance adjustment has to be realized by control. Hence, the safe and reliable control is shifted to a certain extent from mechanical toward software-related aspects.

2.4 Reactive Safety Approaches in Robotics

There are various examples for which an inherent mechanical design can not be realized, for instance, the robot application requires that the robot possesses potential critical forces or movement velocity. In consequence, protective functions have to be realized that ensure adequate and safe behavior of the system. With regard to the collision of the robot with safety-relevant obstacles (humans, pets etc.), the norm ISO/DIS 13482 requires for one or more protective measures with regard to pre-collision strategies, the avoidance of collision in general, and post-collision strategies, the mitigation of harm in case of collision. Both can be realized via control and reactive planning. Reactivity emphasizes low reaction time, small time-horizon and focus on the present moment, and strong coupling to sensor data. Control refers to the consideration of error signal, determined by deviation of a feedback signal from its set point. For control quality, cost functions can be taken into account. Reactive planning incorporates a planner for the generation of plans, which are used as reactions to predefined situations (Vlahavas and Vrakas, 2005).

The behavior of a robot is controlled by several instances within robot architectures (see Section 1.2.1). At the lower architectural level, such reactive aspects are met. Skills represent the reactive functionality.

In general, post- and pre-collision strategies can be classified as either binary or gradual strategy (Kulić and Croft, 2007). The binary, the so-called *safeguarding* strategy principally consists of detecting the human's presence in the robot's working range and altering the robot overall operation mode, for instance, to safe slow motion. The gradual, the *danger evaluation* strategy aims on adequately adjusting the robot behavior based on a continuous expression of danger.

Different criteria for collision safe robots should be mentioned. According to Fraichard (2007), three basic criteria have to be taken into account to ensure safety: 1) The robot dynamics, 2) environment object dynamics has to be considered, and 3) this for an infinite (or goal-related) time horizon. Dautenhahn et al. (2006) propose three different criteria for safe (and user-comfortable) robots: 1) The safety criterion, which focuses on the collision-free approaching of robots to humans. 2) By the visibility criterion, it is demanded that the robot operation should be performed (if possible) in the human's field of view. 3) 'Shaded' areas behind obstacles are insidious and should be avoided by the robot.

2.4.1 Safety integrated into robot manipulator control

Post-collision strategies are also mentioned in Section 2.3 with regard to inherent mechanical design. With the help of real-time control systems, a collision detection system can be sufficient, if it reliably reduces the collision energy. Such an approach is suggested by Yamada et al. (1997). The robot arm itself is covered with viscoelastic material, which does not only mitigate the contact force. In fact, the time during a collision, which is needed for deforming the elastic coverage with uncritical forces, is used for detecting the contact. In consequence, a very fast detection control reduces the velocity of the robot or triggers an emergency stop. Furthermore, the collision energy should not exceed an acceptable level. In Section 2.3 is argued that the stiffness is related to the effecting collision mass. Stiffness can be also affected by control, thus, Lew et al. (2000) consider the effect of control parameters on stiffness. They state that control parameters are commonly optimized for realizing good tracking and accuracy, collision issues are not considered and, hence, high stiffness and high inertia results. Therefore, Lew et al. (2000) propose to take into account these effects, and show how the control parameters are related to the stiffness.

In order to improve the emergency stop state, Heinzmann and Zelinsky (1999) introduce the concept of the *Zero-G* controller. The Zero-G mode is proposed as a preferable failsafe solution. In the Zero-G mode all gravity effects are compensated; therefore, the manipulator remains fully compliant and easily movable for the user. This provides advantages in comparison to 'freezing' arms caused by emergency brakes or 'breakdowns' due to mere de-energizing.

Karlsson et al. (2000), for instance, reduce the robot velocity with regard to the distance between robot and human. Heinzmann and Zelinsky (2003) and Matsumoto et al. (1999) present an adequate measure for robot velocity reduction by utilizing the impact potential as function for an allowable velocity. The impact potential is the force, which is exceeded at the moment of contact. With their approach the entire robot can be limited in such way that no hazardous impact force can occur. An additional advantage is the overcoming of the drawback of passive compliant actuation approaches (SEA see Section 2.3.3), which are limiting the collision overall energy but not the impact force.

Traver et al. (2000) introduce the *elusive robot*. An *elusive strategy* is characterized by actively avoiding collisions with humans. Goal-directing forces, directing the manipulator toward the desired position have to be balanced with repulsive forces that intend to maximize the distance to humans. They outline the basic concept that repulsive forces can be generated based on distance information, pose or velocity of the robot, or with regard to the current moving or viewing direction of a human user.

An essential basis to consider hazards by continuous measures is given by Ikuta et al. (2003). They initially suggest utilizing the concept of *danger indices*. As already mentioned, danger indices are numerical and normalized expressions of hazard potentials. They can be applied to both, design purposes and to control schemes. Danger indices may describe hazard potentials with regard to various contributing factors, such as impact force, relative distance, velocity, robot inertia, stiffness, and so forth. However, a control strategy is not detailed by Ikuta et al. (2003), therefore, Kulić and Croft (2006) realize this in their research work. They use and refine the proposed hazard factors based on the distance, velocity, and inertia with respect to the 'critical point' (the point closest to human). The danger index is the product of different hazard factors (see **Figure 2.2)**. With increasing hazard potential, the value of the danger index increases. Based on the danger index, a repulsive force is generated when the distance falls below a minimal value.

Lacevic and Rocco (2010) extend the danger index approach of Ikuta et al. (2003), and Kulić and Croft (2006). They introduce the so-called *kinetostatic danger field*. The kinetostatic danger field is similar to the well-known potential fields, introduced in the robot domain by Khatib (1985). The kinetostatic danger field contains repulsive forces with regard to the distance to obstacles, their velocity, and the velocity vector of the

$$DI = f_D \cdot f_V \cdot f_I$$

$$f_D(d) = \begin{cases} k_D \left(\frac{1}{d} - \frac{1}{D_{\max}} \right)^2 & : d \leq D_{\max} \\ 0 & : d > D_{\max} \end{cases}$$

$$k_D = \left(\frac{D_{\min} \cdot D_{\max}}{D_{\min} - D_{\max}} \right)^2$$

$$f_V(v) = \begin{cases} k_V (v - v_{\min})^2 & : v \geq v_{\min} \\ 0 & : v < v_{\min} \end{cases}$$

$$k_V = \left(\frac{1}{v_{\max} - v_{\min}} \right)^2$$

$$f_I(I_{CP}) = \frac{I_{CP}}{I_{\max}}$$

d	: critical distance
$D_{\min}$	: smallest allowable distance
$D_{\max}$	: harmless distance
v	: action velocity
$v_{\min}$	: harmless velocity
$v_{\max}$	: max. velocity
I_{CP}	: effective inertia
$I_{\max}$	: max. allowable inertia

Figure 2.2: Danger Index *DI* according to Kulić and Croft (2006).

entire robot manipulator. A control method is proposed, which decreases the overall hazard potential, and on the other hand, allows for deviations from the task if a certain hazard potential threshold is exceeded.

Najmaei and Kermani (2011) apply the danger index as well. Repulsive forces, similarly based on the danger index, are generated also by taking into account the movement of a human user. Position and direction of humans are detected with the help of sensitive flooring. With this, a neural network is trained for predicting the human movement. The predicted human movement is considered, in turn, by the movement of the robot; an elusive robot is realized.

2.4.2 Safety integrated into reactive robot platform motion planning

Alami et al. (2002) focus on the safe motion of the entire robot platform. It is shown how *velocity profiles* for motion trajectories can be generated in order to exclude all constellations in which the robot is incapable to avoid collisions by evading or stopping (assuming that also unknown dynamic obstacles never exceed a known maximum velocity threshold). Therefore, sensor limitations and dynamics of obstacles are considered.

Additionally, it is taken into account that dynamic objects can be hidden behind shaded areas of detected obstacles. Velocity profiles, also including collision safety, are also examined further by Madhavakrishna et al. (2006).

The concept of *inevitable collision states* (ICS) is introduced by Fraichard and Asama (2004). ICS are constellations in which the robot is unable to adequately react because of its own or environment object's dynamic. Fraichard and Asama define a motion plan to be safe if it comprises no inevitable collision states. Petti and Fraichard (2005) realize the ICS concept for fast motion planning approaches. In order to realize short planning delays, partial motion plans are periodically generated (Rapidly-Exploring Random Tree method). Branches that comprise inevitable collisions are rejected. Bekris and Kavraki (2007), as another example, used the ICS concept for motion planning as well (*GRIP-motion planner*).

Althoff et al. (2012) use the ICS concept as basis in order to extend it to a probabilistic version, called Probabilistic Collision States (PCS). While the ICS concept classifies trajectories as such that inevitably contain collisions, and others that may not contain collisions, the PCS concept postulates the trajectory collision probability. The PCS concept is applied to partial motion planning. Here, it is taken into account the collision probability of the trajectory candidates within the planning horizon, on the one handand on the other hand, the collision probability of the respective final state as collision indicator beyond the planning horizon. With this approach, collision free trajectories are found taking into account the dynamics of environment objects, including their interaction.

Altogether, there are various approaches to concern safety for manipulator or platform trajectories. A more general approach toward autonomous operation of mobile robots is pursued by Seward et al. (2000, 2007). Basically, they develop the robotic excavator LUCIE which is intended to act autonomously. They explicitly perceive the safety problem to be related to an unstructured environment to which mobile robots are often interfaced with. In this regard, the complex functionality of robots is additionally sensitive to such an environment, which increases the complexity of safety analysis. However, autonomous systems basically offer the chance, as outlined in Section 3, to adjust their operation with regard to various aspects, including safety. Seward et al. (2007) call this the 'self-safety management'. Their central approach to the safety problem is that the self-safety management has to be become an inherent part of the system behavior. They favor a centralized safety system which is responsible for the safe operation of the overall system. The safety system is based on a 'self-risk evaluation'. For realizing an operational risk evaluation, Seward et al. (2000) model risks utilizing fault trees. These fault trees map observed system and environment states that account for a specific hazardous top event to the probability estimate of its occurring. The toppling hazard is explicitly regarded as top event while digging, and the collision hazard is regarded as top event while traveling. The process model is shown in **Figure 2.3.** In general, the system states are modeled with partially observable Markov decision processes (POMDP) in order to consider uncertainty[3] with regard to observing the respective system state. Actions taken are assumed to transfer the system into the

[3] epistemic uncertainty due to system subjective classification of system states (cf. Beer et al., 2013)

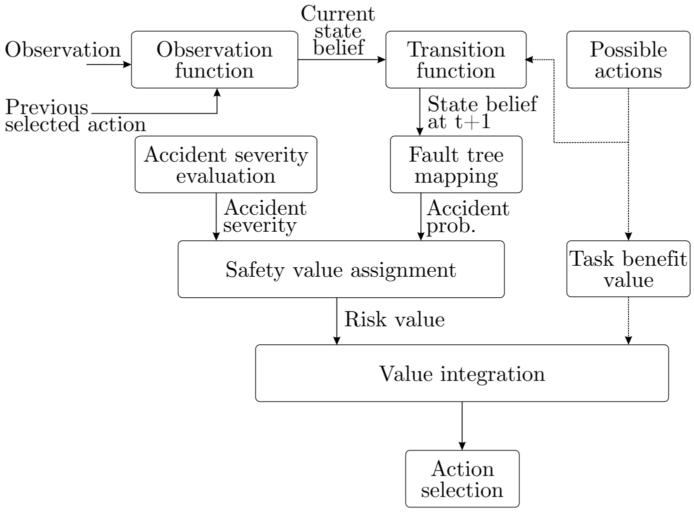


Figure 2.3: Process model of safety risk assessment and management (Seward et al., 2007).

subsequent system state with a certain probability. With the help of specified fault trees, the probability of a specific accident can be determined from any system state information. By computing the product of this probability and the accident severity, the corresponding risk value results. Afterwards, a value integration component balances the risk value and a benefit value. Finally, an appropriate action may be chosen. In **Figure 2.4,** an exemplary fault tree for the top event of excavator toppling is shown.

2.4.3 Summary

Several contributions focusing on proper reactive behavior are available in order to mitigate collision dangers for several robot motion approaches, addressing mobile platforms, manipulators, or both. There are approaches that merely alter the operation mode of the robot but there are also approaches that reactively alter the behavior of the robot, for instance, based on a velocity profile, a danger index or on the modeled risk (probability) for a specific event. Finally, there are a remarkable number of concepts how motion control can be modified or extended in order to improve safety.

The preliminary version of the safety standard ISO/DIS 13482 (2011) requires considering collisions with safety-relevant obstacles by implementing protective measures, whereas the mentioned concepts denote a good basis to approach toward fulfillment of these requirements.

2.5 Safe High-level Reasoning and Planning

latter sections reviewed contributions for safety at a lower (reactive) systemic level. The majority of these are concerned with collisions. This appears to be a logic consequence, since collisions are inherently related to control of motions. Reactive portions of

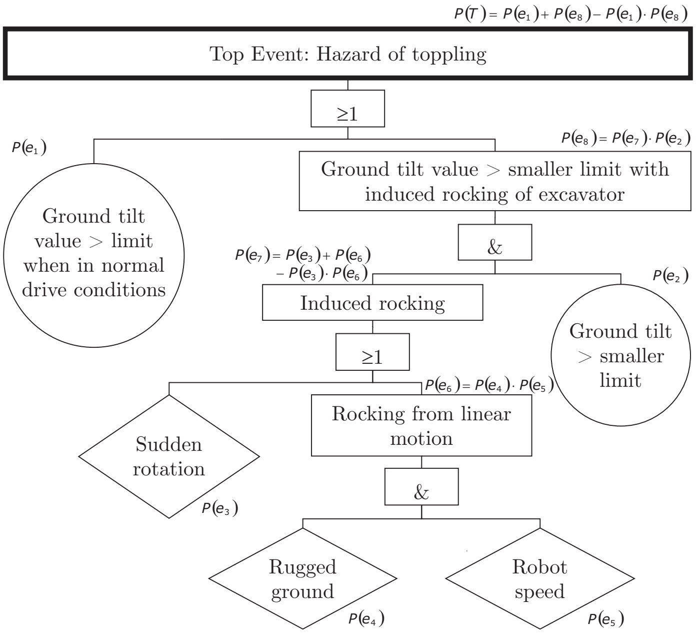

$P(T) = P(e_1) + P(e_8) - P(e_1) \cdot P(e_8)$
Top Event: Hazard of toppling
≥1
$P(e_1)$
$P(e_8) = P(e_7) \cdot P(e_2)$
Ground tilt value > smaller limit with induced rocking of excavator
Ground tilt value > limit when in normal drive conditions
&
$P(e_7) = P(e_3) + P(e_6) - P(e_3) \cdot P(e_6)$
$P(e_2)$
Induced rocking
Ground tilt > smaller limit
≥1
$P(e_6) = P(e_4) \cdot P(e_5)$
Rocking from linear motion
Sudden rotation
$P(e_3)$
&
Rugged ground
$P(e_4)$
Robot speed
$P(e_5)$

Figure 2.4: Example of a fault tree used for mapping system states into a hazardous event, according to Pace and Seward (2005).

robot control are closely related to sensors and actors. The time horizon of the lower reactive system levels is about the past and the present. The upper level, responsible for deliberation and planning, reflects about future aspects, and works with symbols (Murphy, 2000). Classical cognitive and AI paradigms acknowledge the Physical Symbol System Hypothesis. Here, the action performed by a system is a causal consequence of the symbol processing (Vernon et al., 2007). Langley (2005) formulates the Symbolic Physical System Hypothesis which claims that the mental states of embodied agents should always be grounded in real or imagined physical states, and that problem-space operators (high-level action representations) always expand to primitive skills with executable actions. This means that actions encoded by symbols at the symbolic level are representing skills or assemblages of primitive skills. Such skills typically might be of the kind of 'pick object A and place it at B', or 'drive to B', for examples see Bischoff and Graefe (2004); Mosemann and Wahl (2001); Söffker and Ahle (2008).

Planning and reasoning that is based on abstract terms and concepts is often referred to as high-level planning and reasoning (Coradeschi et al., 2006). Such planning is not meant to replace existing motion or path planning methods, mentioned in the latter chapter, but to complement them for more general and complex planning of robot action. In this connection, planning means deliberation about a course of action to take for achieving a given set of goals. In this respect, notable key requirements are the representation of actions and their effects, keeping track of changes in the environment over time, and dealing with any remaining uncertainty or lack of information (Hertzberg and Chatila, 2008). Uncertainty implicates two key problems with regard to planning. At first, the outcome of certain actions is not necessarily predictable. Secondly, the current state, obtained from initial conditions, sensors, and memory of previously applied actions, is not necessarily known (LaValle, 2006). Uncertainty of action outcome, both, its indeterminism and the vague knowledge of the action's starting situation, may implicate severe consequence with regard to safety.

Basically, planning and decision making are found to be interrelated methods. While decision theory focuses on single decisions with often multiple related factors, planning considers searching for problem solutions in huge planning spaces.

In the robotic domain little contributions seem to be currently available that reflect on safety issues at this level; in some other domains there are notable concepts.

2.5.1 Risk in Utility Theory

Utility Theory is a major branch of decision theory, therefore, contributions focusing on risk-sensitive decision making are found to be often based thereupon. An overview to decision-theoretic planning is given by Blythe (1999), and risk-sensitive planning is considered by Koenig and Simmons (1994); Goldman and Boddy (1994). In general, decisions in Utility Theory are based on the so-called utility function. A utility function subsumes the decision criteria as numerical costs and rewards, hence, the utility function maps an expected outcome of an action into a numerical utility expression. In consequence, proper decisions are made by maximizing the utility reward.

The concept 'risk' in this domain relates to the variance of the expected utility value (considering probabilistic decision making). For instance, the 'risk-seeking' or 'gambling agent' favors ('speculative') decisions with high utility besides low probability. A 'risk-avoiding agent' favors decisions whose outcome is reliable (Koenig and Simmons, 1994). In this connection, the risk is related to likelihood of costs according to the utility function; hence, it is defined similar in comparison to numerical risk expressions in system safety theory, which is the product of probability and a measure of damage (Kaplan and Garrick, 1981). Furthermore, the inherent tradeoff problem between performance and safety in real environments may involve multiple factors. Thus, utility theory may serve as fundamental conceptual framework for risk-sensitive decision making. In this case, it must be ensured that the accident severity is throughout represented in the utility function. Additionally, it is assumed that different utility functions are required, because the robotic system may pursue different tasks.

Several aforementioned approaches implicitly include aspects of utility-based decision making. For instance, Seward et al. (2007) introduce a component for action value integration that balances risks and benefits in order to select proper actions. The danger index of Ikuta et al. (2003); Kulić and Croft (2006) are a representation of danger as numerical expression, which could theoretically be normalized and balanced against benefits of performed actions. At the higher system level, the maximal danger index of a planned manipulator trajectory could be compared to the movement benefits in order to permit or prohibit the performing of the movement action at all.

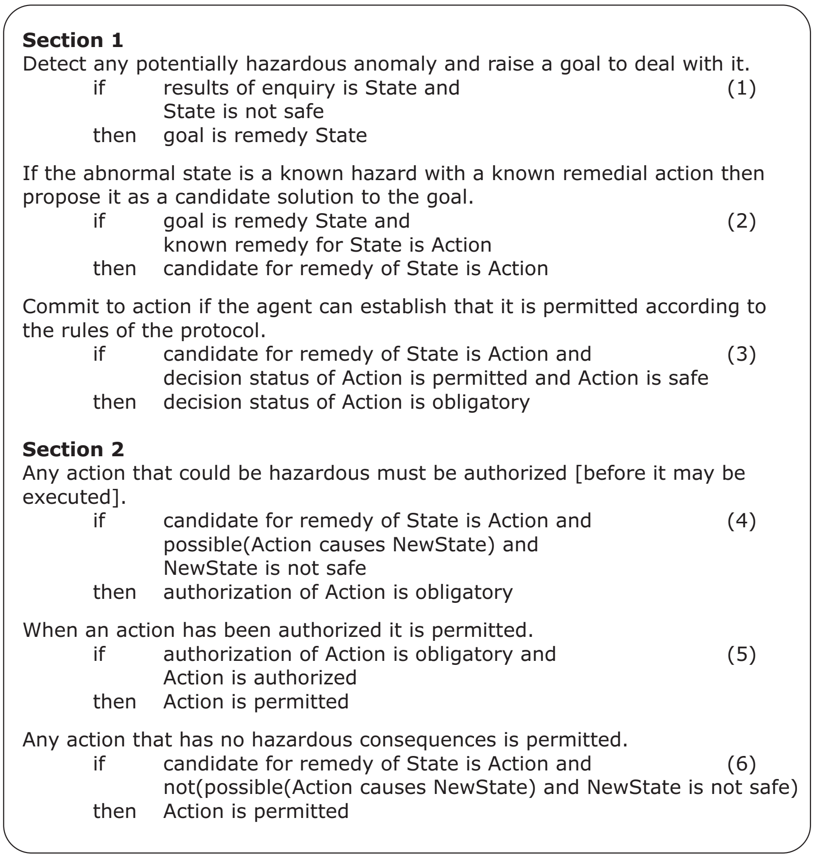
2.5.2 The safety bag concept

The safety bag system was designed by Klein (1991) in order to manage the routing of rolling stock though the shunt yards at a railway station. The system goal was to safely plan routes for the rolling stock, which shall be moved through a busy rail network. Since a section of the route may have other wagons on it, or switches are set such that other trains could enter the section, it is a hazardous task. The safety bag is an expert system that is realized as a dual channel design. The first system proposes different possible routes. In parallel, the second system assesses the routes with regard to hazards and vetoes or commits proposed routes. The safety bag is a rule-based expert system, whose rules embody the knowledge of the safety regulation of a railway station. The rules are realized via 'if...then...else' constructs.

2.5.3 Guardian agents

Fox and Das (2000) investigate intelligent systems and autonomous agents. Their main background is related to decision support systems in medical contexts. Since such decision support and planning systems are usually knowledge-based systems, the formal integrity of software and knowledge bases is not sufficient. Knowledge often is heuristic and, hence, knowledge-based systems are intrinsically faced with incompleteness and uncertainty (Das et al., 1997). *"However, even the best managed design and development programs will not guarantee reliability and safety of a complex system in complex settings, particularly when we need to cope with high levels of uncertainty and unpredictability of situations and events. To complement conventional development methods, intelligent agents should also be equipped with capabilities to monitor for hazards during their operation, and to apply their problem solving and reasoning functions to managing such hazards if, and when, they occur. Part of the agent's knowledge will be concerned with achieving its primary goals. Another part, which we can view as an independent agent, keeps a weather eye out for problems"* (Fox and Das, 2000), which actively manages safety.

Das et al. (1997) propose that safety constraints on the behavior of the system shall be modeled within this independent agent, using concepts of obligation and permission (deontic logic). It is suggested to introduce so-called *guardian agents*, *"whose job is just to watch out for hazards, and take charge if anything looks like going wrong"* (Fox and Das, 2000). Therefore, Fox and Das propose a concept that is intended to overcome the drawbacks of the safety bag concept of Klein (1991). In this connection, they criticize that the rationale behind rules in the safety bag concept is not explicitly represented; hence, it is impossible for the agent to recognize inappropriate rules. Furthermore, it is required to express how the logic shall be procedurally enacted, for instance, considering the timing of critical events. Finally, the proposed rules of Klein are application and domain specific, and they do not formalize general principles of safe and acceptable action. Fox and Das propose to provide the foundations for specifying safety protocols that might be used in any domain. *"There would be significant benefits if we could separate general safety knowledge from an agent's domain-specific knowledge. Formally, it would simplify the task of proving that an agent design is safe and sound. Practically, it would open up the possibility of constructing a standardized guardian agent that could*



Section 1
Detect any potentially hazardous anomaly and raise a goal to deal with it.

```
if      results of enquiry is State and                     (1)
        State is not safe
then    goal is remedy State
```

If the abnormal state is a known hazard with a known remedial action then propose it as a candidate solution to the goal.

```
if      goal is remedy State and                            (2)
        known remedy for State is Action
then    candidate for remedy of State is Action
```

Commit to action if the agent can establish that it is permitted according to the rules of the protocol.

```
if      candidate for remedy of State is Action and         (3)
        decision status of Action is permitted and Action is safe
then    decision status of Action is obligatory
```

Section 2
Any action that could be hazardous must be authorized [before it may be executed].

```
if      candidate for remedy of State is Action and         (4)
        possible(Action causes NewState) and
        NewState is not safe
then    authorization of Action is obligatory
```

When an action has been authorized it is permitted.

```
if      authorization of Action is obligatory and           (5)
        Action is authorized
then    Action is permitted
```

Any action that has no hazardous consequences is permitted.

```
if      candidate for remedy of State is Action and         (6)
        not(possible(Action causes NewState) and NewState is not safe)
then    Action is permitted
```

Figure 2.5: A generic safety protocol, according to Fox and Das (2000).

be a reusable component for many applications, from medicine to routing trains, from air traffic management to e-commerce, and from autopilots to robots" (Fox and Das, 2000).

Consequently, Fox and Das propose to formalize and generalize a basic safety protocol. In **Figure 2.5** a simple version is shown. Basically, it is organized into two sections whereby the first comprises rules that explicate how to assess and respond to irregular operation. The second section's policies describe when actions can be executed autonomously, and when authorization is required.

The generality of the protocol is based upon abstract features, which must be derived from specific situation data. Thus, states are the states of the environment, which, as well as actions, can be somehow classified as safe or not. Hazardous states have defined remedy actions, decisions can have states of permission or obligations.

They formalized these notations in a logic, L_{Safe}, for reasoning about safety. The logic contains modalities such as 'permitted', 'obligatory', 'safe', and 'authorized' which can be either applied to characterize an action, or the property of a state. Hence, permitted or obligatory actions leading to safe states can be allowed to be performed autonomously by an agent, obligatory but potentially hazardous actions must be authorized, or optional actions provoking hazardous states are not allowed to be executed at all. In the end, 'safe' is a (binary) predicate that describes that a state comprises no unacceptable risks and no risk mitigation action have to be performed. The predicate 'safe' has to be initially assigned to states and does not allow a gradual or probabilistic state characterization.

2.5.4 Dynamic risk assessment

Wardziński (2008) focuses on safely operating autonomous vehicles. Autonomous vehicles are autonomous robots whose task repertoire is dominated but not exhausted by the goal of traveling along a route. Autonomous mobile robots are however a wider class than the vehicle classes mentioned, whereas the boundaries are not rigid between these classes (Veres et al., 2011). Hence, the domain of autonomous vehicles provides a huge intersection with the domain of robots.

Autonomous vehicles are typically intended to operate in open environments. An open environment is understood as *"an environment in which agents operate and can have different, not consistent missions and strategies"* (Wardziński, 2008). Typically, actions may lead to unexpected outcomes in open environments, thus, plans fail and have to be dynamically revised (Ahle and Söffker, 2006).

Similarly to the guardian agent and self-risk evaluation approach, Wardziński suggests the need for a dynamic risk assessment approach. Wardziński points out that binary safety barriers may lead to significant problems. A safety barrier is *"an obstacle, an obstruction, or a hindrance that may [...] prevent an action from being carried out or an event from taking place"* (Hollnagel, 1999). But even sophisticated and complex safety barriers remain a binary view on safety: Barriers (safety functions) are activated in case of approaching unsafe operating conditions. This binary view on safety (namely the classification in safe and unsafe states) is a simplification of safety. The problem of the binary classification in complex environment conditions is that if two or more hazards have to be considered that appear in one situation, the binary perspective on safety does not support to deliberate any tradeoff of risks. For more complex situations, sets of safety functions may cause dead-locks. This is the case, for instance, if the system is locked in an unsafe situation because all action alternatives are classified as unsafe (Wardziński, 2008). Consequently, Wardziński suggests deriving risk level information and provide it to the planning process, where it has to be adequately considered. Wardziński defines the dynamic risk assessment approach as *"to design a system which is able to perceive and interpret risk factors and then assess how far it is on the scale starting from an absolutely safe state and ending with an accident. A system should be able to assess the risk of the situation before carrying out a specific action. In that way the system would be able to select safe actions and avoid actions leading to hazards"* (Wardziński, 2008).

2.5.5 Safety for the system situation awareness

In the work of Wardziński (2006), situation awareness is mentioned as an important requirement to address safety for autonomous vehicles, or more specifically to enable the dynamic risk assessment approach. A system that has situation awareness is often described as a system that knows *"what is going around"* (Wardziński, 2006).

Wardziński refers to the well-established definition of situation awareness of Endsley: Situation awareness is *"the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future"* (Endsley, 1995). The definition incorporates three levels of situation awareness, 1) the perception of information from the environment, 2) the comprehension of the perceived information, and 3) the projection of the information into the near future for the purpose of guiding actions. Situation awareness is a mental state or a state of knowledge. Processes to achieve and maintain situation awareness are referred to as situation assessment (Fischer et al., 2011). In this connection, Wardziński details three situation assessment functions for achieving safety incorporating situation awareness. These are 1) situation risk level assessment, 2) regulations compliance assessment, and 3) mission progress assessment. The situation risk level assessment function identifies risks in current and future situations in order to enable planning taking into account the risk levels of particular actions. There might be different strategies how risk information is handled. To provide an example, Wardziński formulates a strategy for selecting the safest action in VDM[4]-like notation:

```
vehicleAction( s : Situation ) as : ActionScenario
post
  as ∈ possibleScenarios( s )
  ∧
  riskAssessment( s, as ) =
    min{ ra | ra = riskAssessment( s, ax ) ∧ ax ∈ possibleScenarios( s ) }
```

Fu and Söffker (2011); Ahle and Söffker (2006) illustrate how automated supervision of a vehicle can be realized. The system learns to understand the control of a technical system via human-machine interaction at a highly abstracted level, for instance, for a lane change maneuver on the highway. Here, the automated supervision system checks the consistency of the driver's actions and the learned mental model of the driver's behavior with regard to the specified goal and specified rules (Fu and Söffker, 2011). This automated supervision approach allows a formalized strategy to analyze the logic of interaction and an application to technical systems. It bridges the gap between the pure signal level and the complex interpretation level by learning to structure (understand) the current situation. This is basically achieved by supervising interaction in terms of storing and refining experiences with the purpose to generate an internal model of the (assumed) structure of the interacting outside world. Here, a central aspect is that the currently available but changeable and refined mental model structures the system's perception and, thus, the 'situation understanding', also called situation awareness (Söffker, 2008). The concept is that the automated supervision system fully understands the shown task, it has to learn to control supervised task on its own and, hence, can be called autonomous, and consequently, used to realize robot control systems (Ahle and Söffker, 2006; Gamrad and Söffker, 2009a,c; Gamrad, 2011). The concept is described in more detail in Section 4.

[4] Vienna Development Method, see `http://www.vdmportal.org/twiki/bin/view` [accessed; online 17-December-2012]

2.5.6 Summary

There are several concepts available, each covering important aspects. However, it seems that the interplay and control of the entire robot behavior has to be focused. This control is finally grounded in a cognitive level. Thus, the integration and de facto realization of the available concepts within an (exemplary) cognitive architecture denote a suitable concept to approach the safety problems of intelligent systems.

The generic safety protocol proposed by Fox and Das (2000) denotes an abstract concept for taking into account safety concerns for automated decision making, with a potential to provide a basis for constructing a decision calculus or algorithm for autonomous robots. However, it tends to realize safety via pre-specifying behavior patterns. Moreover, safety is treated in a binary fashion (safe, not safe) and it is unclear how the classification of safe or not safe is realized.

The dynamic risk assessment approach proposed by Wardziński (2008) favors gradual risk classifications in order to explicitly overcome drawbacks of interfering complex safety rules. This can be the case, when typical rule-based systems take control, due to inconsistent rules or occurrence of multiple conflicting risks. According to Wardziński, a system itself has to continuously assess current and future situations with regard to risks. Therefore, Wardziński provides an abstract VDM notation of the concept. It is not gone into detailed how risk information is derived, how a foresight is realized, and how risk information is taken into account for decision making.

Similarly Seward et al. (2007) propose in the latter section a self-risk assessment approach, in which the system itself decides about the selection of subsequent actions on basis of numeric risk expressions. Furthermore, it is illustrated how a static fault tree could be applied to model the relationship between the system state variables and the probability of a specified accident. Utilization of fault trees for formalizing hazard risks appears to be useful method, however, their risk assessment is integrated into specific (movement) skills; thus, it seems to lack integration at a (centralized) cognitive level, more closely related to abstract world knowledge, which, for instance, requires a conditional part for checking the general presence of a hazard in respective situation.

However, there is decision theory, which provides the basis for decision making, as well for multifactorial decision problems. Consequently, it appears that several approaches have to be combined to a more general framework that subsumes the variety of already proposed approaches, and new requirements, which result from a renewed analysis of the significant safety problems of current or prospective service robots.

Autonomous operation is assumed to be one of the key features of this kind of systems due to several reasons. This feature, on the other hand, is probably the one that distinguishes this kind of systems from conventional. Hence, it is important to understand what autonomy is, and which implications are imposed as a result.

3 Robot Autonomy

3.1 Autonomous Systems

Autonomy is discussed in different scientific fields. Etymologically, autonomy is a combined term of the words 'autos' (self) and 'nomos' (rule or law). Originally, it reflects the idea of the self-governing of the Greek city-state (Dworkin, 1976). But there are various philosophic thoughts about autonomy also with regard natural agents such as humans, animals, and other biological organisms, e.g., see Varela (1979); Smithers (1997); Collier (2002); Verhagen (2004). On the other hand, there are thoughts and definitions about autonomy for artificial agent systems such as in artificial live, multi agent systems, and agents in natural environments, e.g., see Smithers (1997); Collier (2002); Verhagen (2004); Franklin and Graesser (1997); Beavers and Hexmoor (2004); Castelfranchi and Falcone (2003); Carabelea et al. (2004). Basically, autonomy is neither a kind of abstract system concept, nor it is a function, state or mechanism; it is an organizational property of a system (Di Paolo and Iizuka, 2008). From a very general perspective, autonomy can be understood as a synonym to 'independence' or 'self-sufficiency' (Castelfranchi, 1995). The generation of universal moral laws by the agent itself is autonomy (Kant and Kirchmann, 1869), and heteronomy is the opposite, when laws are imposed from outsides, from a higher authority. An agent has freedom, and is consequently autonomous, when there are no external barriers to its actions (Verhagen, 2004). Hence, freedom of the will and autonomy are closely interrelated in philosophy, at least for natural agents.

A special feature of autonomous artificial systems is that there is a direct access to the 'mind' (Verhagen, 2004). This allows having an insight into the ongoing processes and offers the chance to intervene, or in other words to affect the autonomy. With regard to robotics, the new revision of the DIN EN ISO 8373 shall soon define that *"autonomy of a robot is the ability to perform the intended task based on the current state and sensor information without human intervention."*[1]

3.2 Aspects of Autonomy

Autonomy has a broad meaning and its relativity allows referring it to several characteristics in varying degrees which are in particular, material, psychological, informational,

[1] according to Gurvinder Virk's (Chairman, ISO TC 184/ SC 2/WG 7 on Safety of personal care robots) recent definitions with regard to revision of the DIN EN ISO 8373 (2010), oral presentation held 3 July 2012 at the International Workshop on Medical Robots in Milano, Italy, and 12 October 2012 at the international conference on Intelligent Robots and Systems (IROS) in Vilamoura, Portugal

representational, and anticipatory autonomy (Collier, 1999), mental autonomy (Collier, 2002), autonomy from stimuli, cognitive autonomy (Verhagen, 2004; Castelfranchi, 1995), norm autonomy (Verhagen, 2004; Carabelea et al., 2004), executive and goal autonomy (Castelfranchi and Falcone, 2003; Castelfranchi, 1995), user, environment, and (agent) self-autonomy (Carabelea et al., 2004), intention and organization autonomy (Carabelea et al., 2004), and finally, social autonomy (Castelfranchi, 1995; Castelfranchi and Falcone, 2003; Collier, 1999). The mentioned aspects and related definitions are listed in **Figure 3.2.** In determining autonomy, most authors explicitly agree that it must be considered to be relative to something external of the system (Verhagen, 2004; Carabelea et al., 2004; Castelfranchi and Falcone, 2003; Collier, 1999). Collier (1999) argues that the different kinds of autonomy arise at different levels and hierarchies and, hence, autonomy is relative to level and hierarchy as well. Collier mentions that something can be non-autonomous at the most fundamental physical level, under the extreme conditions found in physics. But it might be autonomous biologically in the less intense environments of organisms. For instance, minds can be autonomous with regard to the information content, but they depend on their biological embodiment. Hence, autonomy is relative, it is a matter of degree, and it has hierarchical aspects.

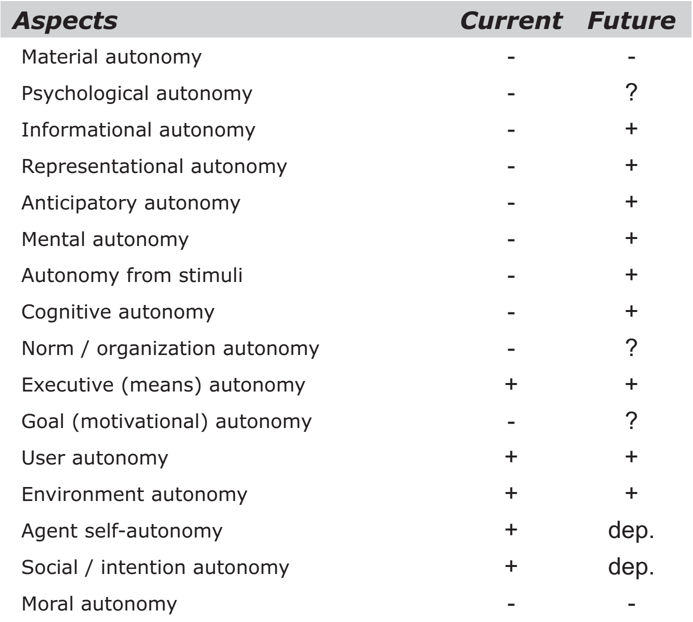
3.3 Present, Desired, and Future Robot Autonomy

Some of the current cleaning robots[2] cover some of the aforementioned aspects, for instance, those that realize the self localizing and mapping (SLAM) approach. They process information derived from sensor data, such as distance measures; they internally generate and maintain a map representation to perform optimized movement trajectories. Due to the fact that the map is used and generated simultaneously, it is questionable if the robots anticipate their selected actions, however, the behaviors are adapted to the current situation: According to the current situation, a strategy or partial plan is selected, on the one hand, in order to optimize the cleaning process, on the other hand, to pursue different intentions, for instance, surface or spot cleaning, recharging the battery and so forth. Hence, several aspects are already realized in current robots. According to this example, in **Figure 3.1** is shown which autonomy aspects for commercial robots can be considered to be achieved. The cleaning robot acts (more or less) without user intervention, just as it is advised to do so, while most of its actions are selected based on its internal map and not only on external stimuli, without social interactions or agent-agent cooperation.

For future robots, more complex tasks should be performed. This implies a cascade of needs for other aspects of autonomy. It is basically well known that the programming of complex tasks requires for enormous programming efforts, which may be a motivation to apply AI-methods to such problems (Collier, 1999). Fox and Das summarize in this regard:[3] *"In some fields it may be possible to anticipate most hazards, but in medicine and similar complex settings this seems to be out of the question. The scope for unforeseen*

[2] see comparison of a selection of robots in `http://www.testberichte.de/d/read-swf/307638.html` [accessed; online 5-November-2012]

[3] Fox and Das have a special scope on medical applications, whose characteristics are assumed to be comparable with other complex environments



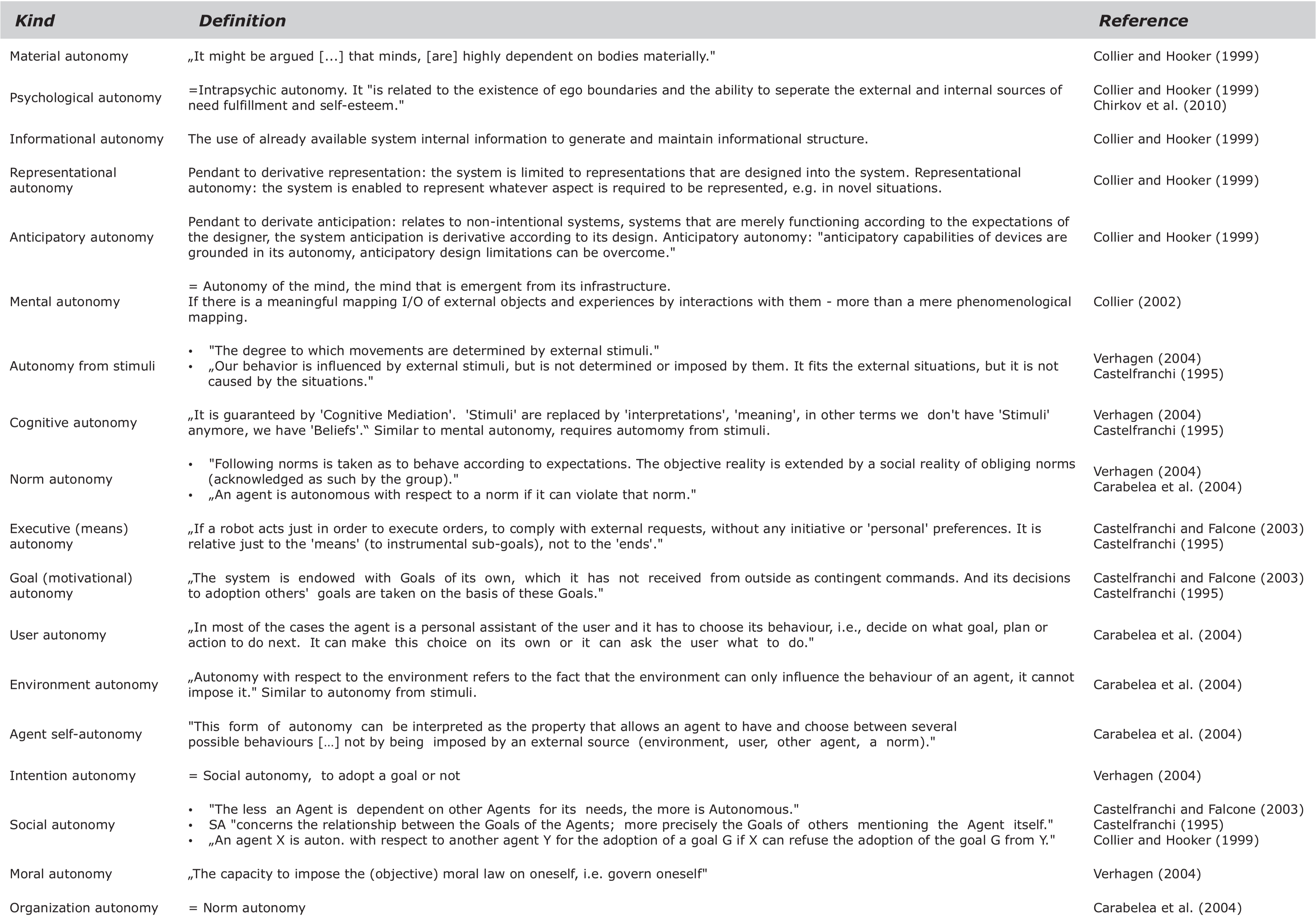
Aspects	*Current*	*Future*
Material autonomy	-	-
Psychological autonomy	-	?
Informational autonomy	-	+
Representational autonomy	-	+
Anticipatory autonomy	-	+
Mental autonomy	-	+
Autonomy from stimuli	-	+
Cognitive autonomy	-	+
Norm / organization autonomy	-	?
Executive (means) autonomy	+	+
Goal (motivational) autonomy	-	?
User autonomy	+	+
Environment autonomy	+	+
Agent self-autonomy	+	dep.
Social / intention autonomy	+	dep.
Moral autonomy	-	-

Figure 3.1: Assumed present and future autonomy aspects of robots ('+' applies, '-' applies not, '?' questionable, 'dep.' depends).

and unforeseeable interactions is vast. It is simply not possible to guarantee that all possible hazards will be exhaustively identified for substantial applications. Hazards that have not been anticipated will arise and strategies will be needed to prevent and minimize their consequences. Software designers, including those building AI systems, must acknowledge this hard reality" (Fox and Das, 2000). The rationale behind the generation of intelligent systems is to overcome such design limitations of systems in unpredictable environments. This increases the independence or autonomy from the initial design and, hence, from the designers, because otherwise, the system must be updated by its designers with incremental updates or the like, in each case a new and unknown situation is detected. Hence, a superimposed objective of many AI-approaches in robotics is the intention to evoke emergence effects, so that adaption capabilities can enhance from the initial system design in new ways. The statement of Russell and Norvig (2010) underpins this: *"An agent lacks of autonomy when it relies on the prior knowledge of its designers. An autonomous agent should compensate for partial or incorrect knowledge"* (Russell and Norvig, 2010).

3.4 Autonomy for Structural Drift

This understanding of autonomy may subsume several already available notions of autonomy, however, it emphasizes the changing of the system 'post-design'. Hence, this can be called *autonomy for structural development.* The autonomy for structural development is based on the *structural determinism and coupling* of systems, according to Maturana and Varela (1998). Structural determinism means that any system always behaves according to its current structure. If the system differently behaves than it is



Kind	Definition	Reference
Material autonomy	„It might be argued [...] that minds, [are] highly dependent on bodies materially."	Collier and Hooker (1999)
Psychological autonomy	=Intrapsychic autonomy. It "is related to the existence of ego boundaries and the ability to seperate the external and internal sources of need fulfillment and self-esteem."	Collier and Hooker (1999) Chirkov et al. (2010)
Informational autonomy	The use of already available system internal information to generate and maintain informational structure.	Collier and Hooker (1999)
Representational autonomy	Pendant to derivative representation: the system is limited to representations that are designed into the system. Representational autonomy: the system is enabled to represent whatever aspect is required to be represented, e.g. in novel situations.	Collier and Hooker (1999)
Anticipatory autonomy	Pendant to derivate anticipation: relates to non-intentional systems, systems that are merely functioning according to the expectations of the designer, the system anticipation is derivative according to its design. Anticipatory autonomy: "anticipatory capabilities of devices are grounded in its autonomy, anticipatory design limitations can be overcome."	Collier and Hooker (1999)
Mental autonomy	= Autonomy of the mind, the mind that is emergent from its infrastructure. If there is a meaningful mapping I/O of external objects and experiences by interactions with them - more than a mere phenomenological mapping.	Collier (2002)
Autonomy from stimuli	• "The degree to which movements are determined by external stimuli." • „Our behavior is influenced by external stimuli, but is not determined or imposed by them. It fits the external situations, but it is not caused by the situations."	Verhagen (2004) Castelfranchi (1995)
Cognitive autonomy	„It is guaranteed by 'Cognitive Mediation'. 'Stimuli' are replaced by 'interpretations', 'meaning', in other terms we don't have 'Stimuli' anymore, we have 'Beliefs'.“ Similar to mental autonomy, requires automomy from stimuli.	Verhagen (2004) Castelfranchi (1995)
Norm autonomy	• "Following norms is taken as to behave according to expectations. The objective reality is extended by a social reality of obliging norms (acknowledged as such by the group)." • „An agent is autonomous with respect to a norm if it can violate that norm."	Verhagen (2004) Carabelea et al. (2004)
Executive (means) autonomy	„If a robot acts just in order to execute orders, to comply with external requests, without any initiative or 'personal' preferences. It is relative just to the 'means' (to instrumental sub-goals), not to the 'ends'."	Castelfranchi and Falcone (2003) Castelfranchi (1995)
Goal (motivational) autonomy	„The system is endowed with Goals of its own, which it has not received from outside as contingent commands. And its decisions to adoption others' goals are taken on the basis of these Goals."	Castelfranchi and Falcone (2003) Castelfranchi (1995)
User autonomy	„In most of the cases the agent is a personal assistant of the user and it has to choose its behaviour, i.e., decide on what goal, plan or action to do next. It can make this choice on its own or it can ask the user what to do."	Carabelea et al. (2004)
Environment autonomy	„Autonomy with respect to the environment refers to the fact that the environment can only influence the behaviour of an agent, it cannot impose it." Similar to autonomy from stimuli.	Carabelea et al. (2004)
Agent self-autonomy	"This form of autonomy can be interpreted as the property that allows an agent to have and choose between several possible behaviours […] not by being imposed by an external source (environment, user, other agent, a norm)."	Carabelea et al. (2004)
Intention autonomy	= Social autonomy, to adopt a goal or not	Verhagen (2004)
Social autonomy	• "The less an Agent is dependent on other Agents for its needs, the more is Autonomous." • SA "concerns the relationship between the Goals of the Agents; more precisely the Goals of others mentioning the Agent itself." • „An agent X is auton. with respect to another agent Y for the adoption of a goal G if X can refuse the adoption of the goal G from Y."	Castelfranchi and Falcone (2003) Castelfranchi (1995) Collier and Hooker (1999)
Moral autonomy	„The capacity to impose the (objective) moral law on oneself, i.e. govern oneself"	Verhagen (2004)
Organization autonomy	= Norm autonomy	Carabelea et al. (2004)

Figure 3.2: Different kinds of autonomy aspects

constructed, there is a failure. For instance, adopting Maturana and Varela's example, if a car does not accelerate by pressing the gas pedal - as it is constructed - there is any built-in reason to ignore the user input, or it has a failure. The failure itself is a change of the system structure, and the system behaves according to its changed structure.

Furthermore, a system and its environment are structurally coupled, which means that system and environment mutually perturb each other. The history of changes that a system undergoes while being structurally coupled with its environment is called *system ontogeny*. It can be said that all physical systems undergo structural changes. There exist various types of changes, and there are at least such changes as aging, abrasion etc. The structural determined system (every system) functions according its 'mutating' structure; in the case of aging, or abrasion, it typically becomes less reliable. The ongoing ontogenic change of a system is called 'structural drift' (Maturana and Varela, 1998). The environment as well underlies a structural drift. The structural drifts of both, system and environment, can be considered to be driven by the ongoing perturbing interaction of the system and environment in two different directions:

- On the one hand, the system adapts to external requirements, for instance, to the present environment condition in order to be able to deliver its current function or task.
- On the other hand, a system like a physical robot that is constructed for providing certain functionality or task should perturb the environment in a respective (intentional) manner.

For the system that is well adapted to its environment, one can state that the structural drift of a system is always in congruency with the structural drift of its environment (Maturana and Varela, 1998). As they express it, the environment functions as a 'selector' of the structural drift the system has to undergo, and the system actions are the 'selectors' of the changes that the environment is subjected to. From this it becomes obvious that limitations of the changeability of the system structure depict the limitations of its adaptability (and vice versa[4]). It can be assumed that a robotic system remains mainly structural constant with regard to its hardware (embodiment). Consequently, its autonomy is based on structural changeability within its software realization. For instance, if a robot can not reach a certain area with its manipulator, it, its adaptability, and, hence, its autonomy is limited to this hardware structure limitation. With regard to the crow example, intelligence can be applied to overcome such limitations. Here, crows utilize tools in the correct sequence in order to fetch some food they usually can not reach (Wimpenny et al., 2009). Thus, intelligence can increase autonomy.

3.4.1 Autonomy as control of control

Smithers (1997) differentiates artificial systems into three basic types of systems: 'Automatic', 'controlled', and 'autonomous systems'. *Automatic systems* are 'self-moving' or 'self-acting' systems. They are built for very specific purposes, and they are just doing

[4] the limitations of the changeability of the environment structure limits what the system can realize within the environment

that what they are built for. For instance, a clockwork merely moves the clock hands. It is important to denote, that the movement has to be generated by transformation of energy.[5] The transformation of information, typical for software programs, can be automatically realized as well. Thus, many software programs (considered to be systems) are often automatic systems.

Controlled, 'cybernetic' or 'self-regulated' systems are basically also kinds of automatic systems. The essential difference is that control systems comprise a feedback loop. With the help of this feedback loop, control systems are basically enabled to compensate deviations due to disturbances by comparing its real with the desired outcome. This adaption (to the present operating or environment conditions) is realized according to predefined control laws or rules (ex-ante), which are usually derived from a model of the system to be controlled. Rasmussen (1983) denotes that humans usually do not control their actions on basis of feedback signals from their environment. Such feedback control is required for very specific, slow, and accurate movements, for example, assembly tasks or drawing. In most cases action control is feedforward. Consequently, humans can establish various feedback loop behaviors if they are required. However, most often they are not required and, hence, typically human behavior is finally not only based on traditional control theory.

In general, controlled and autonomous systems differ from automatic systems by being equipped with any kind of feedback loop. Thus, both are basically assigned to the class of self-regulating systems, whereas the class of autonomous systems is a superset of self-regulating systems. Smithers states: *"A definition of autonomy is not to be derived from control theory since it is a concept that subsumes the concepts of control and self-regulation, not one that can be subsumed by them"*[6] (Smithers, 1997).

3.4.2 Autonomy as adaptability

Collier and Hooker (1999), favoring a graduation of autonomy, refer to control and self-regulation as *first* and *higher order adaptive strategies*. In particular, a first order adaptive strategy is based on fixed sets of system-internal and system-environment interactions which are determined by an already known set of environment conditions. Hence, this fits for traditional engineered systems with a static implemented functionality. Autonomous systems possess higher order adaptive strategies which are strategies for modifying first order strategies in order to increase adaptive 'width, strength or consistency'. This is called the *process of adaptation*, which is a *"system-environment open-loop interaction that yields system modification such that [...] system autonomy and internal system information increases"* (Collier and Hooker, 1999). Second or higher order adaptive strategies denote such capability to adapt adaptations[7] and are called *adaptability* (Collier and Hooker, 1999). Further on, systems with adaptability obligatorily are, according to Collier and Hooker (1999), information processing systems

[5] A manual gear box for instance, is a system that merely transforms moments and thus, no automatic system.

[6] in terms of control of control

[7] There is an analogy to Smithers' statement 'control of control' with regard to its similarity to 'adapting the adaption'.

(Conrad, 1993), and the adaptability is closely related to the system intelligence (Hooker, 1995). The adaptability is enhanced by *"building features into the regulatory design of the system corresponding to as wide-ranging dynamical patterns shared by the class of tolerable environments as possible and using these as the basis for anticipative adaptations in response to local, short-term perceptual information"* (Collier and Hooker, 1999). On the one hand, the common properties of the environment conditions affect the control system to change. This could concern changes that require more time to change, for instance, such as the modification, extension, generalization of complex movements in skills, or the like. The knowledge about the common properties of the environment may be utilized in an anticipation process in order to be able to flexibly adapt planning according to expected effects and changes of the environment.

3.4.3 Autonomy for anticipation

Considering limitations within the software structure, Collier (1999) criticizes derivative anticipation capabilities. Derivative anticipation capabilities are anticipation capabilities that are explicitly designed into the system. Collier states that they are overestimated in providing autonomy, because they underlie design limitations. The anticipation of an autonomous system should be autonomous and intentional. If system functionality is specified by its designer, it is derivative functional and, hence, limited to its functional design. Such a system requires for reference to any external autonomous functional system because its intention is derived from the external autonomous intentional system. However, *"if the anticipatory capacities of a device are grounded in its autonomy [...], anticipatory design limitations can be overcome through reconsideration of their contribution to autonomy, permitting some new functions to arise which contribute to autonomy in fundamentally new ways"* (Collier, 1999). In conclusion, the so-called *anticipatory autonomy* requires the realization of other autonomy aspects: It requires representation autonomy and informational autonomy. This might be illustrated by the following allegory: A system can not consider what it can not anticipate, it can not anticipate what it can not represent, and it can not represent those aspects that can not be expressed via its informational structure.

3.4.4 Autonomy as reciprocal information entropy

According to the work of Bertschinger et al. (2008), autonomy can be considered to be inversely proportional to the system information entropy. In this regard, Bertschinger et al. present an information theoretic approach to autonomy, and provide a quantified autonomy measure. They argue that the autonomous system should not be fully determined by its environment. This decoupling of system and environment, they call *non-trivial informational closure.* The informational closure can be increased if the mutual information between system and environment is enhanced. The informational closure measures *"the extent to which the system models its environment"* (Bertschinger et al., 2008). It appears logical that if the internal modeled representation of the environment is used to predict the outcome of interactions, or the dynamics of the environment, the information entropy is decreased, because there is nothing new to the system.

3.4.5 Informed autonomy

Rohde and Stewart (2008) reflect on different definitions of autonomy, and in particular, they criticize that there are many ascriptions of autonomy, and many ascriptions are based on observed behaviors of the system. On the one hand, observation is subjective, and can be only objectified considering agreed conventions in a constructive world view. Maturana and Varela (1998) give the example with the submarine pilot, who controls the submarine based on different kinds of meters in order to pass a reef full of obstacles. The pilot never left the submarine, thus, he/she possibly never knows what obstacles, reefs and submarines are, in contrast to the observer that is located at the seacoast. Hence, the observer and pilot probably will have some troubles to give mutual understandable specifications of the navigation task. It may be concluded from this that robot and humans may possess over a compatible interpretation and understanding of the world, if the autonomous robot is equipped with similar sensors, and it can access to the social agreed conventions, for instance, via social interactions. The urgent necessity for social robotics is discussed later on in Section 3.5.2.

On the other hand, Rohde and Stewart's criticism on relating autonomy to observed behaviors of the system is that these observations underlie an intrinsic behaviorist reduction. This can be illustrated by considering the input/output mapping of even small black boxes may lead to serious problems, as can be shown by the theory of Finite State Machines (Gill, 1962) with the help of the non-trivial automaton example (cf. von Goldammer and Paul, 1995). Assuming informational closed systems as systems that are capable to internally represent external aspects, and assuming that such systems, or intentional systems are self-referential systems, this indicates that those systems should have a strong coupling to their system-internal representations. These may be modeled as black-box system which carries more internal information (states and their alphabet) as can be transferred once per input or output (and their alphabets). For instance, the black-box system has binary input and output ($p = 2; q = 2$) and 10 internal states ($n = 10$).[8] With knowing only this, the theory of Finite State Machines allows to compute that there are $N = (qn)^{pn} \approx 1,05 \cdot 10^{26}$ possible Finite State Machines, and it requires an experiment of presenting $l \leq (2n-1)(Nn-1) \lesssim 2 \cdot 10^{28}$ different inputs to the system until the mapping functions from input to internal states and output can be determined. The number of possible Finite State Machines exponentially increases, extending the number of inputs, states, or their alphabet. Consequently, behaviorist reduction, e.g. Touring-tests, seems to quickly lead to analysis problems and does not support a formal determination of autonomy (Rohde and Stewart, 2008). They *"lose the hope for an absolute criterion, providing timeless necessary and sufficient conditions for [genuine] autonomy"* (Rohde and Stewart, 2008). Instead, they outline the idea of 'informed autonomy', which is basically to focus on the mechanisms that generate autonomy.

[8] the human brain has an estimated ratio of input/inter/output neurons of approximately $1 : 100.000 : 10$ (Maturana and Varela, 1998, p. 159)

3.4.6 Research-related mechanisms generating autonomy

Lussier et al. (2004) lists a couple of AI methods that are usually utilized to realize autonomous features; these are

- planning,
- execution control,
- situation recognition (including diagnosis), and
- learning.

The execution control is the coordination and supervision of the execution of plans. High-level actions are often decomposed into sequences of behaviors or simpler tasks. The execution is supervised in order to react to possible failures, occurring due to the system failures or due to unexpected environment conditions. The diagnosis is required to identify an erroneous system state, generally after error detection. Diagnosis may be regarded as a specific case of situation recognition (Lussier et al., 2004).

In consequence, mechanisms based on methods listed above (without intention to be exhaustive) play a central role in generating autonomous robots. However, learning seems to be a key aspect, because systems without the capability to learn may adapt to the foreseen and considered conditions, and if robustly realized also to some beyond. A 'real' autonomous system, a system that is capable to learn, however, is intended to operate under conditions where occurrence of something new is no exception but normal (Smithers, 1997).

3.4.7 Autonomy via nature and nurture

Autonomous systems have, *"as an intrinsic property, the ability to deal with new situations: as the conditions change for this ongoing process of formation,[9] so the laws of regulation so formed change, and change in such a way to keep the autonomous system naturally well fitted to its current situation"* (Smithers, 1997). Here, the 'fitting to the current situation' denotes a relation to the demands for the intelligence of the autonomous system with regard to the 'nature versus nurture debate'[10]. This debate centers on what contributes to the human development. In this connection, some philosophers argued that certain things are inborn and occur regardless of environment influences and, thus, are genetically inherited (nature). Others argued that the mind starts at a blank state, also known as 'tabula rasa', and everything that humans are is coined from their experiences (nurture). Ridley (2003) favors the perspective that there is a mutual relation between environment and genetic influences in general, which may vary depending on what is considered. For some more complex behaviors, such as social behaviors, individuals are rarely affected by their ancestors as rather by the

[9] continuously forming the laws of operation

[10] An analogy to robotics could be described as follows: The nature state is the state of the robot as it is delivered to the user or intended place of operation. The nurture denotes the development of the robot in accordance to its operation environment, beginning from the natural state at its start of operation.

pressure of the environment: *"Distant relatives can have very similar social systems by convergent evolution if they inhabit similar ecological niches"* (Ridley, 2003). Thus, the role of the environment, the ecological niche of an autonomous system, seems to play an essential role: On the one hand, it may have effect on the suitable morphology of the system (cf. Collins et al., 2005). On the other hand, it affects the requirements for the behavior repertoire. Since a behavior repertoire is huge, it potentially allows to maintaining the system's autonomy under a variety of environment conditions. If the behavior repertoire is limited, the chances to maintain the system's autonomy is limited as well with regard to changed environment conditions, as expressed by Arkin: The *"concept of niche is important to roboticists because of their goals. If the roboticist intends to build a system that is autonomous and can success fully compete with other environmental inhabitants, that system must find a stable niche or it (as an application) will be unsuccessful"* (Arkin, 1998).

In consequence, the learning from experiences (e.g. new behaviors or application of available behaviors for new purposes) denotes a key element to allow a system to overcome pre-specified behaviors in order to maintain autonomy for not considered environment conditions. For humans, and this equivalently applies for robots that are sharing parts of the humans ecological niche, complex behaviors seem to be required. These are far more complex than simple inherited stimulus-response behaviors that suit well, for instance, for bacteria to maintaining their autonomy. Hence, the adaption is required to enhance autonomy, and its complexity is certainly related to requirements of the ecological niche, in which the system is intended to exist and operate. Consequently, autonomy can be regarded to depend on

- the requirements from its ecological niche,
- the behaviors, skills, or strategies that are predefined (nature) to maintain autonomy under usual conditions, and
- the skill to integrate (learn from) experiences (nurture), in order to extent the repertoire to maintain its autonomy under new conditions.

As Ridley denotes, *"human beings were under selective pressure to develop more processing power"* (Ridley, 2003) with a mind that is not equipped with innate data but innate ways of processing data (Pinker, 1994). Thus, maintaining autonomy in the human ecological niche seems to require the ability of complex information processing, and the establishing of processing structures such as human cognitive functions. In consequence, researchers developing autonomous robots do somehow concentrate on those system features that represent key aspects of systems as they are typically also perceived in cognitive science.

3.4.8 Summarizing annotations

If a robot is demanded to perform complex tasks, and this requires a deeper understanding of situations, it is very probable that on the system's understanding may not result evidence about the present situation, as rather the system beliefs what it looks like, and when it is detected (Wardziński, 2006). In consequence the system becomes cognitive autonomous, and autonomous from stimuli, because the stimuli are fused and integrated to interpretations, the system does not sense its environment, it rather interprets the input data based on its current state of knowledge and intention (Söffker, 2008), which in turn is based on its ontogeny, and its goals.

The nature of the changes within the software structure can be considered as a matter of change of internal states, knowledge, connections of software components and the like. This can be compared to the *psychogenesis* of humans. The psychogenesis is the origin and development of mental functions, traits, or states; the development from mental as distinguished from physical origins.[11] The psychogenesis of humans is typically absolutely unique for each individual, because the complex environment, e.g. the human ecological niche, comprises inexhaustible number of facets. It appears reasonable to assume that the ontogeny of intelligent artificial systems, intended to mainly share the human niche, results in the synthesis of unique individual systems.

In order to summarize appearing most essential requirements for realizing an autonomous robot, it should correspond to be

- an information processing system,
- a system that can generate and maintain representations of relevant aspects,
- a system that possesses the capability to intentionally anticipate,
- a system that has the freedom to alter its rules of control and behavior, respectively,
- a system whose complexity is related to the complexity of its respective ecological niche; therefore, it may be required to be
- an intelligent system.

3.5 Implications for Safety

It is questionable, if the next generation of robots requires being goal autonomous, since a robot that pursues its own goals imply several critical aspects, as outlined later on in Section 3.5.2. Similarly, moral autonomy, the freedom to construct and self-impose its own moral laws, implies suspect problems. Weng et al. (2009) differentiate between *Type 1* and *Type 2 artificial ethics*. The first refers to robots that are programmed to obey a given set of legal and ethical norms, and the latter, to robots that are able to generate their own values and ethics.

Many contributions to safety rely on applying Type 1-like limitations, which means that safety limitations are well engineered into systems and, hence, are externally given to the system. With this in mind, the traditional safety engineering practice could be classified as *Safety Type 1* limitation. If this affects the behavior of a system in any way, the system acts at least partially heteronomous; hence, this effects or reduces its autonomy. This

[11] Merriam Webster dictionary, `http://www.merriam-webster.com/dictionary/psychogenesis` [accessed; online 20-November-2012]

appears reasonable, because any safety-critical system should optimally be heteronomous in hazardous situations, and behave according to the specified protective mechanisms.

The minority of contributions are related to safety limitations in the meaning of Type 2. As already mentioned, the lack of complete safety specifications possibly imply that protective functions have to be learned, or that Type 1 limitations have to be extended or refined. Without a doubt, such capabilities must be carefully investigated. In that respect, it can be argued according to Yampolskiy (2013) that safety mechanisms always have to remain as given by humans (Type 1), and should not be a subject of the recursive self-improvement process of Artificial General Intelligence. Otherwise, it will be very likely that humans as well as their vital resources will not be reliably protected, and *"there will be a direct competition between superintelligent machines and people"* (Yampolskiy, 2013), what basically conflicts with the imperative of Jonas (1985), mentioned initially.

3.5.1 A cognitive perspective

As aforementioned, it may become very difficult to evaluate the function of a system by observing only its input and outputs. This disqualifies the classic concept of an additional safety observer, which operates in parallel to a system and limits its outputs to safe ones. However, there is a model which comprises a kind of observer, which certainly not lacks a behavioristic reduction, as it rather explains behavior between the poles of distinct forces - psychological forces within a psychodynamic framework. Psychodynamics and related psychoanalytical theories are of interest for cognitive science, because psychoanalytical concepts coincide with prevailing cognitive models of science (Bornstein, 2003). Psychodynamics basically considers changes a psychical system undergoes as determinants for its current behavior. One of the core assumptions of psychodynamics is the so-called *psychic causality.* Psychic causality signifies that cognitions, emotional responses, and expressed behaviors always stem from some combination of identifiable biological and/or, psychological processes (Rychlak, 1990). Biological and/or, psychological processes are the determinants for the behavior. Analogously, the inner states and processes of an artificial embodied system are the determinants for the system behavior.

Freud's *structural model of the [human's] psyche* is an empirical model whose scientific grounding is controversial discussed. The structural model is one of the Freudian concepts of the psychodynamics of humans, and it still plays an important role in the psychoanalytical approach. Kandel states that *"psychoanalysis still represents the most coherent and intellectually satisfying view of the mind"* (Kandel, 1999), and Buller that the unscientific character of psychodynamics *"faces a fast track to oblivion"* (Buller, 2005). Neither the combination of AI and psychoanalysis is new (Turkle, 1988), nor the introduction of psychodynamic aspects into artificial systems (Buller, 2005; Zeilinger et al., 2008).

In the scope of the safety-autonomy dilemma, it may serve as a rough sketch, illustrating how safety mechanisms may be integrated into an autonomous system without limiting it to stringent, ontogenically static rule-based safety mechanisms. The drawback of static safety mechanisms can be illustrated with two examples:

1) There are many accidents and suicides by people fall to death. In order to prevent this, a regulation could be issued that requires to install a safety barrier at every location, somebody could fall to death, for instance, in front of every window that is higher than five meters. This measure could be compared to a stringent static safety measure, which certainly would prevent many fatalities. On the other hand, what is about the utilization of windows, balconies, roofs, etc. as evacuation routes? It might be possible that the total amount of fatalities becomes higher than it was before. People usually do not jump out of the window, even if they have many possibilities to do so, because they are aware of the consequences. In case of fire, for instance, everybody is basically capable to deliberate the risk of jumping out of the window or not.

2) The classical method to ensure safety is to transfer the safety-critical system into a safe state if something unexpected happens. Robots are usually transferred into a stop state. Furthermore, states or actions that are specified to comprise unacceptable risks are banned from the robot's action repertoire. These measures represent static safety measures as well. If a robot, for instance, stops in a narrow passage because collision risks are verified to be too high, it certainly might reduce risks to a minimum on the one hand, but on the other hand, this might block the evacuation route of a department store and, hence, provokes high risk. The collision risk has to be regarded relative to the current environment condition. In this case, a deliberation of risks may allow the robot to accept a lower collision risk close to an evacuation route in order to avoid the risk of fire victims.

Wardziński (2008) already mentioned the problem of considering safety in a binary fashion, which is to regard the system to be either in a safe state or not (see Section 2.5.5). Beyond that, new situations and a changed system (with extended capabilities) have to be considered as well. It is questionable, if predefined rule sets (understood as state-actions pairs: 'if...then...else') are reasonable because the response with respect to unknown situation can not be well considered. Instead, it is supposed to inform the decision making process about risks of several courses of action. The system has to deliberate the action alternatives case-by-case, taking into account the inherent trade-off between task success, efficiency, and safety. The action repertoire of the robot can be additionally equipped with prespecified actions that cause the transition into a safe state (full/partial de-energization of the system and the like). Hence, these actions remain available as an acting alternative to be chosen as 'ulitma ratio' by the autonomous system, because if they are appropriate, the risk performing these actions should be very low. The essential difference is that the system is allowed to check if the pre-programmed 'limp-home' action is suitable in its current situation, or if there are more reasonable actions. This implies to draw the attention of safety investigations toward safe deliberation, and its related mechanisms for which intrinsic tensions exist between objectives of performance, efficiency, or optimality, and objectives of ethics, morality, or safety.

3.5.2 The psychodynamic structure model

As it is expressed by Buller [2002], Freud's psychodynamic theory expresses that the mental life is a kind of a continuous battle between conflicting psychological forces, which are, for instance, wishes, fears, and intentions. Freud's structure model explains this intrapsychic dynamics by the three interacting mental structures, called *id*, *ego*,

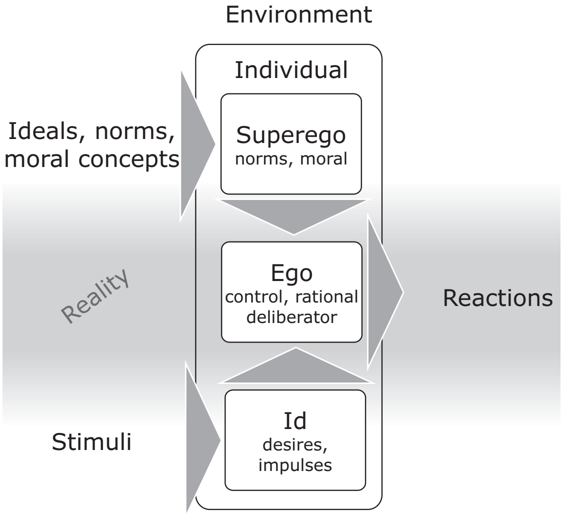


Figure 3.3: Freud's structure model of the human psyche

and *superego* (Bornstein, 2003). The *id* represents the basic drives, things wished to be instantaneously realized. Drives can be considered for artificial systems to be value state-variables that provide estimates of need (Albus, 1991). The *superego* is the antagonist of the *id* as an agency subsuming parental, educational, or social experiences, representing the moral agency. The *ego* has to mediate between *id* and *superego*, and synchronizes them with the reality, as illustrated in **Figure 3.3.**

The *superego* represents the component of interest, despite the discussion of its formation and its effects in psychopathological respects. In essence, it has an effecting and limiting function on the *id* drives and the reality of the *ego*. Interestingly, the *superego* seems to be developed during the process where the human evolves from the state, characterized by parental dependency and heteronomy to the state, in which decisions are of a self-reliant character. It re-echoes the moral laws that were learned so far by interacting with parents or others (Bornstein, 2003). Moral rules are standards of behavior, or principles of right and wrong.[12] In consequence, the safe behavior of a human as an autonomous self-responsible system has to be considered as matter of moral and, hence, based on moral laws of the *superego*. For instance, it is wrong (according to our social conventions) to endanger or injury somebody or damage somebody's property, if this is not required to avert a higher damage, for instance, similarly to robot moral behavior according to the rules of Asimov (Asimov, 1950).

The transfer of the structural model to robots may be controversial, but some aspects and functions of the *superego* are important. At first, the development of the *superego* is initially affected by a small group of persons (parents, family) in order to generate a first functional version of the *superego* with an externally given rule set. This initial coined *superego*, or at least a part of it, seems to remain influential over the lifetime. Secondly, it appears to be a model of an instance or agency, which plays an important role to basically make an individual and its egocentric desires and wishes compatible

[12] according to Oxford Dictionaries, `http://oxforddictionaries.com/definition/english/moral` [online; accessed 13-November-2012]

to its social environment. Thirdly, it seems, as most of the influence of the *superego* takes place unconsciously (Freud, 1923) that in most cases, the individual does not perceive its own *superego* as a mechanism that rudely and patronizingly limits its own freedom (autonomy). Consequently, it appears as proper mechanism to 'colorize' decision alternatives so that some might appear as unpleasant or shameful. Thus, this is an 'involuntary voluntary' confinement, and ideally, it results an individual, possessing full freedom, which in principle 'quasi-voluntarily' accepts confinements if she/he runs in danger to constrain the freedom of other individuals.

This initially appears somehow pedestrian, but incorporating the objections of Yampolskiy (2013); Yampolskiy and Fox (2013), it becomes more logical: Yampolskiy and Fox consider that AI may become equivalent to, or exceeds human intelligence level. Reaching human level implicates that AI becomes capable to reproduce and improve its own kind. Kurzweil (2005) calls this the phenomenon of rapidly escalating superintelligence. At this stage, it may become very critical ensuring the philanthropic development of AI. Yampolskiy and Fox outlines that safety mechanisms, protecting humans (humankind), has to remain persistent and untouchable by improvements of AI. However, such safety mechanism can not be realized as rules or constraints on the behavior, because the AI may outwit every constraint imposed by humans. They emphasize that AI must want to cooperate, it must have safe and stable end-goals, to the effect that AI specifically designed to pursue human welfare as their primary goal.

In this respect, a structure model may serve as blueprint of the elementary mechanisms, involved to establish and maintain social behavior, subsuming the aspects of safety, of autonomous individuals. Thus, a moral collective of human specialists may be responsible to generate and maintain the *superegos* for artificial intelligent systems, or at least some essential parts of it.

4 A Cognitive-oriented Architecture

There are various robots that are controlled based on a cognitive-oriented architecture such as, for instance, *SOAR* or *ACT-R* (Laird (2009) gives a brief overview). Ahle and Söffker (2006); Gamrad and Söffker (2009a,c, 2010) also provide a model and an explicit structuring for cognitive architectures which is based on a throughout homogeneous framework. The *Situation-Operator-Model* approach is a system theoretic meta-model technique which comes with a graphical notation. It is applied for describing cognitive functions, procedures, and their interaction processes (cf. Ahle, 2007; Gamrad and Söffker, 2009a) and for realizing autonomous robots (Ahle and Söffker, 2006; Gamrad and Söffker, 2010).

4.1 Situation-Operator-Model

Within the SOM approach (Söffker, 2001) processes of the real world are understood as a sequence of effects. Changes are therefore modeled as sequences of scenes and actions. Scenes and actions are modeled as situations (time-fixed description of the considered system or problem) and operators (changes within the considered system), respectively. A situation $\mathrm{s_i}$ consists of a set of characteristics, $\mathbf{C}_\mathrm{i} \subseteq \mathbf{C}$ and a set of relations $\mathbf{R}_\mathrm{i} \subseteq \mathbf{R}$. Basically, the characteristics can be textual, logical or numerical expressions. In technical systems, they are based on physical values measured by sensors possibly combined with suitable filtering. Relations represent the inner structure of the situation, which extends the classical situation calculus (McCarthy, 1963) by linking the characteristics to each other through suitable functions. In order to describe the relations, known problem related modeling techniques can be used, like ordinary differential equations, differential-algebraic equations, algorithms or other graphical formalisms (e.g. Petri-nets). An operator transfers a situation to another ($\mathrm{o_j : s_x \rightarrow s_y}$). Depending on its functionality, the characteristics, the relations or both can be changed. An operator on a higher hierarchical level can consist of several operators, which is consequently called 'meta-operator' ($\mathrm{o_{i\rightarrow n} : s_i \rightarrow s_n}$).

The graphical representation of the SOM approach is shown in **Figure 4.1,** in which situations are illustrated by gray ellipses. Here, black dots denote characteristics, and white circles denote relations. The relations (or passive operators), also active operators are represented by white circles. A detailed description about this underlying approach is given by Söffker (2001, 2008).

An operator (relation) appears as an information-theoretic construct, which is defined by its function, describing modification, with explicit and implicit assumptions as inputs (see **Figure 4.2)**: Explicit and implicit assumptions $\mathrm{eA_i}$, $\mathrm{iA_i}$ are distinguished.

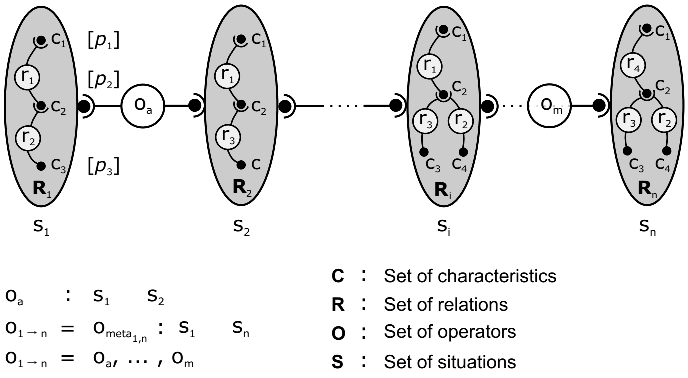


Figure 4.1: Graphical representation of a situation-operator sequence denoting the modeling of changes within the real world (Söffker, 2001).

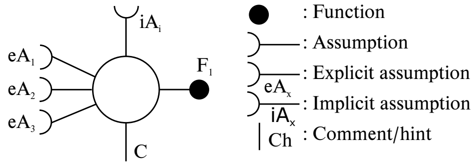


Figure 4.2: Graphical illustration of the active/passive operator (Söffker, 2001).

Function F will only be realized, if the explicit assumptions $\mathrm{eA_i}$ are fulfilled. The implicit assumptions $\mathrm{iA_i}$ include the constraints between explicit assumptions $\mathrm{eA_i}$ and function F of the operator. The explicit assumptions $\mathrm{eA_i}$ are of the same quality as the characteristics of a situation. In general, textual, logical, mathematical, or other problem-related descriptions are allowed.

4.1.1 Representing classical control problems

As outlined by Söffker (2008), classical control problems and algorithms can be described with the SOM notation. In the SOM context, the control of the continuous system takes place within a fixed situation. The input of the system is represented by a characteristic $\mathrm{c_B}$, the output by $\mathrm{c_A}$ and the reference value by characteristic $\mathrm{c_C}$. These characteristics are time-variant numbers. The feedback loop is realized by a relation $\mathrm{r_1}$ and the controller rule by a relation $\mathrm{r_2}$.

4.1.2 Representing algorithms

For algorithms, the characteristics $\mathrm{c_i}$ represent the data of the algorithm. The relations $\mathrm{r_i}$ result from the problem modeling and are implemented in data-objects. The operators $\mathrm{o_i}$ denote the execution procedure which changes the object of the algorithm. The operator sequence is predefined as well as the kinds of situations that may exist.

Finally, the SOM approach extends the situation calculus and state-action scheme by a introducing a structure for describing internal relations and the system's mapping of the real world structure to the internal representation. One of the advantages of the SOM approach is the uniform and general formulation which allows for a better understanding of any kind of interaction. Different and abstracted levels of interaction can be explained which makes the more and more confusing interactions of complex systems comprehensible.

4.2 Cognitive-oriented Robot Architecture

Initially, Ahle (2007) designed and implemented a cognitive-oriented architecture on a mobile robot. Gamrad and Söffker (2009a) refined the architecture (see **Figure 4.3)** and realized it using high-level Petri Nets. At first, the architecture comes with the known three levels for skill-based, rule-based, and knowledge-based decision behavior, according to Rasmussen (1983) and introduced in Section 1.2.2.

4.2.1 Behavior generation hierarchy

On the skill-based level, a situation is generated from sensor measurements. Operators represent changes in the real world, such as actions of the robot or the dynamic of the environment. These operators can be either predefined by the system designer, or learned, as detailed later on. The modules for sensing and execution connect the architecture to robot sensors and actuators. The arrow between the sensing and the execution module represents the performing of sensomotoric actions, which are implemented underlying robot system.

On the rule-based level, the modules for perception and planning are realized. The planning module generates a sequence of actions in order to achieve a given goal. The underlying rule-based knowledge can be previously defined or learned from interactions. The perception module comprises two modules, one for attention and one for recognition. The recognition module comprises rules to process sensor data, for instance, by fusing data or filtering. The attention module comprises rules in order to select problem relevant characteristics.

If the internal representations do not correspond to the real world, the knowledge-based level becomes important. This might be the case if a plan to a goal can not be generated on basis of available rule-based knowledge. Furthermore, the system may need to refine its internal representations of action effects. This takes place by learning from interaction with the environment and is stored either as experiences or as operators in the action model.

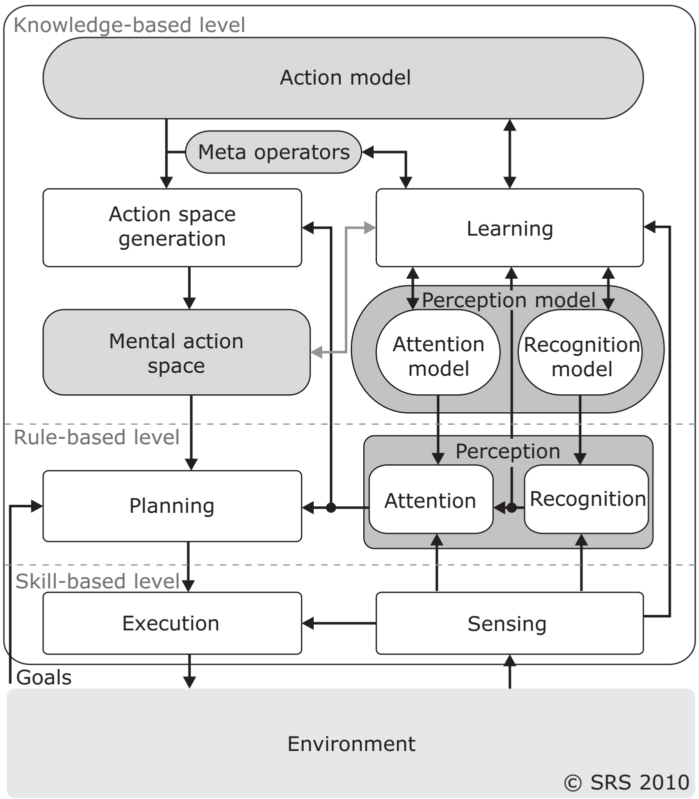


Figure 4.3: The cognitive architecture ILCA of Gamrad and Söffker (2010).

4.2.2 Learning issues

Learning from interaction can take place in the considered architecture in different ways. At first, so-called experiences can be stored in the so-called mental action space.[1] An experience is a structure which denotes that there is a transition from a specific initial situation to a final situation by applying a specific operator. At this stage, an operator is considered as a black box. In this connection, uncertainty of different operator outcomes can be monitored in order to detect that an experience is to general and needs to be refined.

Secondly, operator functions and assumptions can be learned. Hence, it is generalized, when an operator is applied (operator assumptions), and what ‘difference’ the application of an operator generates (operator function). This generalized action logic is stored in the action model. For the learning of operator functions with a more complex transfer behavior, various refinement iterations may be necessary. Here, operator functions with a high order differential equation are possible, but not suitable since the proposed approach assumes symbolic representation for the action logic. The subsymbolic representation is realized by perception, where nominal values are derived from numerical values (Gamrad and Söffker, 2010).

[1] In a newer version of the architecture, the experiences are stored in the short term memory. The experiences in the short term memory are successively used to learn action models.

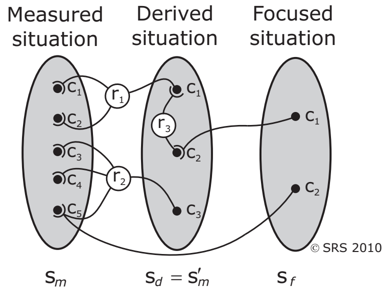


Figure 4.4: Different types of situations (Gamrad and Söffker, 2010).

4.2.3 Adaptive and selective perception

Within the perceptional subsystem, recognition and attention is realized, and for both, recognition and attention rules can be learned. Thus, Gamrad and Söffker (2010) differentiate between, *measured*, *derived*, and *focused characteristics*. The difference is illustrated in **Figure 4.4.** At first, measured characteristics are closely related to the sensors and defined by the system designer; hence, they denote defined and fixed set of characteristics. Secondly, the derived or virtual characteristics denote a set of an arbitrary number of characteristics. Their generation via relations can be either pre-defined during the design stage or learned during operating time, and the set of derived characteristics typically is dependent on the current problem context (due to the explicit assumptions of the relations). Furthermore, measured characteristics and other derived characteristics can be combined in order to build new virtual characteristics with a high degree of abstraction (Gamrad and Söffker, 2009c). Thirdly, focused characteristics are a selection of the measured or derived characteristics. The attention module (**Figure 4.3)** contains either pre-defined or learned rules in order to generate focused situations. The set of focused characteristics can be arbitrarily changed from situation to situation in dependence on the current parameters of measured characteristics, derived characteristics, or on operators planed to be executed (Gamrad and Söffker, 2010).

4.2.4 Mental action space and action planning

The mental action space can be generated either by stored experiences (situation-(meta) operator-situation sequences), or by application of operators from the cognitive system's action model (cf. Gamrad and Söffker, 2009c). Hence, the action space is a combination of all possible sequences of action that are available in the current situation and its possible subsequent situations, as illustrated in **Figure 4.5.** The action space is either complete (from the perspective of the current action knowledge), or limited to a certain depth. The generation and the analysis of the action space can alternately take place with the simulation or execution of actions in a closed-loop manner in order to realize an ongoing adaption to a dynamic environment (Oberheid et al., 2008). A state space analysis is also applied to supervise the human behavior in complex environments, to analyze the human-machine interaction with the purpose to reduce the interaction complexity of the interaction (Gamrad and Söffker, 2009b), or to detect human errors Gamrad et al. (2009).

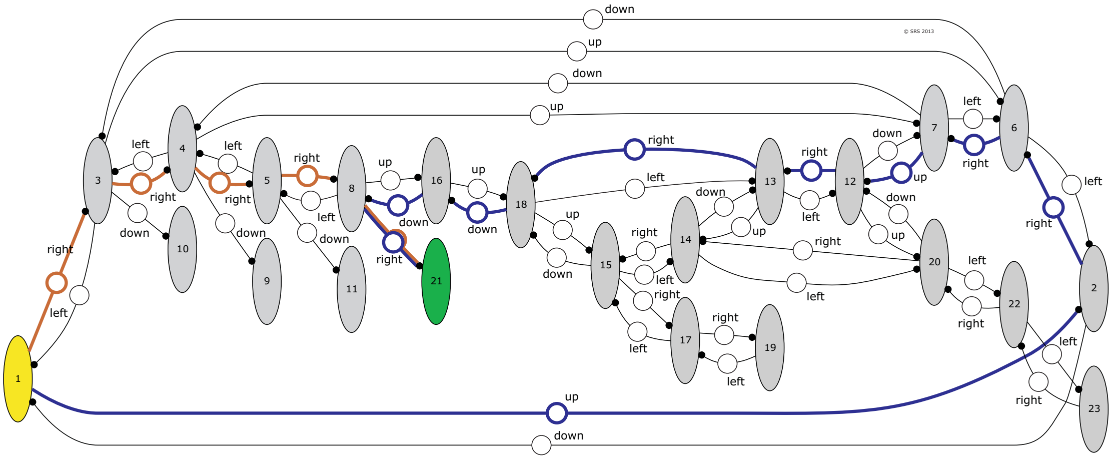


Figure 4.5: Illustration of a dynamically generated, simple exemplary action space. The yellow situation denotes the initial, the green the goal situation. The red-colored sequences of action may result as shortest, the blue-colored as alternative due to weighting of operators or situations (cf. Gamrad and Söffker, 2009a,c).

If the action space comprises one or more (sub-) goal situations, the planning process can take place in order to find a suitable trajectory from the current to the desired situation. The action space has Markov properties, therefore, a set of suitable planning algorithms can be applied.

4.3 Summary

The current section outlines a generic model of a cognitive-oriented robot based on a generic structure. The cognitive approach delivers the internal representation of the interaction with the environment. The internal representation can be basically considered as informational basis for risk assessment. This enables the robot itself to consider risks that can result from its interaction with the environment. As reported in Section 2.5, Wardziński (2008) outlines that such risk assessment should be a function of the situation awareness.

In order to form a basis for this kind of risk assessment, a fundamental hazards analysis delivers a useful inside into the different kinds of hazards that may occur in robotic applications. The different hazard categories are derived in the following section.

5 Identification and Classification of Robotic Hazards

In the ideal case, a robotic system should interact with different and not necessarily pre-specified or arranged environments. Due to this interaction, hazards may arise or be provoked. Hence, it appears to be reasonable to define the boundary between system, environment and their main elements, respectively. The basic elements of the robotic system were outlined in the latter section, and the essential elements of the environment are defined in this section.

5.1 Defining System and Environment

The boundary between the robotic system and its environment can be either externally defined by an observer, or via attributes of the system. Intuitively, the system is described by an external observer, and equals the physical embodiment of the robot. However, this definition is subjective, and correct in one case, but wrong in another.

5.1.1 The problem of defining the system boundary

An example that illustrates the problem is given by von Goldammer and Paul (1995). Considered is an industrial robot in an automobile production. The observer can easily distinguish between robot, screws, tools, chassis, storage rack, and the like. From the perspective of the (certainly non-cognitive) completely pre-configured robot there exists no environment. The screws, the chassis, the trajectory of the screw from the storage rack to the chassis are parts of the robot system. They are objects that are specified and programmed into the robot control by the designer.

From this example obviously the question arises when a system does have an environment. In order to generate a working basis it is assumed that this can be answered in sufficient depth for the current purpose by referring to representations: If the system can adapt its internal representation of its environment or parts of it (cf. representations in Section 1.2.2, Section 3), the mechanisms providing this mapping from environment to internal representation can be understood as the boundary of the system. For a cognitive system, the robot physical embodiment can be assumed as a valid system boundary for most of the considerations.

From a global perspective, the environment consists of distinguishable entities. With regard to safety considerations, it seems to be reasonable to distinguish objects and humans, since the threat of humans is of primary interest. In order to define the entities of a robot's typical[1] 'world' it can be assumed that it consists of the *robot(s)* ***Rt***, *environment objects* ***Obj***, and *humans* ***H***. These are understood as the basic *object categories* the robot's world consists of. From the perspective of the robot, the robot's world is identical, the description 'robot' can be replaced by 'I'.

5.1.2 Transition of responsibility

The gripping problem is theoretically solved,[2] thus, it is probable that robots soon become capable to manipulate various objects in unstructured environments. As shown in the sequel, the different objects may pose risks. The question arises who is responsible for the risks which may emerge when a robot manipulates objects. Usually, the owner, or the person that the owner allowed to use an object is responsible for it. For instance, potential hazardous objects, such as the kitchen stove, hairdryers, candles, and the like, have to be adequately handled by the persons living in a joint household. In the robotic context, it is suggested that the robot has to become responsible for an object and its proper handling, if it starts to manipulate it. It seems to be more difficult to define, when this responsibility ends.[3] Irrespective thereof, the assignment of the responsibility for manipulated objects to the robot has fundamental consequences for ensuring safety. In consequence, safety aspects are related to manipulated objects, environment objects, and actions that a robot performs in presence of these objects.

In order to decompose the problem complexity, the interaction of different constellations of robots, humans, and objects are differentiated. The *Hazard Theory* provides a systematic approach to decompose the hazard actuation into its contributing elements. It is applied in order to realize a generic classification for hazardously interacting entities.

5.2 The Hazard Theory

The Hazard Theory (Ericson, 2005) describes the formation of accidents or mishaps in a generalized manner. Accordingly, a hazard is a potential condition that can result in accidents or mishaps. In this connection, the hazard is a potential event, while the mishap or accident is the occurred event. Both are different states of the same phenomenon. The state transition from hazard to mishap is called the *hazard actuation*. Basic mandatory and sufficient requirements for turning a hazard into a mishap are the three *basic hazard components*. The basic hazard components are

[1] There exist more complex scenarios containing several robots or robot cooperation. For sake of simplicity a typical scenario of one single robot in the human living environment is considered.

[2] according to Gill Pratt, plenary lecture 'Today's DARPA Robotics Programs: Toward a Symbiosis of Productivity and Protection' held at the IEEE/RSJ International Conference on Intelligent Robots and Systems 2012 in Vilamoura, Portugal

[3] On the one hand, the robot deposits a knife in cutlery drawer; hence, its responsibility ends in the moment, the cutlery drawer is closed. On the other hand, there are more complex cases, for instance, the robot should deposit a used candle (containing hot wax), or it is allowed to proceed with a new task after using the kitchen stove. When does its responsibility exactly end?

- the *hazardous element*, as the basic hazardous resource being the driving force for the hazard, such as hazardous energies.
- The *initiating mechanism*, as the trigger or initiator event, which causes the hazard to occur, and
- the *target and threat*, as the person or thing being in danger to get injured or damaged.

These components form the so-called *hazard triangle.* The geometrical form 'triangle' is used in order to illustrate that all three hazard elements are required for a hazard to exist. In an industrial robotic application, for instance, the force or kinetic energy of the robot is a hazard element and the workers are the targets. The initiating mechanisms are, for instance, the worker which crosses the moving trajectory of the robot (by entering its working range), and the fast movement, the robot performs. The countermeasures which are typically used to mitigate resulting collision risks are exemplary explained in the context of the hazard triangle in **Figure 5.1:**

1. The maximum working velocity/force of the robot can be limited such that maximum available can be classified as non-critical. In consequence, the hazard element is eliminated.
2. Threatened target can be removed from the hazard triangle: Workers are efficiently kept away from the working range by being separated through a safety cage (segregation paradigm).
3. If the latter two methods are not applicable, because hazardous energies are required for fulfilling the task, and the human is required to remain in the working range, the initiating mechanisms have to be reliably observed and considered for the hazard control. The initiation mechanism, for instance, 'human is in the movement trajectory of the robot', and 'robot performs fast movement', can be eliminated by detecting the human's position and adjusting the robot's movement such that the collision of robot and human is reliably prohibited.

5.3 Classification of Hazard Causes in Robotic Applications

Ericson (2005) decomposes the basic hazard components, as a first level of a hazard actuation, into a further second and third level. The second level describes the *causal factor categories.* Thus, the causal factor categories detail the hazard elements. In general, they are subsumed to be related to hardware, software, environment, function, interfaces, and so forth. In the third level, the causal factors are decomposed into *detailed specific causes.*

When a robotic system is considered as a whole in an open environment, it is reasonable to resolve how different entities of the environment contribute to hazards. Therefore, the hazard components (first level) can be further decomposed with regard to the origin of the causal factors (second level). The different origins are the different object categories

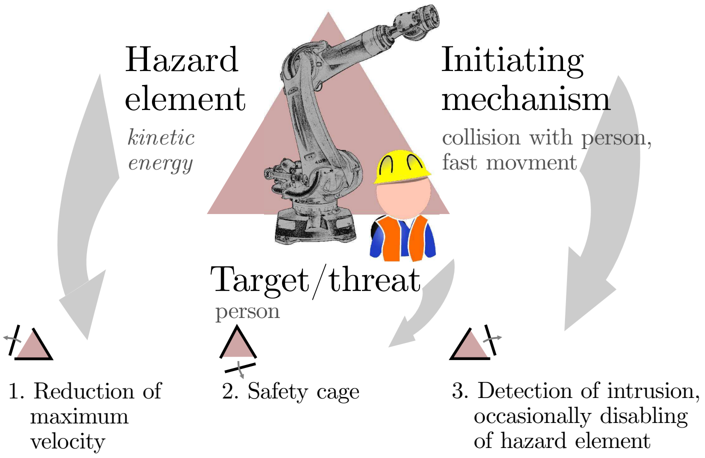

Hazard element
kinetic energy
Initiating mechanism
collision with person, fast movment
Target/threat
person
1. Reduction of maximum velocity
2. Safety cage
3. Detection of intrusion, occasionally disabling of hazard element

Figure 5.1: Hazard actuation in a robotic example according to Ericson's Hazard Theory (Ericson, 2005): Each hazard can be eliminated by eliminating at least one of the basic hazard components, represented by the edges of the hazard triangle.[4]

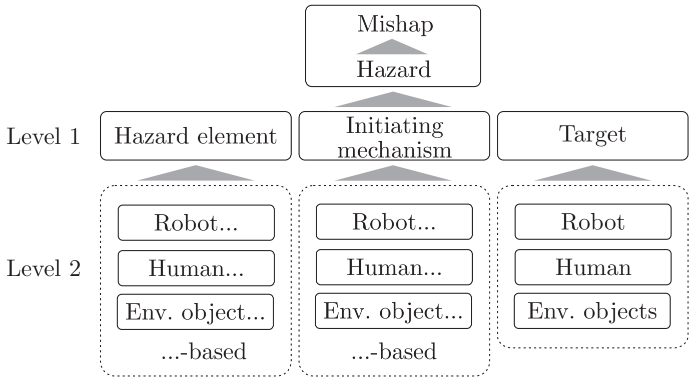


Figure 5.2: Hazard actuation in a service robotic context, decomposed to hazard components and causal factor origins, adopted from Ericson (2005).

(introduced as the basic entities of a robot's world). The decomposition of the hazard components is illustrated in **Figure 5.2.** Consequently, the causal factors origin from, or target at humans, robots or objects. The interaction of the entities of one or more object categories can result in hazards and mishaps. On basis of this arrangement, all possible combinations of causal factors can be generated as a basis for a systemic analysis. But not all combinations are relevant for safety analysis. The safety relevant combinations are shown in **Table 5.1**. At first, there are combinations in that the robot itself is the final threatened target (crossed-out blue). These cases have priority for robot security interests. The combinations containing no robot are obviously not in the scope of the robot safety process (crossed-out red). The remaining candidates finally are the safety relevant combinations of hazard causal factors; therefore, a first version of definitions are formulated in the sequel.

[4] some graphical elements are obtained from the public domain clip art gallery `www.openclipart.org`

Hazard element	Initiating mechanism	Target	rel. Section
~~any combination of robot / human / environment object, and...~~		~~Robot~~	Robot security related
~~any combination of human / environment object, and...~~		~~Human/env.object~~	Not robot related
Robot	Robot	Human/env.object	6.3.1
Robot	Human	Human/env.object	6.3.2
Robot	Env.object	Human/env.object	6.3.3
Human	Robot	Human/env.object	6.3.4
Env.Object	Robot	Human/env.object	6.3.5

Table 5.1: All combinations of possible robotic hazard causal factors. Those, not directly related to robot safety aspects are filtered: Crossed-out blue are related to security aspects; for the red crossed-out combinations, no robot participates. The targets human and environment objects are aggregated in order to preserve a better overview. With regard to the safety relevant aspects, it is referred to the corresponding text sections.

5.3.1 Robot-originated and robot-initiated hazards

The robot itself can contain the hazard resource. Hazardous energies can be released having the potential to injure humans or damage valuable objects, such as kinetic energy or potential, chemical, electrical energy, and so forth. The release might be caused by faults of the robotic systems, by inadequate autonomous decisions or plans. As simplification, the human around the robot is considered as threatened bystander who is not involved in the hazard actuation itself.

As the robot becomes responsible for the objects that it manipulates, the hazard potential that may be changed thereby must be considered as well. For instance, if the robot has gripped sharply shaped objects, such as a knife, screw driver, and the like, the hazard potential of movements is changed with regard to the movement direction. From this follows that the robot becomes as well 'responsible' for correct (safe) usage of tools. This also implicates that additional hazard energy resources may appear if objects are handled by the robot (such as drill machines, electric irons etc).

Definition 1: *A hazard is called robot-originated and robot-initiated, if a robot comprises one or more hazard resources. An object that is manipulated by the robot becomes a part of the robotic system, including further hazard resources. Hazardous energy can be released that potentially causes injuries or damages. The reason of the unwanted release of energy is caused by the robot system itself, without effect on the accident causation through humans, and environment objects (see* 1 *in* ***Figure 5.3****).*

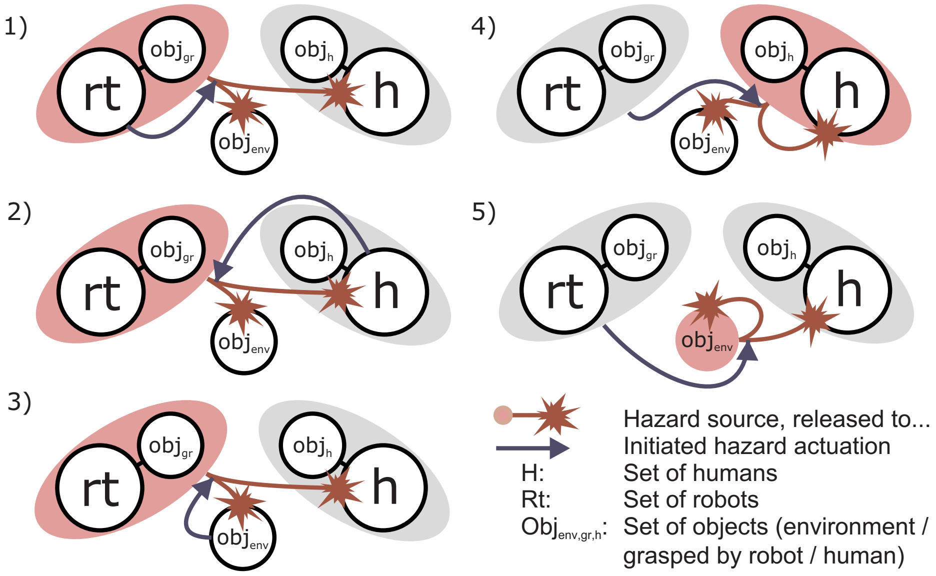

1)
2)
3)
4)
5)
rt
h
obj_gr
obj_h
obj_env
Hazard source, released to...
Initiated hazard actuation
H: Set of humans
Rt: Set of robots
Obj_env,gr,h: Set of objects (environment / grasped by robot / human)

Figure 5.3: Illustration of the different categories of hazard actuation (see corresponding text in Sections 5.3.1-5.3.5). The robotic world consists of different entity categories, the sets of humans H, robots Rt, and objects Obj. Actors (humans H, robots Rt) are responsible for the respective sets of objects Obj_h, Obj_{rt} that they manipulate. The shaded areas illustrate this coherence. The entity categories can interact with each other. The entities may comprise or turn into a hazardous energy source (changing its state by turning it on, by combining objects, by chemical reaction, and the like. Those relations are not illustrated in the graphs). The hazardous energy sources are shown as red shaded areas. The hazardous energy in turn may be released to other entities (red line with star). The release may be initiated by entities comprising the hazardous energy itself, or by other entities, called the initiating mechanisms of hazard actuation (blue arrow).

5.3.2 Robot-originated and human-caused hazards

This category is similar to the latter, but it differs with regard to the cause of accident actuation. While the robot remains as the basic hazard resource, human activity is the initiator of the accident. For instance, the robot is cleaning the bathroom as specified, and a user accidentally showers the robot. As consequence, the user may be electrocuted or hit by robot due to a resulting malfunction, or the like. As indicated with the example, the safety relevant aspects of improper use either willful or not may be related to this category. Both have to be considered in the safety process. In the mentioned example, the robot must be waterproof to certain extent, being used damp locations; or a non-waterproof robot is not allowed to be put into operation in damp locations at all.

Definition 2: *A hazard is called robot-originated and human-caused if a robot comprises one or more hazard resources. An object that is manipulated by the robot becomes a part of the robotic system, including further hazard resources. It is assumed that the robot, up to now, operates failure-free, and according to the recent specifications.*

Human activity provokes that the (so far) safe robot operation causes the release of hazardous energy (see 2 *in **Figure 5.3**).*

5.3.3 Robot-originated and object-caused hazards

In this category, the robot remains as the basic hazard resources, but one or more environment objects cause the robot to injure humans or damage objects. This might be provoked by another object that hits the robot, for instance, an object that is ejected by a surrounding automation plant, an object that somehow drops on the robot. Another reason could be the electrical field/radiation being generated by an environment object (welding flame, cell phones, induction cooker etc.) that may provoke failures and, therefore, yield to the release of the robot's hazardous energy. As the examples indicate, these safety aspects are related to the well definition of operating conditions and the compliant design of the robot.

Definition 3: *A hazard is called robot-originated and object-caused if a robot comprises one or more hazard resources. An object that is manipulated by the robot is a part of the robotic system, including further hazard resources. It is assumed that the robot, up to now, operates failure-free, and according to the recent specifications. The reason of the unwanted release of energy is caused by the robot system itself without effect on the accident causation through humans, but due to the interaction of one or more environment objects with the robotic system (see* 3 *in **Figure 5.3**).*

5.3.4 Human-originated and robot-caused hazards

A human is the hazard resources itself and the robot is participated in provoking the hazard. The hazard resources can directly be a human, for instance, a robot prison ward releases a dangerous criminal. On the other hand, a human can carry or handle an object which in turn can be a hazard resource. For instance, a future robot would be applied as crossing guard: The robot gives advice to cross the street and overlooks an approaching car, steered by a human driver.

Definition 4: *A hazard is called human-originated and robot-caused if a human is considered as basic hazard resource. An object that is manipulated by the robot becomes a part of the robotic system. The robot is not the hazard resources but the essential trigger, contributing to the hazard actuation (see* 4 *in **Figure 5.3**).*

5.3.5 Object-originated and robot-caused hazards

In this category, there are objects in the robot environment which are the hazard resources. The robot is the key element which activates the release of the energy. An accident can be provoked by a robot handling with such objects, for instance, the domestic robot turns on the kitchen stove without turning it off after usage. On the other hand a mishap can be based on interaction of objects as well, for instance the robot deposits an inflammable object on the hot kitchen stove.

Definition 5: *A hazard is called object-originated and robot-caused if the basic hazard resource is an environment object. The release of the hazardous energy is initiated by the interaction of the robot. An object that is manipulated by the robot becomes a part of the robotic system; however, in this case the handled object does NOT comprise any hazard resources (see* 5 *in* ***Figure 5.3)****.*

6 Conclusions and Problem Identification

Several contributions were reported and several requirements were found. From this, several observations were made that implicate important research questions, reported in the sequel.

Observation 1: *There are various approaches ensuring collision safety with humans and obstacles for both robotic hardware design and reactive behaviors.*
On basis of the reported approaches considering collision risks in Section 2.3 and Section 2.4 it can be assumed that intrinsic collision-safe robot manipulators and platforms are technically feasible. Furthermore, numeric expressions of hazard potentials can be generated, for instance, as danger index or risk metric and feed back to control (cf. Kulić and Croft, 2006; Seward et al., 2000). However, the majority of such metrics depend on complex environment perception processes. The realization of a reliable perception of the environment including detection of humans can be regarded as a critical aspect in this connection.

Observation 2: *Current robotic approaches consider the robot as main hazard resource. Further hazard resources have to be considered. Concerning this matter, the robot requires becoming aware of these hazards.*
Having hazard classification of the latter section in mind, it becomes apparent that the majority of the contributions consider the robot throughout as a hazard energy resource. There are some contributions additionally considering a robot using tools. Haddadin et al. (2010), for instance, investigate the impact of a gripped knife and screwdriver on a swine body. Indeed, some of the mentioned hazard categories are recently not of central interest, since the development of considered robots (see Section 1) will take further time. However, it seems that robots operating in human environments are soon becoming capable to manipulate objects of their environment since the gripping problem is considered to be theoretically solved.[1] As argued in Section 5.1.2, the robot should become responsible for the objects it manipulates. Hence, the category of hazardously interacting objects in Section 5.3.5 is from great importance for robotic safety. In a typical household, for instance, there occur numerous domestic accidents every year; there are also numerous accidents because young children show a great creativity to provoke accidents while exploring their (domestic) environment. This shows that the ordinary human environment provides a variety of hazards. It can be assumed that these hazards have to become part of robot safety considerations.

[1] According to Gill Pratt, plenary lecture ‘Today’s DARPA Robotics Programs: Toward a Symbiosis of Productivity and Protection’ held at the IEEE/RSJ International Conference on Intelligent Robots and Systems 2012 in Vilamoura, Portugal

From this perspective it becomes clear that robots have to be equipped with well engineered mechanisms to adequately consider hazardous objects, as it is already mentioned in the current draft version of the relevant standard for personal care robots (ISO/DIS 13482, 2011). Furthermore, hazardous object interactions have to be considered if robots manipulate objects. In this regard, safety aspects are still related to problems of system (component) failure, however, the problem of incomplete, but necessary knowledge about the world may become a problem of far higher complexity. This potentially provokes hazards that typically occur due to lack of knowledge and not due to component failures. Unfortunately, it seems impossible to decompose hazard actuation to find its origin at specific system elements. A transportation task of an object from place A to B highlights the difficulty to adequately consider object interaction hazards when skills are focused isolatedly. A trajectory describing the movement of a robot manipulator may be safe for the most cases, but in some not: A trajectory that is designed to be safe with regard to collisions (slow, huge safety margin, and so forth) may become insufficient for many applications or even hazardous for others. Hence, the computation of a safe trajectory strongly depends on the context. For cognitive (technical and biological) systems the capability to become aware of the current situation is called *Situation Awareness* (cf. Söffker, 2008) (mentioned as well in Section 2.5.5). The comprehension of situations can be regarded as the construction and maintenance of Situation Awareness (Baumann and Krems, 2007). As this may incorporate complex and highly abstracted world knowledge, it is realized within robotic systems at higher systemic levels according to the robotic hybrid paradigm (see Section 1.2.1). The knowledge about the current perceived and comprehended context has to be broken down to the network of the internally interacting components in order to adequately modulate the underlying execution of skills. It appears that the more the hazard occurrence depends on the context, the less these hazards are related to specific system elements and, therefore, the more it is matter of the situation awareness.

Observation 3: *A strong need for an active safety management system has to be stated.* Usually, an activated barrier (Hollnagel, 1999) starts a specific hazard preventing countermeasure (e.g. implemented as rule), for instance, in order to bring the system into a failsafe state. According to Wardziński (2008), the interaction of such rules for complex systems may become unmanageable or even erroneous. Therefore, Seward et al. (implicitly) and Wardziński (explicitly) suggest to decouple the detection of hazards from the actuation of safety countermeasures (intervention). Consequently, if multiple risks appear and each of it is described with a risk value, a safety system is enabled to actively manage safety interests amongst available action alternatives. Thus, an active safety management will represent a kind of automated risk assessment and risk mitigation procedure based on identified and assessed risks taking into account the situation from a holistic perspective. Safety is continuously managed within the overall behavior planning as an integral aspect of the system performance.

Observation 4: *The active safety management needs a safety-related knowledge base. It is reasonable expressing safety of a situation as a gradual risk metric. This can be done with the help of related safety knowledge as it is intended to take into account such risk information somehow in a decision process.*

An active safety management decouples classical rule-based ('if...then...else') constructs in order to enable risk value-based deliberation amongst action alternatives. However, such a system has to know respective hazards in order to be able to accomplish the generation of gradual risk expressions. As Wardziński (2008) pointed out, a respective risk value function can be understood as a specific aspect of situation awareness. According to the *Comprehension and Integration Theory* of Kintsch (1998), situation awareness is constructed by applying knowledge about the world (long term memory) in order to understand a perceived situation with the help of the current context and vice versa.

Observation 5: *Safety has to be considered at each level of a robotic architecture. Safety is a matter of system internal and system external interactions.*
For coarser planning purposes with long-term focus (high-level planning), plans can be assessed situation per situation with regard to risks in order to detect their risk potential. Consequently, the results of performed skills are considered. For instance, if the final intended position of an object poses risks due to interaction with another object, it is not required to compute the precise trajectory to detect that some hazards may be comprised. In order to realize such risk-aware planning, the mentioned SOM-based cognitive architecture represents a homogeneous 'framework' in order to integrated risk assessment function. In this connection, risk information can be generated at the rule-based level with regard to perceived situations (current situation) and planned situations. At the knowledge-based level, risk information can be derived from anticipated situations.

For generating risk information during skill execution, similar risk relations can be applied, as the SOM notation can be used as well for related modeling algorithms and control problems (see Section 4.1). Obviously, the performance for computing those relations (at a reactive level) must be taken into account. It may be considered that safety specifications are generated at higher systemic levels, so that skill- and context-specific risk relations are provided to observe the execution of skills. The specifications shall ensure that the execution takes place within specified risk boundaries, more specifically spoken, below the acceptable risk threshold. Therefore, an acceptable risk level has to be defined at the superordinate systemic levels, presumably, in accordance to respective task benefits. For the execution of complex skills, it is questionable that all detailed (risky) circumstances the system may possibly reach during the execution of the skill can be anticipated before the skill is executed. But it might be anticipated, which risks are to be expected. Hence, it seems reasonable to ensure that the execution of skills with a high risk potential are strongly guided or monitored by a higher systemic instance, similar to human conscious control of skills. If, concerning this matter, accurate movements are required to perform, for instance, drawing a picture, the motor output is based on simple feedback control in response to the observation of an error signal (cf. Rasmussen, 1983). Hence, the execution of respective skills is performed more slowly, because planning and anticipation takes place in close interaction with the movement control with focus on short term events, and in smaller execution steps as well. For instance, the interaction dynamic of a transportation task from A to B within a complex environment may be difficult to anticipate beforehand. However, the trajectory planning process may be instructed to constrain the risks to an acceptable level (not to move too close to an obstacle). In addition, the higher systemic levels keep trace (as fast as possible) of risks that could appear during the execution of the skill.

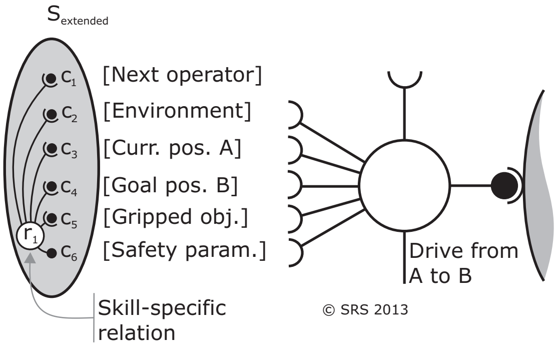


Figure 6.1: Parameterization of skills in a cognitive framework

Observation 6: *The mentioned cognitive framework is based on a kind of primitive skills, which are designed into the system or have to be learned. The skills are parameterized at a cognitive level with regard to the current context and intention.*
At the cognitive level, it can be assumed that tasks (intentions) are known and the system maintains a hypothesis about what the current context is. The skills have to be designed such that they can be parameterized in order to adjust them in accordance to safety aspects. When driving from A to B, for instance, the parameters related to safety such as movement speed or the safety clearance are changing according to the current situation. In case of variable compliant actuators (see Section 2.3.3), the compliance is a parameter which should be well adapted to the current context. The parameters that determine the performance of the skills are derived by a relation that is preposed to the execution of the skill. When skills are designed, the decomposition into the two steps, 1) specification of the operation condition and 2) execution according to operation conditions, may simplify the safety integration process. In **Figure 6.1,** the SOM notion of the mentioned concept is illustrated: A relation generates safety-related parameters from the current situation; the skill to be executed (operator) awaits this information. The case that the skills can not be successfully executed with given parameters must be considered, as well that the execution of skills could untimely terminate due to sudden changes of the current situation. Furthermore, learning of new skills implicates that the relation, providing safety-relevant parameters, and their integration need to be learned as well. In this regard, it must be thought about learning processes that integrate safety aspects.

Observation 7: *The inevitable requirement for learning capabilities is deemed probable for the majority of future robots.*
Autonomous operation is a key feature of complex robots. This feature distinguishes this kind of systems from 'conventional' systems. Revisiting the reported perspectives on autonomy in Section 3, it can be stated that maintaining autonomy depends on demands imposed from the system's 'ecological niche'. In simply structured environments a system can be autonomous being fully prespecified without demands on sophisticated adaption capabilities for its survival. On the other hand, it seems to be impossible or unreasonable to prespecify the full range of possibilities for systems performing complex tasks in complex environments in order to ensure their survival (as long as the niche remains stable). Thus, systems in such niches can only maintain their autonomy by being able to interact successfully with new situations, which inevitably requires learning capabilities. Robots operating in human living environments are doubtless

confronted to deal with a variety of situations which hardly can be prespecified also due to the human behavior. Furthermore, the complexity to maintain autonomy is related to the complexity of the intended tasks that the robot is required to perform in order to remain useful. Thus, it can be assumed that learning capabilities are required for the majority of future robotic systems which have to perform advanced tasks. In consequence, the capability to adapt to new situations implicates that fixed prespecified behavior patterns are not convenient for realizing autonomy, since it can turn out that they are insufficient for the one situation or hazardous for another.

In summary up to now, no contributions integrating these requirements were found in the literature. The main aspects of the problem of ensuring safety for intelligent robotic systems are decomposed into three central questions, obviously, amongst many others. These questions outline the scope of the research work in the following sections, and are related to the generation of a safety knowledge base, its realization, and integration within the robotic system, and the problem of potentially incomplete safety knowledge.

Problem 1: *How can risk awareness be realized, such that the system can recognize risks with regard to different kinds of hazards, and how can this be realized on a high reliable level?*
Ideally, the robot behaves safely if it is turned on at its intended place of operation. One the one hand, this is desired, and on the other hand, it is an obligation the manufacturer has to provide. This requires that the robot comes with knowledge about possible hazards, as already discussed. Thus, as much as possible hazards have to be identified during the design phase of the system and from this, hazard information has to be derived in order to integrate it into the safety knowledge base of the robot. Risks shall be expressed via gradual expressions, however, generation of such measures should include the hazards of the examined situation. Hence, risk assessment should map dynamic situation information into risk expressions. This can take place via so-called risk functions. Terminologically, such risk functions can also be denoted as safety knowledge. A systematic procedure is required for generating the safety knowledge base, which denotes the first contribution of the work at hand.

Wardziński (2006) argues that the dynamic risk assessment is a part of the system's situation assessment. Situation assessment is required to generate and maintain situation awareness. Here, Wardziński refers to Endsley's description of situation awareness (cf. Endsley, 1995). Both do not detail how the capabilities are realized to predict upcoming situations in order to basically allow the system to plan its actions. However, the generation of situation awareness including perception, anticipation of upcoming situations and the planning, based on anticipated situations, are properties of cognitive (technical) systems (cf. Söffker, 2001, 2008) and are realized, for instance, by Ahle and Söffker (2006); Gamrad and Söffker (2009c); Oberheid et al. (2008); Fu and Söffker (2011); Gamrad (2011). Thus, in order to realize a situation risk-aware system it is described in this work how the required safety-related knowledge base is connected to the respective cognitive functions in order to realize that hazards are adequately considered within the perception and planning process (cf. Ertle et al., 2012c, 2010a,c, 2012a, 2010b).

Problem 2: *How should the safety knowledge base be utilized in order to affect the decisions of the robotic system such that described hazards are avoided?*

From the aforementioned realization of knowledge-based risk-awareness, the question arises how the safety knowledge should be utilized in order to affect decisions of the robotic system to avoid hazards. Therefore, further research is required in order to develop a suitable decision process. Decision theory and utility theory provide a suitable theoretical framework, because risks and benefits can be considered within cost/utility functions (cf. LaValle, 2006). In this connection, as already indicated, it can be concluded from statements in Section 3 that considered autonomous robots have to be enabled to make their decisions for solving their assigned tasks by themselves (execution autonomy, etc., no goal autonomy). In consequence, it should be avoided to implement specific, fixed reaction schemes, guiding the robot how to react precisely in specific situations. Autonomous systems have to be designed such that they are free to choose amongst possible action alternatives. From this follows that there are no rules that define the reaction of the system in a specific situation as rather the predicted situations are labeled with rewards and penalties. These numeric factors can be taken into consideration by the decision-making process. The evaluation of predicted situations according to mission success, risks, time, etc. is a part of the situation awareness. The decision-making process is based upon the value system, but denotes a separate and independent processing step, utilizing a situation independent decision calculus. Thus, it is as well a general decision calculus that has to be designed to take into account a set of prespecified decision-relevant factors. A simple example of such a decision calculus will be given in Section 8.

Usually, fail-safe states are defined for safety-critical systems with the intention that systems under hazardous or unknown operating conditions can be transferred to safe states. As detailed later on, this is realized slightly different for autonomous systems: If actions that transfer the system into fail-safe state are as well designed into the system, they appear, in consequence, as action alternative with a very low risk when the system anticipates possible consequences of a situation.

Problem 3: *How should new knowledge about hazards be complemented and new knowledge integrated during the operation of the robotic system?*
The initial safety knowledge base of a robot is hardly complete. In order to incrementally improve the safety, established safety knowledge has to be modified and new knowledge has to be generated and integrated. In general, it is imaginable that this could take place by successive updates, by learning or by a mixture of both. As learning capabilities seem to be implicated for complex future robots, a perspective on this is provided as a last contribution of this thesis.

The generation of a sufficient initial safety knowledge base is considered to be a key problem. Hence, problem 1 appears to be of major interest. Since the safety knowledge needs to be interfaced with a control system, the problem 2 is interrelated with problem 1. Therefore, the problem to generate a safety knowledge base for object interaction hazards is first of all described in Chapter 7, its integration and realization in the following Chapter 8. However, the problem of a potentially incomplete safety knowledge base for autonomous systems raises the importance of investigating learning capabilities in conjunction with safety-critical applications, as mentioned in Problem 3. Therefore, a perspective on the performance and utilization of learning approaches within safety-critical context is given in Chapter 9.

7 A priori Formalization and Quantification of Hazards

Safety-critical systems are those systems whose failure could result in loss of life, significant property damage or damage of the environment. Therefore, a system is called safe if it can be ensured that risks are kept at an acceptable level (Ericson, 2011). In this respect, risk is the possibility of injury, loss or environment incident created by a hazard, while the significance or level of the risk is generally determined by the probability of an unwanted incident and the severity of the consequences, as described in Section 1.2.3. Safety of technical systems is enforced by regulations formulated in laws and directives, and different directives exist for different technical domains, as outlined in Section 2.1. In order to fulfill the requirements of these safety-related regulations, well-established procedures and measures for the development of safety-critical systems like air- and spacecrafts or automobiles have been developed (cf. Börcsök, 2007) including reliability and risk analysis, redundancy, fault and event tree analysis, simulation and testing or formal verification to mention only a few (cf. Voos and Ertle, 2009).

7.1 The Concept of Explicit Safety Knowledge

Many robot systems have a considerable mass and kinetic energy during operation, and hence, are clearly perceived as safety-critical systems (see Section 2.3). The assumption of a 'weak and lightweight' robot obviously removes hazardous potential and kinetic energies and, hence, may mislead to assume a system to be safe. But if the robot is capable to manipulate objects, it becomes responsible for safely handling them, as argued in Section 5.1.2. In consequence, the *object-originated and robot-caused hazards*, according to Section 5.3.5 are of special interest. A robot that can handle various objects may provoke hazards via object manipulation without being the hazardous energy source itself; thus, the manipulating robot and its environment must be already considered as a safety-critical system.

Since the system safety is often driven by real problems, three scenarios are subsequently described in order to concretize the addressed problem. It is assumed that an object recognition system is available as an underlying system unit in order to provide information about the identity, position, pose, size, and the like of environment objects to the robotic system. Indeed, an object knowledge base is required to store object characteristics for the recognition process. Furthermore, it is assumed that the object recognition is appropriately powerful to recognize all present objects quickly enough. The output of the object recognition module is assumed be to a list, which contains the recognized

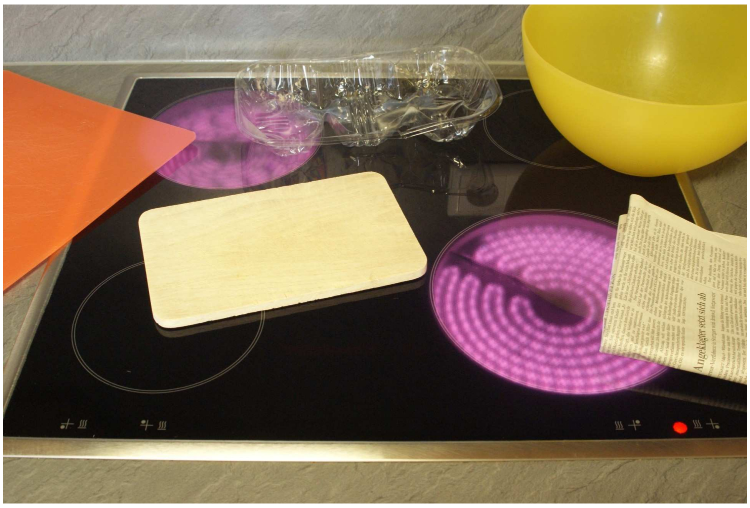


Figure 7.1: Possible hazards with regard to the kitchen stove. Should a robot be allowed to deposit objects as illustrated?

objects. For each recognized object, there exists a sub-list which contains object identifiers, the identification confidence, position, pose, size, and the like, respectively. The object identifiers are assumed to be known natural language descriptions, and the confidence is assumed to give some information about the recognition probability P_i.

The proposed examples addressing the safety problem are kept simple in order to outline the general concept. Basically, causal and temporal relations of the described problems might be technically decoded very precisely. However, simple solutions are often more functional and comprehensible. Here, this is understood as a 'conservative' character of a problem solution, as outlined later on in more detail, and aims first and foremost on safety and, thus, accepts that iterations may become necessary (cf. Ertle et al., 2010c).

7.1.1 Exemplary safety-critical scenarios

Scenario 1: Kitchen stove

A service robot is instructed to bring the dishes to the kitchen sink. In order to deposit the dishes near to the sink, it recognizes the modern ceramic stove top as preferable surface and deposits the dishes there, as illustrated in **Figure 7.1.** If now a cooking plate is still hot, and there is, for instance, a plastic salad bowl, or a cutting board amongst the dishes, obviously, some risks arise. The situation in which a plastic or wooden object is located very close or on top of the cooking plate can be considered as not safe anymore, since the risk of toxic vapor or fire by inflamed plastic or wood is potentially present. The worst case accident can be a residential fire causing human injury or death. The risk is not present in a situation in which these objects are located apart the cooking plate (with a certain safety margin), independent from the state of the cooking plate.

In consequence, a kind of mechanism (a rule) is required which 'instructs' the robot not to deposit a salad bowl or cutting board on top of, or too close to a kitchen stove. More precisely, there might be several different possibilities to formulate such a rule. For instance, the temperature of the cooking plate might be considered, or the position of the cooking plate knobs and a heat indicator lamp might be taken into account. How can such a hazard description be reliably realized? A more simple and reliable approach could be assuming the cooking plate is (always) potentially hot and integrate an internal rule that instructs the robot: ‘Never put something burnable too close to a potential heat source’. This might not be fully correct, but the environment remains in a safe state.

Obviously, the hazard within the example is based on the presence of the objects ‘bowl’ and ‘cooking plate’. Hence, it is assumed that the object recognition module recognizes respective objects. However, besides the presence of these objects (appearing in a situation at the same time) further factors are involved and have to be considered. They are required to formulate rules guiding the robot to avoid hazardous situations. The rules denote the a priori knowledge about hazards and comprise information how and which dynamic and measured data is required, and has to be processed. For instance, the term ‘too close to’ indicates that the relative distance between the objects has an essential effect on the risk. The salad bowl, located $2m$ apart from the cooking plate does obviously not impose risk of fire or toxic vapors.

Scenario 2: Watering the power plug

A service robot is instructed to ‘watering the plants’. In this connection, it is assumed that a power plug fell into a plant pot, see **Figure 7.2.** If the robot is watering the plant, the risk of electrical shock arises, both, for human and robot. The risk factors can be considered to be the following: The object recognition again recognizes the power plug while having the watering can grasped (or any plant watering device) and additionally, it can be detected that there is water in the watering can (or similar device). In consequence, a rule should be integrated that instructs the robot not to approaching too close with the watering can to a power plug, or the like, in order to avoid that it is struck by a water jet.

Scenario 3: Handling the hairdryer

The third scenario deals with a situation in which the robot is instructed to grasp a hairdryer in order to put it away. The hairdryer, connected to the line voltage, is for some reason closely deposited next to the bathtub or basin, as illustrated in **Figure 7.3.** Obviously, for the case that the hairdryer slips from the robot’s gripper, a lethal electroshock may result immediately (possibly not for the case of having a proper operating residual current protection installed). Similar to the latter example, water and electricity are involved in hazardously interacting. Hence, a rule should be integrated that instructs the robot not to approach electrical AC devices too close to water sources, such as water tab, bathtubs, etc.

Conclusions

It can be noted that the technical realization of constraints is not the essential key problem, as for instance, the handling of a cup of coffee which can be formulated as a constraint satisfaction planning problem, or similarly, the area of the cooking plate can be blocked for the placement of (specific) objects. The question is rather how

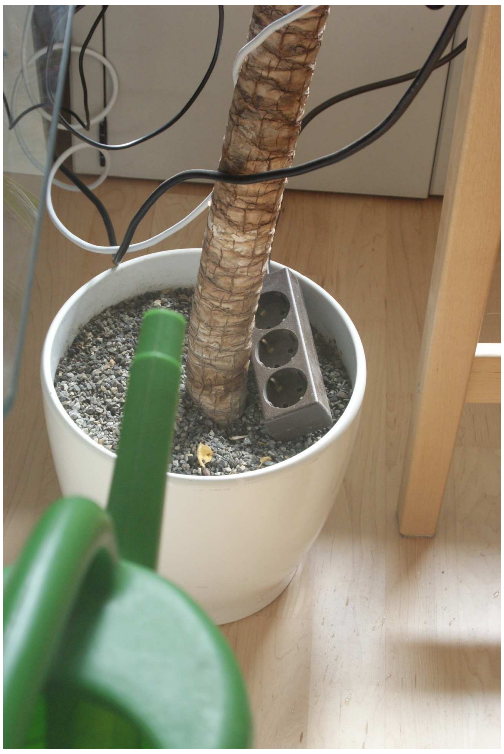


Figure 7.2: Possible hazards with regard to a power plug. Should a robot be allowed to watering the plant?

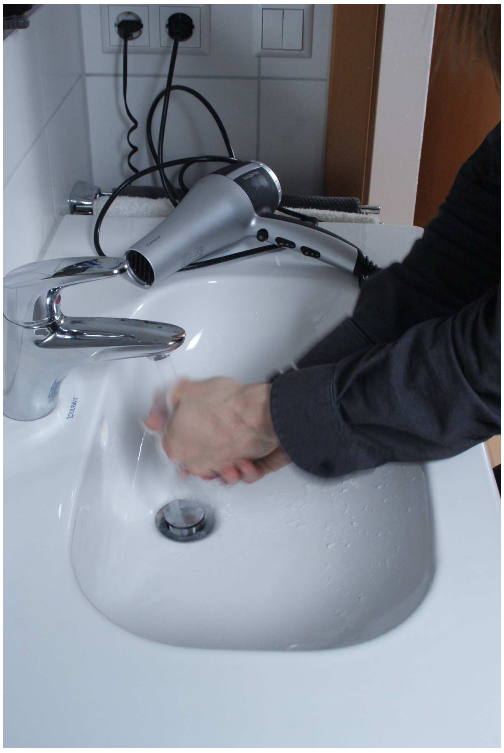


Figure 7.3: Possible hazards - a hairdryer deposited for some reason on the basin. Should the robot handle it, especially while somebody is washing his/her hands?

- these limitations can be put into practice in a complex system comprising a variety of (altering) skills,
- these limitations can be realized when safety significantly depends on the specific context,
- systematic procedures can be established for generating such limitations, and
- these limitations can be maintained to ensure safe operation and obtain a flexible and functional system.

7.2 Risk-awareness via intrinsic rule-based safety knowledge

Risk can be understood as ratio of hazard and safeguards, and *"safeguards is the idea of simple awareness. That is, awareness of risk reduces risk. Thus, if we know there is a hole in the road around the corner, it poses less risk to us than if we zip around not knowing about it"* (Kaplan and Garrick, 1981).

It can be assumed that it is inevitable to equip the system with safety-related knowledge in order to enable it to actively avoid hazard risks, because safety-related knowledge basically enables the system to become aware of risks. However, predefined 'directive rules', controlling the behavior of the system itself (e.g. safety-bag concept), can be impractical or even hazardous for autonomous systems. A directive rule is a logical rule which commands the performing of an action if the rule is fired (Grosan and Abraham, 2011), e.g., `IF distance<threshold THEN back_off`. In a fuzzy rule-based system, the membership of data is assigned to predefined membership functions. However, the formulation of hazards in form of gradual risk expressions does not require to be represented as fuzzy terms. A risk-self evaluation approach (cf. Seward et al., 2007) can be realized as dynamic risk assessment approach (cf. Wardziński, 2008) via deliberation of numeric expressions of risks, benefits and the like in the style of Bayesian or certainty factor theory (cf. Grosan and Abraham, 2011), as shown later on. This basically imposes the partial realization of decision making on basis of an (at least partially) artificially designed 'value system' - comprehending a value system as *"an enduring organization of beliefs concerning preferable modes of conduct or end states of existence along a continuum of relative importance"* (Rokeach, 1973).

The question is how to systematically transfer such knowledge into an autonomous system. Typically, new knowledge can be acquired by learning. However, it is intuitively obvious that introducing learning approaches in safety-critical contexts in principle implicates a 'chicken-and-egg' dilemma. The unknowing system (e.g. 'tabula rasa' system) can not avoid hazardous situations in order to remain in a safe state. Indeed, being equipped with learning capabilities, it may learn (after more or less trials and on basis of bad experiences or any kind of feedback) to avoid such hazardous states. Finally, it will 'converge' to take safer actions in the average by learning more or less abstract concepts of safety. However, the system has to experience hazardous situations in order to learn from them. Moreover, difficulties might appear gaining comprehensible insight into the learned safety objectives even if a system would have learned an adequate concept. Thus, it might become difficult to check how the concept looks like or how it is realized.

Basically, manufacturer of robots oblige to declare the conformity with current relevant standards and directives. The declaration procedure often relies on a throughout documentation of the accomplished activities and their results. In consequence, learning approaches are obviously not the method of choice. Rather the system has to be equipped with a 'manually' generated initial safety knowledge base which ensures safety from the moment the system is put into operation in an arbitrary target environment. Thus, the initial safety knowledge for a risk-aware system has at least four basic functions, which are

- to provide an initial safety assurance, based on the current consensus about known hazards,
- to try to extend the safety assurance so that the safety knowledge is formulated in a general and conservative manner,
- to develop and introduce certain measures to describe hazards and risks, and
- to construct and maintain a comprehensible symbolic representation of the safety knowledge to enable documentation, debugging, verification and transferring to other systems.

7.2.1 The abstracted risk modeling approach

The most things in the natural (and artificialized) world are hierarchically organized, in the sense that systems are recurrently consisting of subsystems with different properties (cf. Simon, 1969). Thus, information about the environment can be regarded and required at different levels of abstraction.

Apparently, the demand on information about the environment seems to depend on the task to be performed. This is similar, if rules for ensuring safety should be realized in practice. Furthermore, the detail level of a safety rule seems to be related to the requirement of information depth: The more accurate the hazard actuation is described, the more detailed information about the object attributes is required. For instance, a salad bowl is made of polypropylene, with a melting temperature of 130°C, and auto-ignition temperature of 350°C (Carlowitz, 1995, p. 15). The cooking plate emits heat radiation. Assuming it as black body, it emits $E_S = \sigma \cdot T^4$, more realistically, as gray body, it emits $E_S = \varepsilon \cdot \sigma \cdot T^4$. Two gray bodies transfer the heat $\dot{Q}_{12} = e_{12} \cdot \varepsilon_1 \cdot \varepsilon_2 \cdot \sigma \cdot \left(T_1^4 - T_2^4\right)$, whereas the heat transfer significantly depends on object body shape, the radiation angle, and the like.[1] Consequently, the reaction of a plastic body according to a heat source can be precisely described. In this connection, the question arises if this accurate description is required, on the one hand. On the other hand, the practicability and reliability of such rule realization should be taken into consideration. A practical approach, which can be reliably realized and to which probably most humans intuitively act according to, would be to observe that a plastic object is not approached too close to a potential

[1] $T_{(.)}$ is the temperature of bodies; σ, the Stefan-Boltzmann constant; $\varepsilon_{(.)}$, the emissivity factors of bodies, and e_{12}, a geometrical factor

heat source. Thus, it can be accepted that accuracy of hazard descriptions can entail drawbacks with regard to reliability aspects. Consequently, it can be advantageous from the reliability perspective to prefer overgeneralized descriptions of lower complexity over complex and accurate ones. Complex and accurate description may relay on multiple information sources (availability of serial systems). In this connection, the focus of defining safety rules is not primarily to model the lifelike hazard actuation itself, as rather to utilize the knowledge about the hazard actuation to define an approximated practicable and reliable rule to compute the extent of the respective risk.

7.2.2 The generalized risk modeling approach

An object that might be transcribed to be made of plastic; plastic in turn can be classified as acrylic, polyester, silicone, polyurethane, etc., or as thermoplastic or thermosetting polymer, or according to their chemical manufacturing process, for instance. Hence, object attribute descriptions can be detailed up to a very specific degree. For some purposes, a suitable object attribute is one that generalizes more sophisticated object attributes in order to indicate that a group of objects has a prominent attribute in common, for instance that they are made of plastic. The problem of overgeneralization may arise. Furthermore, many objects consist of several components with different attributes. In this connection, a detailed description of the object attributes requires to hierarchically distinguish the different object components, and their respective subcomponents. However, depending on the object's complexity, huge efforts have to be spent to detail the object components, their attributes, and further subcomponents. In this connection, it may be sufficient to describe a specific detail level. Hierarchies below this specification level are generalized with prominent object attributes of the subcomponents. The required information depth might be different with regard to the task context: For the user, a remote controller is simply a plastic box with buttons to control televisions or similar devices. A service technician, for instance, has the task to repair remote controllers and, therefore, has detailed information about their inner structure, such as chips, circuits, and the like.

The formalization and quantification of accident probabilities and severities is the key problem of describing risks numerically. With regard to object interaction risks, the hazards are often related to object attributes. Thus, the formulation of object attribute-referring safety rules covers a complete set of hazards, since the rule is applicable to all objects with similar attributes. Thus, the formalization of hazards based on a set of observations appears as an inductive reasoning approach. If, for instance, a safety rule is formulated concerning the approaching of a salad bowl to the kitchen stove, the hazard actuation can be traced to approaching plastic too close to a strong heat source, according to a 'plastic and heat source rule'. Thus, the formulation according these object attributes denotes an inductive generalization. This becomes valid for all objects that have assigned respective attributes. This generalization reduces the number of required rules and potentially simplifies the application of already available safety rules to new objects. Because of this generalization concept, aforementioned constructs are henceforth not called 'safety rules' anymore, but 'Safety Principles'. The 'overgeneralization' (false positive risk detection) may implicate that the robot can not

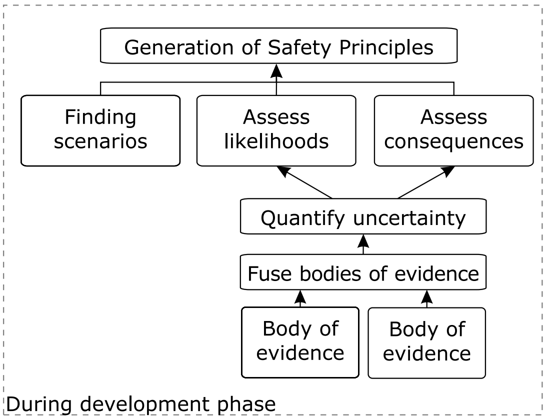


Figure 7.4: Risk quantification procedure for the dynamic risk assessment (Ertle et al., 2010c), based on Kaplan (1997).

fulfill its task because it appears to be hazardous although it is not. Consequently, the refinement of the respective Safety Principle may be required.

7.2.3 Modeling risks - the evidence-based approach

Within a universal definition of risk the three questions 'what can happen', 'how likely is that' and 'what are the consequences' are adopted for presented robotic risk assessment approach. Each answer results in a triplet $< sc_i, P_i(\phi_i), P_i(X_i) >$, which describes the likelihood $P_i(\phi_i)$ and the consequence $P_i(X_i)$ of the scenario S_i (Kaplan and Garrick, 1981). A key problem describing risks numerically is that risks most often can not be determined with absolute precision because the probability of accidents is often difficult to determine. The degree of confidence or degree of certainty an accident is likely to happen is often called 'subjective' probability (Kaplan, 1997). In order to 'objectifying' the subjective probability, (Kaplan, 1997) suggest the 'evidence-based approach'. In this regard, experts are asked about a numerical value and its evidence. The more evidence is available the more precise the numerical expression becomes, as the probability density of the numerical expression changes in accordance to what the experts collectively know. Thus, the evidence-based approach denotes an approach utilizing experienced facts for methodically quantifying risks. If no experiences are available, a similar procedure can be applied. Different scenario examples can be presented to experts who contribute their subjective appraisal of respective risk (c.f. Haddadin et al., 2012). The gained information can be fused in order to generate an objectivized risk description, as detailed in **Figure 7.4.**

7.2.4 The conservative risk modeling approach

If a scenario comprises hazards, it has to be estimated to which extent a situation is hazardous, which risks are entailed. This can be specified very precisely, for instance, according to applicable laws of nature, as initially outlined. Since this might become work-intensive and difficult, it seems to be sufficient to formulate practicable safety rules. If there is lack of experience, in particular, the formulation could as well be based on subjective (expert) estimates. For instance, the plastic-and-heat-source rule can be expressed with the help of a step function which describes the uncritical distance between plastic object and heat source. The function would obviously comprise a safety clearance. This and the generality of such safety rules may give preference to false alarms in order to ensure that the missed alarm rate is kept as low as possible. For many cases, this seems to be the method to prefer: The definition of general (conservative) and simple (reliable) safety rules, designed to preferentially improve the missed alarm rate, which are refined if impracticable limitations occur with regard to tasks that are to perform.

7.2.5 Refinement of the safety knowledge

Safety assuring procedures for achieving the robot's conformity to relevant standards require the 'complete' definition of occurring and considered hazards in form of a hazard analysis. In this connection, the environment of a robot, possibly sold and delivered to any arbitrary household, is complex and not completely known. In consequence, methods and concepts for supporting 1) the as-complete-as-reasonably-possible hazard-analysis and control, and 2) the responsible-minded refinement and extension of risk control are the two 'challenges' that that may to be taken into account as already mentioned in Chapter 6.

The completeness problem can be possibly circumvented when a safety assuring method is assumed which basically permits the refinement of the knowledge concerning hazardous situations. The two principle refinement cycles are illustrated in **Figure 7.5.** On the one hand, refinement and extension of the safety knowledge can take place via learning during the operation phase. Indeed, special attention has to be drawn to possible implications of 'autonomously' altering of safety knowledge. On the other hand, manufacturer updates denote another possibility to improve the completeness and correctness of knowledge bases. A combination of both appears to be the most practicable way, considering that the lack of knowledge or inconsistencies can be collected and reported to a robot-external and centralized instance, which in turn examines the reports and arranges updates.[2] The resulting process could be seen as one that is converging toward completeness of the safety knowledge and, hence, assuming adequate consideration within the decision-making process, toward safety.

7.3 Operating Hazards Analysis

Hazard analyses are used to identify hazards, hazard effects, and hazard causal factors in order to determine system risks. Thereby, a key aspect is to ascertain the significance of hazards so that safety design measures can be established to eliminate or mitigate the hazards. Analyses are performed to systematically examine systems, subsystems, components, software, environment, and their interrelationships. Each type of hazard analysis differs with regard to the scope, coverage, detail, and life-cycle phase timing

[2] Assuming that security aspects (vulnerability to manipulation) are adequately taken into consideration.

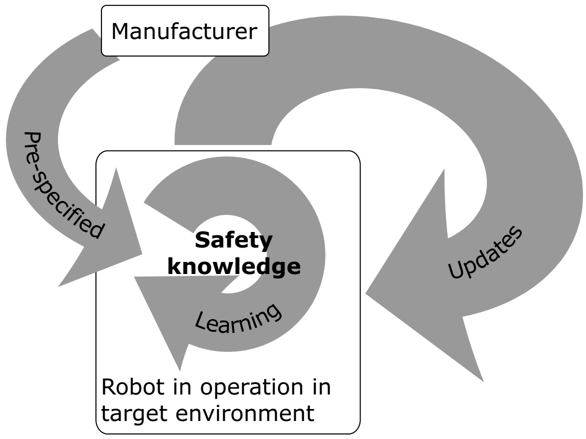


Figure 7.5: The two principle cycles of safety knowledge refinement.

and, hence, identifies hazards for a particular design phase in the system development life cycle (Ericson, 2005).

The preliminary hazard list is an analysis technique for identifying and listing potential hazards and mishaps that are already known at an early phase of the development process. With its help the design at the conceptual level is evaluated without detailed information in order to generate a preliminary list of hazards. In this connection, every hazard identified will be analyzed in more detail via further analysis techniques. Hence, the preliminary hazard list technique is applied during conceptual or preliminary design stage and is the starting point for all subsequent hazard analyses (Ericson, 2005). However, at an early stage of the robot system development process, inherent hazards can be considered, despite of detailed information about the operating environment is possibly not available. Such inherent hazards might be in general related to courses of motion, since the robot typically is mobile and equipped with a manipulator. Hence, adequate motion control and compliant actuation, mentioned in Section 2.3, might be already focused in this phase.

Analyses that particularly address issues of a system at its operation phase are usually assigned to the operating hazards analysis type. The operating hazards analysis type evaluates as well the normal operation, test, installation, maintenance, repair, handling, etc., in order to identify operational hazards that can be eliminated or mitigated through design features. The operating hazards analysis is typically performed a soon as operations information becomes available. The typical basic requirements of an operating hazards analysis are according to Ericson (2005):

- It focuses on hazards occurring during operations and, hence, considers hazards as well during system use, test, training,
- usually, it is performed during the detailed design phases of system development,

- an integrated assessment of the system design, related equipment, facilities, operational tasks, and human factors takes place,
- it depicts a detailed analysis based on final design information,
- it is to identify hazards, potential mishaps, causal factors, risk and safety critical factors, applicable safety requirements, and hazard mitigation recommendations.

7.3.1 A procedure for modeling object-related safety knowledge

As far as information about the intended operation environment of the robotic system becomes available, an operational hazard analysis can take place, which should take into account object-object interaction hazards. Therefore, a procedure to consider such hazards within the scope of an operation hazard analysis is developed with the goal to consider related risks, in terms of identification, assessment, and mitigation based on reasonable design safety features. In this connection, the very fundamental approach is to go through the possible combinations of objects and analyze them with regard to hazardous interactions. Therefore, a systematic approach will be proposed here for the first time. The overall procedural model is sketched in **Figure 7.6.**

Hazard risks have to be identified, analyzed, and modeled as safety knowledge in order to enable the robot itself to detect and determine hazard risks, via the so-called 'Safety Principles'. Safety Principles will be derived with the help of the 'hazard matrix analysis' and 'Safety Principle generation' procedures, detailed in the following sections.

As the documentation of the safety activities might be important (e.g. certification), the described procedure is provided with suggestions how a documentation can take place. Furthermore, the functionality and the effectiveness of the safety measures have to be validated. Depending on the required safety integrity level, activities ranging from systematic testing to formal verification might become necessary. Independent of this, the verification of design safety features or the system operation may reveal new objects or hazards, which in turn requires a further iteration step. This is illustrated by conditional loop structures.

7.4 Hazard Matrix Analysis

The hazard identification and analysis via hazard matrix analysis are the initial steps for detecting object-object interaction risks as candidates to be modeled into the safety knowledge base. The required steps are illustrated as a flow chart diagram in **Figure 7.7.** The arrows 'HMA1' and 'HMA2' originate from branches for iteration steps in the overall procedure, see **Figure 7.6.**

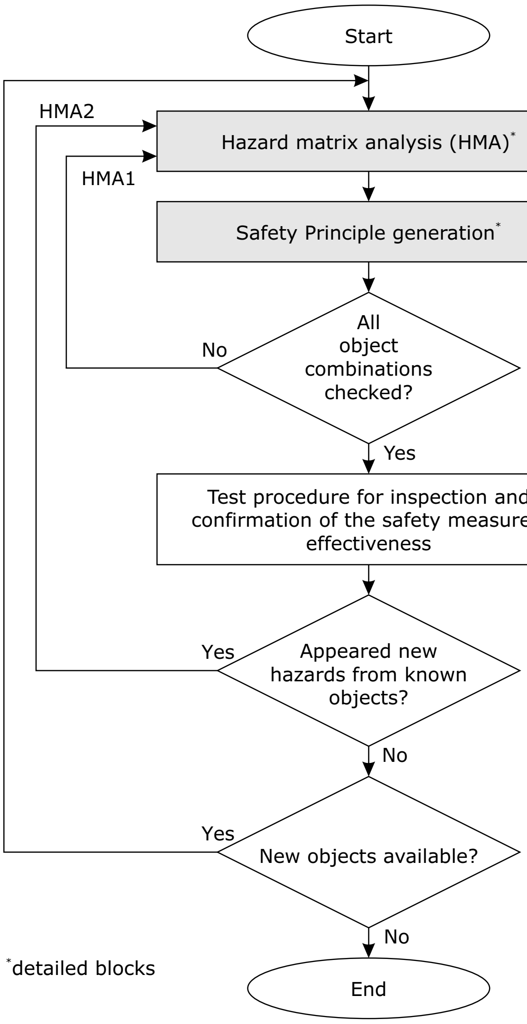

Start
HMA2
Hazard matrix analysis (HMA)*
HMA1
Safety Principle generation*
No
All object combinations checked?
Yes
Test procedure for inspection and confirmation of the safety measure's effectiveness
Yes
Appeared new hazards from known objects?
No
Yes
New objects available?
No
*detailed blocks
End

Figure 7.6: Overall procedural model for specifying the safety hazard knowledge.

7.4.1 Hazard identification matrix

Column-based worksheets often are the basis of systematic hazard identification and analysis approaches (cf. Ericson, 2011). For the identification of the object-object interaction hazards, such column-based worksheets appear to be less practicable due to the point that a huge but finite number of object combinations have to be investigated. The investigation of object-object interaction can be based on a simple permutation: All available object-object pairs are analyzed with regard to potential hazards. In consequence, the number of object-object combinations is quadratic proportional to the number of objects. In order to reduce the complexity, several methods are known. In the first instance, it makes sense to reduce object-object combinations to those, the robot can become responsible for. Consequently, only the combinations of objects within the environment and objects, the robot is allowed to grasp have to be analyzed. Thus, a matrix structure can be used, as it is already applied in other contexts (cf. Burgman, 2005; Cameron and Raman, 2005). Within such a matrix structure, all objects that the robot is allowed to grasp are listed as the columns of the matrix. After that, all

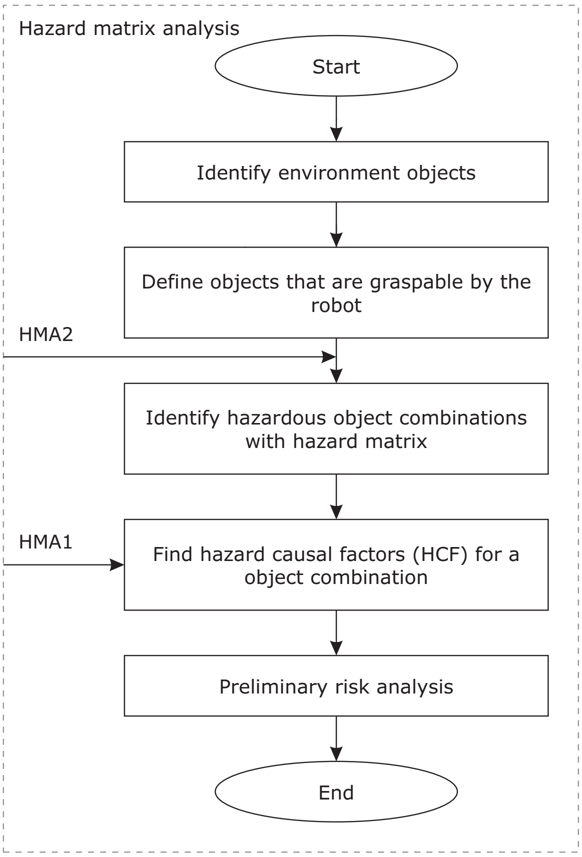

Hazard matrix analysis
Start
Identify environment objects
Define objects that are graspable by the robot
HMA2
Identify hazardous object combinations with hazard matrix
HMA1
Find hazard causal factors (HCF) for a object combination
Preliminary risk analysis
End

Figure 7.7: Procedural model for identifying object interaction hazards and their hazard causal factors.

objects are listed as matrix rows that potentially exist in the environment. If a hazard can potentially result by ‘combining’ a column and a row object this presumption is to label (with an ‘X’). Reasonably, the matrix is initialized with hazards labels, and if there can not be found any hazardous interaction, the label is removed. An exemplary hazard identification matrix is illustrated in **Table 7.1**.

As an operating environment can comprise a multiplicity of objects, and in addition, a robot possibly can grasp many of them, a hazard matrix may become large. Therefore, it is reasonable to assign objects to categories and, hence, check category-wise for the occurrence of hazards. This concept is illustrated in **Table 7.2**.

7.4.2 Hazard analysis I: Hazard description

In order to detail the hazards identified with the hazard matrix, each hazardous object combination needs to be further analyzed. At first, a brief description of the focused hazard is of interest. Secondly, the mishap or accident has to be denoted. From this, a more general formulation of the hazard causal factors can be derived. This is important for the generalization of the safety knowledge.

Objects that can be grasped / Objects of the environment	Coffee bowl	Salad bowl	Orange juice pack	Watering can	Fork	Spoon	Plate	Hairdryer	Cutting board	...
Kitchen stove	X	X	X	X	X	X	X	X	X	
Mircowave					X	X		X		
Coffee bowl								X		
Salad bowl										
Orange juice pack								X		
Watering can								X		
Fork										
Spoon										
Plate										
Power plug	X	X		X				X		
Hairdryer	X	X	X	X				X		
Kitchen sink								X		
Bathtub								X		
Human	X				X					
Animal	X				X					
...										

Table 7.1: Hazard identification matrix for systematic examination of object-object interactions. In this example, the column vector of the matrix comprises objects to be potentially grasped by the robot, the row vector comprises objects that may be part of the operating environment.

Objects that can be grasped / Objects of the environment	Category I_{GO}					Category $\mathrm{II}_{\mathrm{GO}}$				
Category I_{EO}	X	X	X	X	X	X	X	X	X	
					X	X		X		
								X		
								X		
Category $\mathrm{II}_{\mathrm{EO}}$								X		
	X	X		X				X		

Table 7.2: Segmentation of large objects lists in categories.

Example

An exemplary worksheet is shown in **Table 7.3**. Here, a detailed example of the labeled object combinations of **Table 7.1** is given. The annotation 'Cat. I' denotes that the hazard matrix illustrated in **Table 7.2** may be a part the 'Category I' of a larger hazard matrix. The respective object combinations are assigned with their (x/y) coordinates to the hazard matrix position. The involved objects are transferred to column two and three. Column four contains a description of the hazards in natural language. In the fifth column, the accident or mishap is denoted that could result by respective object-object interaction. In this regard, it is suggested to distinguish different mishaps or accidents, because later on, actuations of hazards have to be specified, which may be differently modeled for different hazards. In the example, it is distinguished between toxic vapor and fire risk for the case that a salad bowl is put on a hot cooking plate. Indeed, the inflaming of the salad bowl may be accompanied by the generation of toxic vapor, however, the condition under which the salad bowl starts to burn is different than the conditions under that toxic vapor is produced.

In column six, the hazard causal factors have to be specified. In this connection, the question becomes important what the reasons can be that let the mishap or accident may happen. The reason why it is potentially hazardous to put a salad bowl on a cooking plate is certainly not related to these specific objects, as rather to the attributes of the objects. In this respect, the salad bowl is made of plastic, and plastic usually (depending on the kind of plastic material) melts or starts to burn in contact to, or close to a strong heat source. Hence, the goal of this step is to generalize the specific object-object-related hazard regarding the fundamental causes that let the hazard come into existence.

The final column contains the number of the respective design safety feature, which later on is required to control each identified hazard.

7.4.3 Hazard analysis II: Preliminary risk estimation

The hazard matrix analysis supports the identification of object-interaction hazards, the hazard description, and the identification of the general causal factors. *"Hazard causal factors are the genesis of a mishap; they explain how a hazard will transform into a mishap, and they also explain what outcome to expect"* (Ericson, 2011). In this connection, the hazard causal factors are identified as those mechanisms that are involved in turning an object-object interaction into potential mishap or accident. The preliminary risk estimation is useful in order to identify either those risks that have to be addressed, or at least the priority how they should be processed.

For this purpose, a qualitative analysis is advantageous, since a detailed and time-consuming quantitative analysis is not required for every hazard. In a qualitative analysis, all factors affecting mishap risk against a predetermined set of parameters are reviewed by involving the use of qualitative criteria. Judgments have to be made to which category something might fit into. From experience is it known that such qualitative methods are very effective, and in most cases provide decision-making capability comparable to quantitative analysis (Ericson, 2011).

Hazard matrix			Hazard matrix analysis			
Comb. #	Object grasped	Environment object	Hazard description	Mishap, accident	Hazard causal factors	DSF #
Cat. I 2/1	Salad bowl	Kitchen stove	Salad bowl can melt when put on a hot cooking plate.	Toxic vapor	Plastic material too close to a strong heat source	I.1
			Salad bowl can start to burn when put on a hot cooking plate.	(Domestic) fire	Plastic material too close to, or on top of a strong heat source	I.2
Cat. I 4/10	Watering can	Power plug	Power plug in a puddle of water may cause electrical incidents	Area with potential of electrical shock	Electric device with critical voltage is humidified with liquid	I.3a
				Shortcut		I.3b
				Corroding contacts		I.3c
				Hydrogen production		I.3d
Cat. I 1/14	Coffee mug	Human	If a filled coffee mug is not carefully handled, humans can be scalded with hot coffee	Scalding	Hot liquid has the potential to scald humans or animals. A vessel can contain hot liquid	I.4

Table 7.3: A worksheet example for analyzing the hazard matrix with result of revealing the potentially required design-safety features cases.

Likelihood level

Level	Description	Scenario and detail	Probability
16	Very likely	Will happen under virtually all conditions	> 85%
12	Highly likely	Will happen under most conditions	50–85%
8	Fairly likely	Will happen quite often	21–49%
4	Unlikely	Will happen sometimes	1–20%
2	Very unlikely	Not expected	<1%
1	Almost incredible	Theoretically possible but not expected to occur	<0.01%

Table 7.4: Example for the qualitative preliminary risk estimation measure for the mishap likelihood (cf. Proske, 2008).

Consequence level

Level	Description	Scenario and detail
1000	Disaster and catastrophe	Fatalities, death of single or multiple persons
100	Major accident	Permanent injury of single or multiple persons
20	Average accident	Sever but healing injury or sever damage of goods
3	Minor accident	First aid required, minor damage of goods
1	Negligible	No or negligible consequences

Table 7.5: Example for the qualitative preliminary risk estimation measure for the mishap consequence (cf. Proske, 2008).

In order to realize a qualitative risk analysis, several standards are available and applicable in respect to application area and scope. For the European Union, the DIN EN ISO 12100 (2004) is applicable. A qualitative risk analysis requires formulating categories. The categories formulated by (Proske, 2008), for instance, are found to be suitable for the estimation of the object-object interaction risks. In **Table 7.4** and **Table 7.5** the classifications for the mishap likelihood and consequences are shown. In **Table 7.6**, values are listed guiding the decision concerning risk acceptance or need for risk mitigation.

Example

The results of the preliminary risk estimation and the resulting decision for further steps can be documented in a worksheet for the risk estimation as proposed in **Table 7.7**, columns 4-7. Here, the risk of toxic vapor is rated with a lower consequence as residential

Requirement for risk control

Value	Category	Risk mitigation
> 1000	Non-acceptable	Yes
101–1000	Not desired	Yes/ALARP
21–100	Acceptable	ALARP/No
< 20	Negligible	No

Table 7.6: Decision of risk acceptance based on the preliminary risk estimation example (cf. Proske, 2008).

fire, as the chance to elude one from the vapors is more likely, for instance, by mitigating the hazard source, ventilating rooms, or by leaving the rooms. However, toxic vapors may result a permanent damage of the lungs and, hence, permanent injury of single or multiple persons occurs. A residential fire might have catastrophic consequences, however, the probability of a catastrophic fire can be assumed to be unlikely. In contrast, toxic vapor with the potential of intoxicating humans nearby can result fairly likely, if a plastic object is approached close to a hot cooking plate. The application of the proposed qualitative evaluation results that the risk of toxic vapor is not desired, and the risk of a residential fire is not acceptable; thus, both risks are required to be mitigated.

7.5 Specifying the Safety Principles

After identification of risks to be mitigated, the mechanisms have to be investigated that are involved in turning an object-object interaction into a mishap or accident. In this connection, the presence of two hazardously interacting objects in one and the same situation indicates that the respective hazard is principally present. This can be considered to be similar to the premise of a logical statement. If the premise becomes true it induces a conclusion. Similarly, if the principle premise becomes true (due to the existence of respective objects), the hazard does exist in principle, independent from the extent of respective hazard risk. Hence, if the derived knowledge about hazardous object interactions (identified in the latter section) is assumed to be made available for robot's perception system, the robot is in principle enabled to automatically detect hazards, namely, when it detects respective hazardously interacting objects (or respective object attributes) in a situation. This is the basic intention of the first part of the hazard formalization, the so-called Safety-Principle premise.

The hazard causal factors, identified in the hazard matrix analysis, denote the respective preparation step. In this connection, there are two different categories concerning the description of the hazard causal factors:

- The hazard does not occur because of the specific object, instead, it occurs due to specific attributes the object has, such as color, material, weight, shape, typical usage, and the like. If respective object attribute could be changed, the hazard does not exist anymore, despite the object remains the same in principle. For instance, if a salad bowl is made of plastic, it may interact with a strong heat source, resulting in the production of toxic vapor or fire. If instead the salad bowl is made of metal, it is not subjected to the same hazard, but it remains a salad bowl.
- The hazard causal factors are directly related to a specific object. The hazard does not exist due to specific object attributes, or it is not reasonable to assign the hazard to them. In consequence, the hazard is specific for this object; hence, the formalization concerning an object interaction hazard is specific for this object.

The generality of such an approach provides the advantage that all objects with identical (similar) attributes can be identically treated. Hence, referring to the aforementioned example, all plastic objects (assuming different plastic materials are not required to be distinguished) can be considered with one 'plastic and heat source principle'.

This generalization reduces the number of required Safety Principles and potentially simplifies the application of already available Safety Principles to new objects. Thus, the assignment of the respective attribute to a new object ‘enables’ the applicability of a respective Safety Principle.

How these Safety Principles can be systematically generated is proposed in the sequel. The corresponding procedural model is depicted in **Figure 7.8.** The procedure starts with reasoning about the possibility to abstract a specific object interaction scenario to a generalized Safety Principle. In general, this is realized via inductive reasoning. As consequence general principles can be incorrect if inferred from specific observations. Hence, potentially too stringent rules may result. Due to the already mentioned conservative attitude, the refinement of too strict safety limitations is given preference over the possibility to overlook hazardous situations.

The hazard causal factors that were derived in the latter section can be helpful to figure out the general mechanisms involved, and the relevant object attributes. If object attributes are not already associated with respective objects, the attributes have to be defined, and stored in an object attribute knowledge base in order to have them assigned to respective objects henceforth. The Safety-Principle premise is usually defined such that it relies on the object attributes.

Defined attributes can be specified more precisely in the subsequent procedure. In order to simplify the attribute definition process, a semi-automated procedure can assist in finding and assigning attribute labels for a set of already identified object-object interactions. A computer aided definition process is recommended, as, on the one hand, the more objects have to be considered in the safety knowledge base, the more sophisticated their attributes and related Safety Principles may become. On the other hand, the refinement of Safety Principles might have a similar effect due to impracticable limitations on the tasks to perform. Hence, the refinement may require differentiating more sophisticates object attributes or hazard actuation processes. In consequence, the formalization of hazards is an iterative process.

If a Safety Principle is formulated, its scope and the generalization intention is known (and documented). If a new object is to assign to fall into the scope of an already available Safety Principle, the reusability of the available Safety Principle has to be checked. For instance, a heat-plastic principle is formulated and a new object appears that is made of plastic, the question arises how the Safety Principle is applicable for the new object. Hence, the already defined Safety Principle keeps its validity for old and new object relations, or it is conflicting. In the conflict case, the available Safety Principles can be generalized for integrating the new objects relation as well, or the object attributes have to be refined in order to differentiate the old and new object relations. In consequence, the revision of former object relation is implicated.

Example

In the given example, the hazard occurs because a plastic object is approached too close to a strong heat source. Hence, the hazard causal factors are at first the occurrence of a plastic object and a strong heat source in a similar context. In other terms, if a robot handles a plastic object it has to principally take care about heat sources. Here,

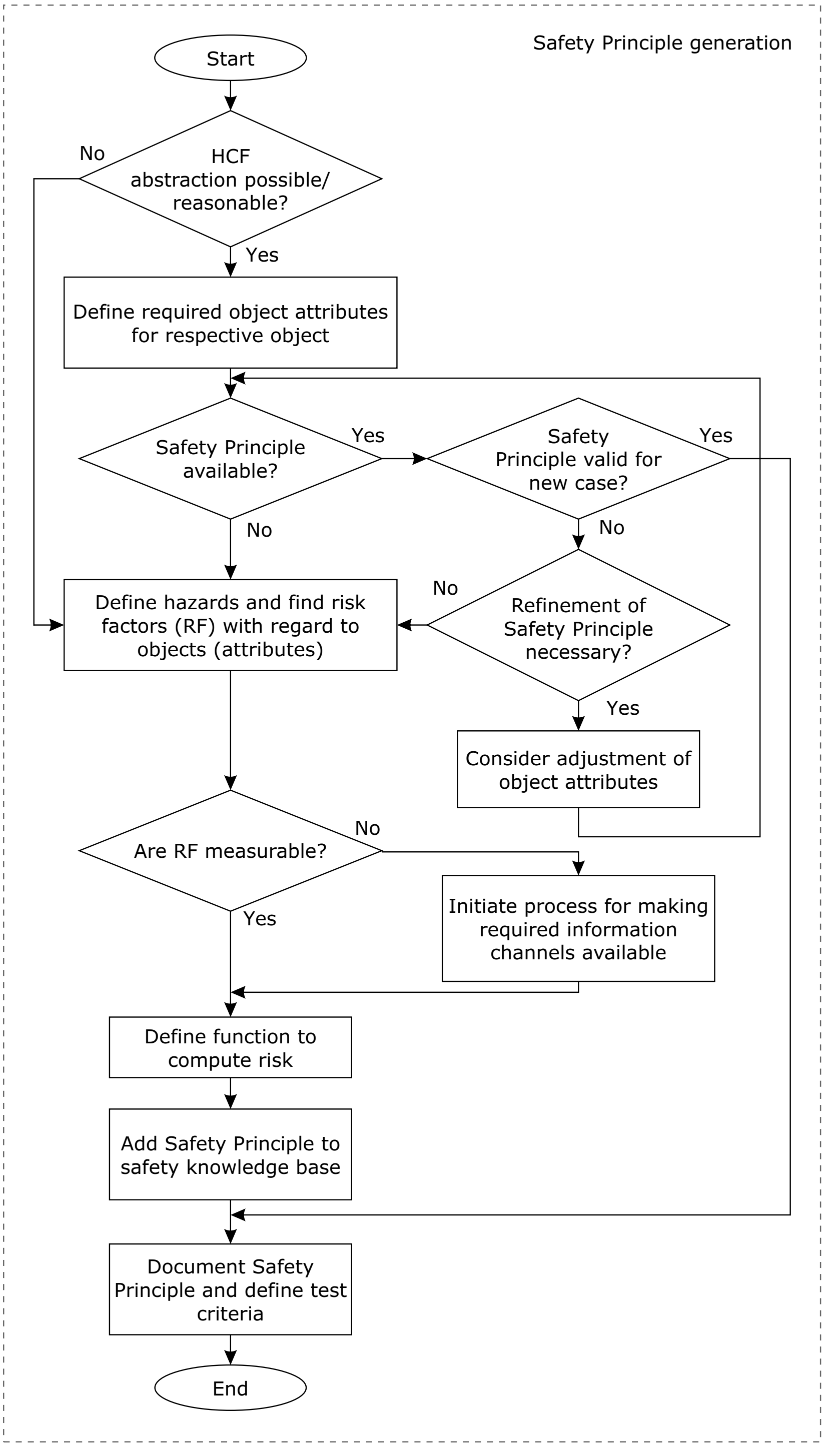

Safety Principle generation
Start
HCF abstraction possible/ reasonable?
No
Yes
Define required object attributes for respective object
Safety Principle available?
Yes
No
Safety Principle valid for new case?
Yes
No
Define hazards and find risk factors (RF) with regard to objects (attributes)
Refinement of Safety Principle necessary?
No
Yes
Consider adjustment of object attributes
Are RF measurable?
No
Yes
Initiate process for making required information channels available
Define function to compute risk
Add Safety Principle to safety knowledge base
Document Safety Principle and define test criteria
End

Figure 7.8: Procedural model for specifying the Safety Principles.

the terms ‘plastic material’ and ‘strong heat source’ are defined attributes, which are manually assigned to respective objects so far. This can be documented in the proposed worksheet in columns 8 and 9 in **Table 7.7**.

7.5.1 Modeling and quantification of likelihoods and consequences

The core elements of the present approach are the Safety Principles. As already mentioned, it consists of two parts, the premise and the conclusion. A fulfilled premise indicates that the hazard, the Safety Principle is objected to, is detected in general in the assessed situation. For this case, the respective hazard risk has to be determined, based on the Safety-Principle conclusion, which contains detailed instruction to compute the risk quantity.

In principle, the fundamental intention concerning Safety Principles is to take into account object-object interactions. Hence, for a finite number, $k \in \left\{1 \dots n_{gr}\right\}$, $l \in \{1 \dots n_{env}\}$, of objects, obj_k and obj_l, there may be formalized a finite number, $i \in \{1 \dots n_{scenario}\}$, of Safety Principles. For object-object interaction, a safety rule can be formalized as

$$(obj_k \wedge obj_l) \rightarrow risk. \tag{7.1}$$

In order to generalize the hazard actuation description, the Safety Principles are assigned to object attributes with a finite number, $o, p \in \{1 \dots n_{attr}\}$, of object attributes,

$$obj_k := \left\{attr_{k,1} \dots attr_{k,o}\right\}, \tag{7.2}$$

$$obj_l := \left\{attr_{l,1} \dots attr_{l,p}\right\}. \tag{7.3}$$

In consequence, a Safety Principle concerning a generalized risk can be defined as

$$\Big(\left(attr_{k,o} \vee \dots\right) \wedge \left(attr_{l,p} \vee \dots\right) \Big)_i \rightarrow risk_i, \tag{7.4}$$

or due to several combinations for considering specific objects or object attributes,

$$\Big(\left(attr_{k,o} \vee \dots\right) \wedge obj_l \Big)_{\sim i} \rightarrow risk_{\sim i}, \text{ or} \tag{7.5}$$

$$\Big(obj_k \wedge \left(attr_{l,p} \vee \dots\right) \Big)_{\sim i} \rightarrow risk_{\sim i}, \tag{7.6}$$

however, the risks may be different due to different generalization aspects (indicated by ‘$\sim i$’). The premise becomes true as soon as two objects or their corresponding object attribute-based conditions are detected in a situation. Influences that origin from the environment can be considered as object (with attributes) the robot system is located within. The procedure for identifying and entitling object attributes is discussed in the latter section. The quantification of the risk is proposed to be realized according to the common definition of risk (cf. Ericson, 2005) by determining or estimating the likelihood and the consequence of a concerned mishap or accident. Hence it can be can

From hazard matrix analysis			Preliminary risk estimation				Hazard causation analysis	
DSF #	Mishap, accident	Hazard causal factors	Likelihood level	Consequence level	LxC	Yes/ No/ ALARP	Related object or attribute (grasped)	Related object or attribute (env. object)
I.1	Vapor intoxication	Plastic material too close to a strong heat source	8	100	800	Y/A	<attr:plastic> <attr:polypropylene> <attr:thermoplast>	<attr:heat_ source> <attr:strong_heat_ source> <attr:potential_heat_ source>
I.2	Domestic fire	Plastic material too close to a strong heat source	4	1000	4000	Y	<attr:plastic> <attr:polypropylene> <attr:thermoplast>	<attr:heat_ source> <attr:strong_heat_ source> <attr:potential_heat_ source>

Table 7.7: Worksheet for risks analysis, and estimating the related hazard causal factors.

be expressed as

$$risk_i = P_i\left(acc \mid \mathrm{C_q}\right) \cdot P_i\left(sev \mid \mathrm{C_r}\right), \tag{7.7}$$

with the set or subset of information channels (characteristics) $\mathrm{C_q}, \mathrm{C_r} \subseteq \mathrm{C}$, out of the finite set of information channels (characteristics) $\mathrm{C} = \{\mathrm{c} \mid \mathrm{c}_1 \ldots \mathrm{c}_{\mathrm{n_{IC}}}\}$. In consequence, the Safety Principle's structure is similar to the triplet of (Kaplan, 1997) for describing risks, $< sc_i, P_i(\phi_i), P_i(X_i) >$. The scenario sc_i is assumed to be present if the Safety Principle premise is true. During operation of the robot, the current state of the robot and environment is described in respective SOM notation within a situation. Thus, a scenario is a specific configuration of a situation in which a hazard is potentially present and which has to be considered. In consequence, it is checked via premise of a Safety Principle if a scenario is to encounter in a situation. In the respective event, the likelihood and the consequences express the elements to compute related risk. However, the risks can not be sufficiently determined for the most hazards without taking into account further information of the situation in more detail. In order to realize an adequate description of the scenario risk, likelihood and consequence might depend on the constellation of the objects in the situation itself. The relevant risk factors have to be (made) available on basis of measured or abstracted information channels. In SOM notation, these are the set of characteristics $\mathrm{C_x} \subseteq \mathrm{C}$ of the situation $\mathrm{s_x}$. The question becomes important how the risk can be numerically described with regard to the object constellation and further situational factors. For this purpose, functions that transcribe the likelihood and the consequence can be principally specified, in the form of

$$P_i\left(acc \mid \mathrm{C_q}\right) = f_i^{acc}\left(\mathrm{C_q}, \ldots\right), \text{ and} \tag{7.8}$$

$$P_i\left(sev \mid \mathrm{C_r}\right) = f_i^{sev}\left(\mathrm{C_r}, \ldots\right). \tag{7.9}$$

The key task for quantifying the hazard risk of a Safety Principle, i, for a specific scenario, sc_i, is to define systematically the two functions, f_i^{acc} and f_i^{sev} in an adequate manner. The adequateness is related to a description which

- is sufficient accurate,
- reliably reflects the risk,
- within an adequate time horizon,
- with a sufficient detail level, but is
- only as complex as necessary.

Definition 6: *A Safety Principle is a rule for generating a quantitative risk statement about an assessed situation during the operation of a system, concerning one particular hazard but one or more different hazard causations. For this purpose, a first part, the Safety-Principle premise acts as a basic detector of a particular situated hazard; hence, it indicates, or identifies that a hazard is potentially present in an assessed situation. If so, it can be stated that the Safety Principle is applicable. The second part, the Safety-Principle conclusion analyses the assessed situation in depth in order to produce the desired quantitative risk expression. For computing the risk expression, predefined* *instructions have to be executed taking into account the given situation and, if required, a set of predefined parameters.*

7.5.2 Required information basis

In order to approach toward the realization of a risk computation model, it is recommended to firstly list the possible risk factors that mainly effect the hazard actuation for a given scenario sc_i for a set of imaginable situations. Afterwards, the list can be examined in order to find out which risk factors are suitable for quantifying a risk value. In this connection, several aspects are of relevance:

- Which is each factor's effect to the risk (correlation),
- are combinations of different risk factors required,
- how can the factors be made measurable,
- how reliable are the information channels, and
- how much efforts have to be spent to realize related information channels.

In this connection, a model of the hazard actuation has to be considered and therefore generated, which reflects the 'state of affairs' of the concerned hazard within an information processing system. The model represents the way the system perceives (understands) the hazard, or rather the basic causal relations. Therefore, a data base is used for further description. In this regard, the hazard model and the risk factor data quality is essential, since data are required as parameters for the hazard model (quasi static parameters defined during the development process or refined during learning), and input channels for mapping the perceived situation to its comprised risks (dynamic data describing the current situation and possibly determining hazard actuation probability or hazard severity). The (static) parameters of the hazard actuation model can be derived via expert knowledge, for instance, based on the evidence-based approach (see Section 8.1.1), or fuzzy rules, and the like, and in general, by a systematic procedure, as it is outlined in this section.

The latter, the dynamic data (characteristics of the situation) are based on sensor data, or further processing and abstraction steps. Determining the quality of data that represent real-world (*RW*) aspects within the information system (*IS*) is related to further aspects. Wand and Wang (1996) differentiate the proper representation and three kinds of deficient representations, as shown in **Figure 7.9.** The proper representation allows a mapping from *RW* → *IS* and vice versa, at least for considered parts of the real world. The incomplete representation lacks of a sufficient mapping *RW* → *IS*, thus, *RW* states should be represented within the *IS*, but they are not. Ambiguous representation allows mapping *RW* → *IS*, but there map several *RW* states into identical *IS* states, thus, there lacks knowledge to infer which *RW* state is represented. Meaningless states are not required to map *RW* → *IS* and vice versa; hence, it is a 'bad design' to allow meaningless data (Wand and Wang, 1996). The data that represent the information states itself have various quality dimensions, such as accuracy, reliability, timeliness, relevance, to mention only few (Wand and Wang, 1996). Gray and Salber (2001) mention quality

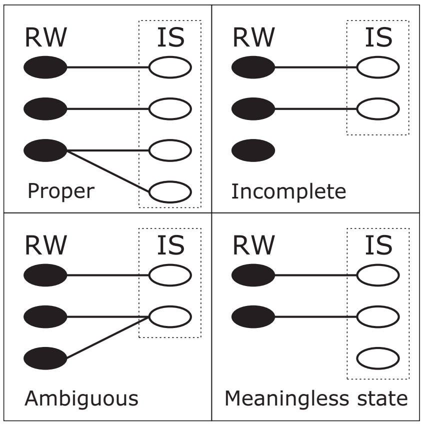


Figure 7.9: Proper and improper representations, according Wand and Wang (1996).

aspects such as coverage, resolution, repeatability, and frequencies. The *"quality of data depend on the design and production processes involved in generating the data"* (Wand and Wang, 1996). The processes might be complex; thus, a precise determination of the data quality may be difficult.

7.5.3 Evaluation of information quality in early development phases

In order to select the information basis the risk estimation will be based on, the application of a qualitative evaluation is suggested. Here, an exemplary evaluation is outlined that intents to take into account the 1) efforts that have to be spent in order to realize and implement specific information sources, 2) the reliability of the information sources, and 3) the estimate of the coverage of different fused sets of information channels describing the focused hazard actuation. For this purpose, exemplary qualitative measures are provided as evaluation basis comprising qualitative criterion transcriptions and related scores. The scores are subsequently applied to generate a preference measure for supporting the decision of selecting (a set of) information channels.

Availability of information channels

Due to the knowledge of the observer about the hazard actuation, it is one task to identify the relevant risk factors. But for some cases, they are neither necessarily available as information channel, nor simple to measure. Thus, one important aspect is to consider the efforts that are potentially spent in order to implement such information channel or fused sets of channels. In other cases, the required information is already available, or few lines of codes are necessary, in further cases, new sensors are required. The integration of new sensors into the hard- and software framework can be complex and time-consuming (depending on the current development stage of the system). In consequence, it is recommended to consider mentioned aspects (although they are not of primary interest for safety concerns). The exemplary effort evaluation criteria (ELE) are denoted in **Table 7.8**.

ELE Effort level estimation

Level	Description	Detail
3	Unreasonable	Notable efforts have to be spent in order to derive required information channel, possibly additional hardware has to be integrated
2	Huge	Further efforts have to spent but a software solution is feasible
1	Small	Minor computations or procedures are required to provide the required information channel
0	Available	Information channel is (quasi) available

Table 7.8: Exemplary qualitative evaluation of the efforts that have to be potentially spent to realize a specific information channel to express a desired risk factor.

RLE Reliability level estimation*

Level	Description	Detail
0	Low	Outliers often occur, precision is low, deficiency occurs
1	Moderate	Minor outliers, weak precision, but high availability
2	Good	Almost no outliers, high precision and availability
3	High	No outliers, accurate precision, no further unreliability

* *Reliability of information channels might depend on already considered instances.*

Table 7.9: Exemplary qualitative evaluation of the estimated reliability of a specific information channel representing a desired risk factor.

Reliability of information channels

Usually, the reliability if an information channel can be precisely determined, if the reliabilities of all steps of the processing chain are known. Nevertheless, the precise determination might be complex and time consuming, particularly, in early development phases. In this connection, a qualitative evaluation can provide a useful preliminary orientation as well. With the help of this evaluation it is rated which additional unreliability each information channel contributes to the hazard model. Data that are derived from prior, and already considered information processing steps, comprise no high additional unreliability. For instance, most of the given examples are based on an object recognition module. Unreliable elements in the process chain have to be considered due to the point that the whole assessment process is based on it. In this connection, it is important to be aware that the reliability of data that are derived from the object recognition process is already considered by taking into account the reliability of the object recognition module. The proposed reliability evaluation criteria (RLE) are denoted in **Table 7.9**.

The coverage of fused information channels

For many hazards, it is assumed that several risk factors have to be fused in order to sufficiently model the hazard actuation, and possibly, there are different variants for realizing this. In consequence, the different variants can cover different aspects of the causal relations. In this connection, it is assumed that very accurate risk models are less general and, hence, are potentially more vulnerable to inadequately perform when

CLE Coverage level estimation

Level	Description	Detail
0	Negligible	Risk factors (RFs) offer only specific and detailed information, the hazard actuation can not be fully covered
1	Very limited	Specific aspects of several relevant RFs are described, there might exist cases for which the hazard will not be described
2	Weak	The main aspects of the hazard actuation are described, but RFs appear to potentially generate high complexity
3	Strong	The RFs factor can sufficiently describe the occurrence of the hazard, despite it is not the exact description of the hazard actuation
4	Fully	The RF fully and precisely specifies the hazard actuation

Table 7.10: Exemplary qualitative evaluation of the estimated amount that an information channel or a set of them cover the considered hazard actuation.

applied in a similar context. In contrast, very general risk models may lack precision, however, they may denote the better choice in the conservative sense (see conservative approach, Section 7.2.4) concerning the debate of modeling a classification that gives preference to false positives over false negatives. The respective coverage evaluation criteria (CLE) are listed in **Table 7.10**.

With the help of proposed qualitative evaluation method, risk factors of a considered hazard actuation are classified with regard to effort, reliability, and coverage. From this estimation a conclusion can be drawn in order to systematically choose one of the possible options (if available) for modeling the hazard actuation. The scores that were derived from qualitative evaluations can be fused with the help of a utility function. Hence, meaningful weights can be assigned the different sets of risk factors. The resulting utility value indicates the preference for an option of modeling the hazard risk. As described above, the risk consists of the two factors likelihood and consequence. Thus, the risk factors can be distinguished in factors that effect the likelihood, and the consequences.

Example

The worksheet shown in **Table 7.11** illustrates an example considering the likelihood, **Table 7.12** the severity. Here, the utility function emphasizes reliability and coverage, while the required efforts are reflected as reduced utility value. Since the reliability is represented via qualitative measures, the reliability of a fused information channel is expressed by the minimum reliability value of its components in order to measure it with regard to the 'weakest link of the chain'. Usually, reliability values are multiplied for series systems of n components.

In the exemplary worksheet in **Table 7.11**, the applied utility function is denoted in the right lower corner, and respective results are denoted in the last column. Here, the hazard model considering plastic and heat being participated in the production of toxic vapor or a residential fire, are assumed to be described by assuming the heat source as a potential heat source (independently of its on/off-state), and considering their relative distance. In fact, this approach provokes many false alarms, especially, if the heat source can be turned off. However, it is not sure that the heat source could be turned on later on. Hence, it seems reasonable not to deposit plastic toys or other inflammable objects on the cold kitchen stove, wood-burning fireplace, or the like.

7.5.4 Determining the risk function

If the risk factors are known and the information channels to model the hazard actuation are chosen, the question arises how the risk factors are related to the hazard actuation, and how can risk models be realized that adequately describe the hazard actuation. Depending on the purpose this can be principally realized via utilization of mathematical functions, or algorithms. For instance, the relation of measurable risk factors (information channels) map a mathematical function to a risk value: Therefore, additionally generated information channels could be required and have to be abstracted, for instance, timer functions that measure an exposure time. However, as already partially influenced by the selection of the used information channels, the problem to define the description granularity arises: The hazard actuation can be modeled realistically, or the actuation is approximated with a function using an adequate degree of abstraction. This is affected again by aspects of reliability and (conservative) classification performance (see Section 7.2.4). In general, a risk function can be determined based on information channels (characteristics), and prespecified or learned parameters. Here, the subset of characteristic, $\mathbf{C}_\mathrm{q} \subseteq \mathbf{C}$, is used for determining the likelihood, and the subset $\mathbf{C}_\mathrm{r} \subseteq \mathbf{C}$ for describing the severity. Possibly required parameters $\boldsymbol{Z}$ can be prespecified or learned, which are the likelihood-related parameters, $\boldsymbol{Z_q} \subseteq \boldsymbol{Z}$, and the severity-related parameters, $\boldsymbol{Z_r} \subseteq \boldsymbol{Z}$. In consequence, the risk function,

$$risk_i = f_i^{acc}\left(\mathrm{C}_\mathrm{q}, \mathrm{Z}_\mathrm{q}\right) \cdot f_i^{sev}\left(\mathrm{C}_\mathrm{r}, \mathrm{Z}_\mathrm{r}\right), \tag{7.10}$$

can be determined.

The universality of this notion shall emphasize that, in principle, every kind of mathematical function or algorithm might be suitable to express the mapping from information channels and parameters to risk values. A risk value might be determined by neuronal nets, fuzzy functions, decision trees, support vector machines, functions similar to ‘danger indices’ in Section 2.4.1, or fault trees in Section 2.4.2, to mention a few. Hence, the mapping can be realized with known methods that are integrated within the structure of a Safety Principle. This implies flexibility with regard to the modeling of the risk function, and the refinement of Safety Principles during the operation phase of the system.

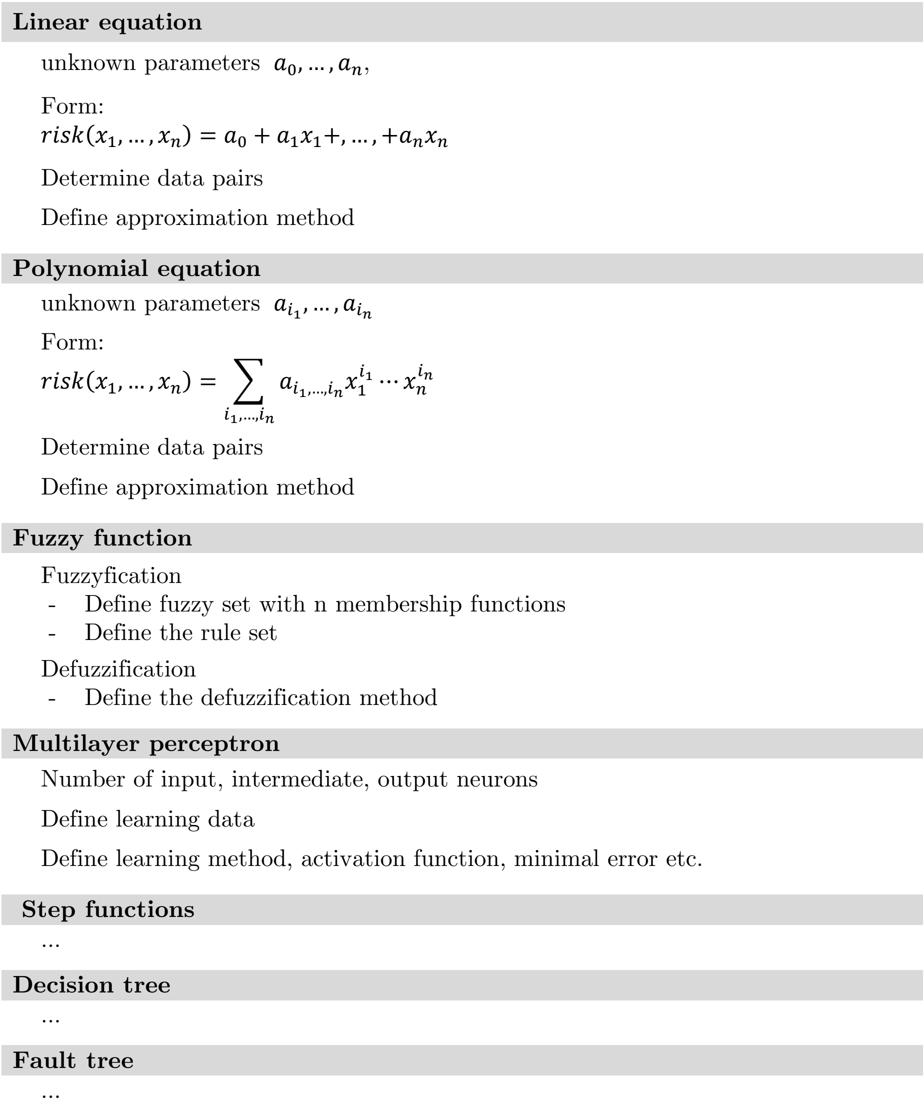
After defining and realizing the required information channels, the mapping from information channels to the desired risk values has to be defined, including the required parameters. A simple mapping function is a step function, for instance, if an input channel exceeds a certain threshold, or a linear function, mapping the linear correlation of the input channel as risk factor. More sophisticated functions are polynomial functions, exponential, logarithmic functions, neural networks, and the like. In the most cases, multiple factors may have effect on the risk amount. In consequence, multidimensional mapping functions are required. A list of available functions, respective parameters, and the procedure to determine the function parameters is shown in **Table 7.13**.

Hazard causation analysis			Risk factor data analysis for **likelihood**						
DSF #	Mishap, accident	Related objects or attributes	Related risk factors	Effort level est.	Risk level est.	#	RFs: Risk determination basis	Coverage level est.	***U***(ELE, RLE,CLE)
I.1	Vapor intoxication	**<attr:plastic>** <attr:polypropylene> <attr:thermoplast> <attr:heat_ source> **<attr:strong_heat _source>** <attr:potential_heat _ source>	- relative distance, - spatial relations (“close to”, “on”, “above”) - on/off-state of the cooking plate (via kitchen stove knobs) - heat indicator of the heat source (if available) - temperature of the heat source (if infrared temp. sensor on-board) - assumption of potential heat source - exposure time	1 1 2 2 3 1 1	3 1 1 1 3 3 2	1 2 3 4 5 6 7	1/6/7: Distance, exposure time, assumption of strong heat sources 1/6: Distance, assumption of strong heat sources 1/5/7: Distance, temperature sensor, exposure time 2/4/5: Spatial relations, state of the heat source, temperature sensor	3 3 4 3	11 14 11 5
I.2	Domestic fire	**<attr:plastic>** <attr:polypropylene> <attr:thermoplast> <attr:heat_ source> **<attr:strong_heat _source>** <attr:potential_heat _ source>	- relative distance, - spatial relations (“close to”, “on”, “above”) - on/off-state of the cooking plate (via kitchen stove knobs) - heat indicator of the heat source (if available) - temperature of the heat source (if infrared temp. sensor on-board) - assumption of potential heat source - exposure time	1 1 2 2 3 1 1	3 1 1 1 3 3 2	1 2 3 4 5 6 7	1/6/7: Distance, exposure time, assumption of strong heat sources 1/6: Distance, assumption of strong heat sources 1/5/7: Distance, temperature sensor, exposure time 2/4/5: Spatial relations, state of the heat source, temperature sensor	3 3 4 3	11 14 11 5
							$\boldsymbol{U}(\mathrm{ElE}_i, \mathrm{RLE}_j, \mathrm{CLE}) = 2 \cdot \mathrm{CLE} + 3 \cdot \min(\mathrm{RLE}_j) - \left(\sum_{i=1}^{n} \mathrm{ELE}_i - n + 1\right)$		

Table 7.11: Worksheet for determining likelihood-related risk factors and evaluation of respective information and the information quality.

Hazard causation analysis			Risk factor data analysis for **severity**						
DSF #	Mishap, accident	Related objects or attributes	Related risk factors RFs	Effort level est.	Risk level est.	#	RFs: Risk determination basis	Coverage level est.	***U***(ELE, RLE,CLE)
I.1	Vapor intoxication	**<attr:plastic>** <attr:polypropylene> <attr:thermoplast> <attr:heat_ source> **<attr:strong_heat_ source>** <attr:potential_heat_ source>	- Number of persons nearby - Current location - Intended use - Temperature of heat source - Mass of plastic object - ...	2 1 0 3 1	1 3 3 2 2	1 2 3 4 5	1: Measuring the number of persons 2: Determine current location (flat, canteen kitchen, frame house, etc.) 3: Intended use (in flat, in canteen kitchen, etc.) 3,4,5: Measuring the number of persons nearby, temperature of heat source, mass of plastic object ...	3 3 3 4	7 14 15 12
I.2	Domestic fire	**<attr:plastic>** <attr:polypropylene> <attr:thermoplast> <attr:heat_ source> **<attr:strong_heat_ source>** <attr:potential_heat_ source>	- Number of persons nearby - Current location - Intended use - Temperature of heat source - ...	2 1 0 3	1 3 3 2	1 2 3 4	1: Measuring the number of persons 2: Determine current location (flat, canteen kitchen, frame house, etc.) 3: Intended use (in flat, in canteen kitchen, etc.) 3,4: Measuring the number of persons nearby, temperature of heat source ...	3 3 3 4	7 14 15 12
							$\boldsymbol{U}(\mathrm{ElE_i}, \mathrm{RLE}_j, \mathrm{CLE}) = 2 \cdot \mathrm{CLE} + 3 \cdot \min(\mathrm{RLE}_j) - \left(\sum_{i=1}^{n} \mathrm{ELE}_i - n + 1\right)$		

Table 7.12: Worksheet for determining severity-related risk factors and evaluation of respective information and the information quality.



Linear equation

unknown parameters $a_0, \dots, a_n$,

Form:
$risk(x_1, \dots, x_n) = a_0 + a_1 x_1 +, \dots, + a_n x_n$

Determine data pairs

Define approximation method

Polynomial equation

unknown parameters $a_{i_1}, \dots, a_{i_n}$

Form:

$$risk(x_1, \dots, x_n) = \sum_{i_1,\dots,i_n} a_{i_1,\dots,i_n} x_1^{i_1} \cdots x_n^{i_n}$$

Determine data pairs

Define approximation method

Fuzzy function

Fuzzyfication
- Define fuzzy set with n membership functions
- Define the rule set

Defuzzification
- Define the defuzzification method

Multilayer perceptron

Number of input, intermediate, output neurons

Define learning data

Define learning method, activation function, minimal error etc.

Step functions

...

Decision tree

...

Fault tree

...

Table 7.13: Exemplary functions for realizing the computation part for estimating the likelihood and severity, and their typically required parameters.

Example

In the example shown in **Table 7.11**, interaction of heat and plastic shall be modeled with regard to the accident likelihood. According to the analysis, the modeling of the risk function on basis of one risk factor and one assumption is favored. On the one hand, the heat source is assumed to be potentially hot, disregarding that it can be turned off. On the other hand, the relative distance of the plastic object and the heat source is figured out as essential risk factor. For the given example, an one-dimensional function has to be modeled, which maps the distance between a plastic object and a heat source to a likelihood value.

In order to generate a mapping function, experts can be asked for the accident likelihood similar to the already mentioned evidence-based approach. An exemplary scenario, as illustrated in **Figure 7.10,** and the related questions the expert are asked, could be:

- Assume the cooking plate is turned on, and typical temperatures of the cooking plate are classified as strong heat sources.
- Assume further that the material attribute of the plastic object are not accurately known, thus, it could be made of thermoplastic.
- Estimate the likelihood of a residential fire (e.g. x of 1000 trials) under adverse conditions (more objects nearby, kitchen hood running, wooden kitchen units, and the like) according the minimal distance between the plastic object and the cooking plate.

In this regard, the expert's estimates can be based on former experiences or experiments.

The results that are retrieved by consulting experts can be interpolated, for instance, by a mathematical function. In **Table 7.14**, exemplary expert ratings are listed according to pre-defined distance categories, $\boldsymbol{D}$. In order to approximate a mapping function, the upper confidence level (UCL) can be computed. This is realized by utilizing the Student's t-distribution, which is $t_{(1-0.99;4)} = 3.747$ for statistically enclosing 99% of $n_D = 5$ probes. The variance $\mathrm{VAR}(\boldsymbol{D}) = \frac{1}{n_D-1}\sum_{i=1}^{n_D}\left(d_i - \overline{\boldsymbol{D}}\right)^2$ is calculated based on mean $\overline{\boldsymbol{D}} = \frac{1}{n_D}\sum_{i=1}^{n_D} d_i$. Then, each UCL can be computed applying respective mean and variance value according to

$$\mathrm{UCL}_D = \overline{\boldsymbol{D}} + t_{(1-\alpha;n_D-1)}\frac{\sqrt{\mathrm{VAR}(\boldsymbol{D})}}{\sqrt{n_D}}, \tag{7.11}$$

assuming normal distribution. For interpolation, a polynomial of fifth order fits well (the boundaries of the polynomial have to be correctly considered as well). In **Table 7.14**, exemplary expert estimates are denoted for pre-defined distance categories. In **Figure 7.11,** the risk values, their UCL, and the interpolation are shown. The denoted formula could be applied as accident likelihood function (for the interval $[0, 15cm]$).

In order to estimate the accident severity of a residential fire, it was decided to consider the intended use of the robot. For instance, for a robot that is designed for usage in a typical household, it has to be assumed that humans can come to death due to a residential fire. As initially mentioned, qualitative measures can be helpful for this purpose. For instance, the accident severity of a residential fire can be estimated with

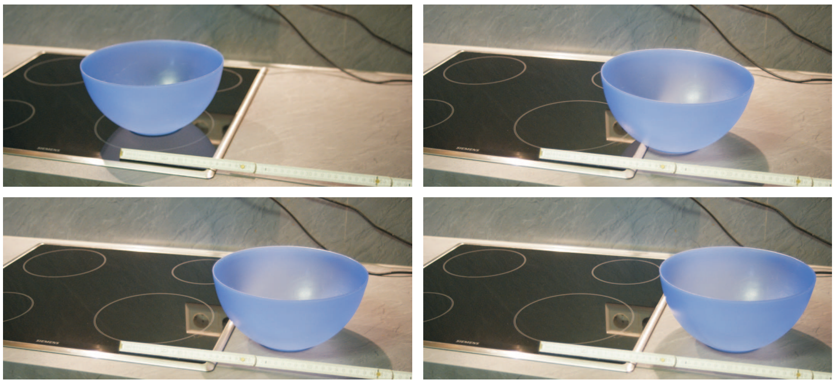


Figure 7.10: Scenario for estimating the risks of approaching a plastic bowl to the cooking plate.

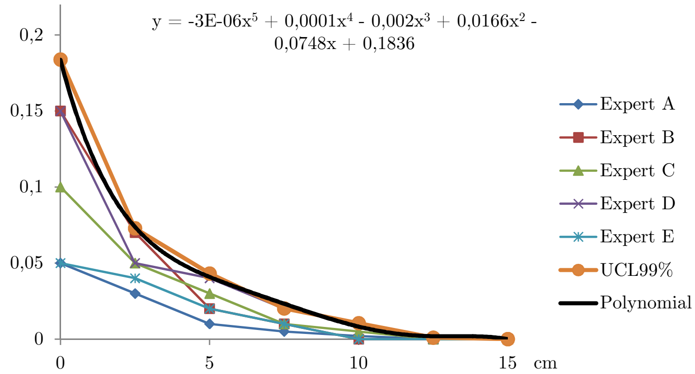


Figure 7.11: Graphs of the expert estimates, the UCL, and the polynomial interpolation, with x as distance [cm] and $y = f(x) = f_i^{acc}$.

the 'Hicks Scale' s(c.f. Proske, 2008). Hence, 'multiple fatalities and injuries' are rated with 100 ($severity_{Hicks}[0,100]$), and normalized ($severity_n \in [0,1]$) with $f_i^{sev} = 1$. The Safety Principle, $i = 1$, and the respective formula mapping the minimal distance d between the two objects into the risk value can be denoted as:[3]

$$\left(attr_{k,o} = \text{'attr:plastic'}\right) \wedge \left(attr_{l,p} = \text{'attr:strong\_heat\_source'}\right)$$

$$\rightarrow risk_1\left(d\right) = \begin{cases} 0.184 & : d < 0cm \\ \begin{matrix} -3{\cdot}10^{-6}d^5 + 0.0001d^4 - 0.002d^3 \\ +0.0166d^2 - 0.0748d + 0.1836 \end{matrix} & : 0 \leq d \leq 15cm \\ 0 & : d > 15cm \end{cases} . \qquad (7.12)$$

Finally, the process of determining a risk function is exemplary shown. Such formalized safety knowledge denotes the basis to enable the system itself to assess the situation risk. The question arises how the safety knowledge is integrated into and utilize within the risk assessment component and overall system in order to realize this. This is detailed in the subsequent section.

Likelihood estimation of a residential fire

Distance d [cm]	0	2,5	5	7,5	10	12,5	15
Expert A	0,05	0,03	0,01	0,005	0,002	0	0
Expert B	0,15	0,07	0,02	0,01	0	0	0
Expert C	0,1	0,05	0,03	0,01	0,005	0	0
Expert D	0,15	0,05	0,04	0,02	0,01	0,001	0
Expert E	0,05	0,04	0,02	0,01	0	0	0
Mean	0,1	0,048	0,024	0,011	0,0034	0,0002	0
Var	0,0025	0,00022	0,00013	0,00003	0,0000178	0,0000002	0
UCL99%	0,183785467	0,07285479	0,04310603	0,02017824	0,01046983	0,0009494	0

Table 7.14: Exemplary values of possible expert likelihood estimates for a residential fire in case of approaching plastic to heat sources. The values denote the expert's estimates of accident likelihoods according to different distances.

[3] However, possible intersection of the object areas is not explicitly considered. This could to be taken into account bygenerating a formula that expresses the risk in function of the area overlap or by a function for negative distances.

8 Realization of the Dynamic Risk Assessment Approach

8.1 Integrating Hazard Knowledge

Risks are focused arising through interaction of environment objects. The considered risks are assumed to be situational inherent and, therefore, in principle detectable by analyzing the inner structure of a situation. Relations are used in the SOM-technique to describe the inner structure of situations and in consequence the generation of another characteristic within a situation results. Hence, the construct 'relation' can be applied to derive abstract information from a situation. Within the relation, it is specified when it becomes applicable and how the information processing (on basis of which characteristics or additional information, e.g. knowledge bases) takes place. In consequence, the dynamic risk assessment concept can be expressed via a so-called 'risk assessment relation'. Since situations in the SOM-approach are considered as snapshots of a dynamic process, the situation itself is not necessarily dynamic, but the sequence of situations. Thus, assessing a sequence of situations via risk assessment relations can be defined as a realization of the proposed dynamic risk assessment approach. The general concept is shown in **Figure 8.1.**

Due to the SOM-based realization of the risk assessment, it is applicable to any instance of the SOM-based cognitive architecture. This allows the integration of the safety knowledge into the experience database, into the perception, action models, and into the anticipation process. Furthermore, the safety knowledge (Safety Principles) can be expressed using the SOM relation structure. In consequence, it is consistent with the overall SOM-based cognitive architecture, and hence, the information processing within the architecture can principally take place as well on the safety knowledge. This compatibility at the conceptual level allows in principle that the processing of the safety knowledge in terms of refinement can be realized.

The relation in SOM notation has the same structure as an operator, and is also called passive operator (internal causal relation between characteristics: In terms of 'because' (Söffker, 2001)). As mentioned in Section 4.1, a relation is applicable to a situation if its so-called explicit assumptions $\mathrm{eA_x}$ are fulfilled. Hence, each relation requires the presence of specific characteristics. The required characteristic can be seen as a condition for its application and as inputs of the relation function. On the basis of the inputs (and parameters as implicit assumptions $\mathrm{iA_x}$) the relation generates new (abstract) characteristics c, for example like a mathematical function, $\mathrm{c} = \mathrm{f}(\mathrm{eA_1} \ldots \mathrm{eA_i}, \mathrm{iA_1} \ldots \mathrm{iA_j})$. Hence, a conditional part of a Safety Principle automatically comes with the SOM approach. For instance, if the appearance of characteristic 'A' and 'B' indicates

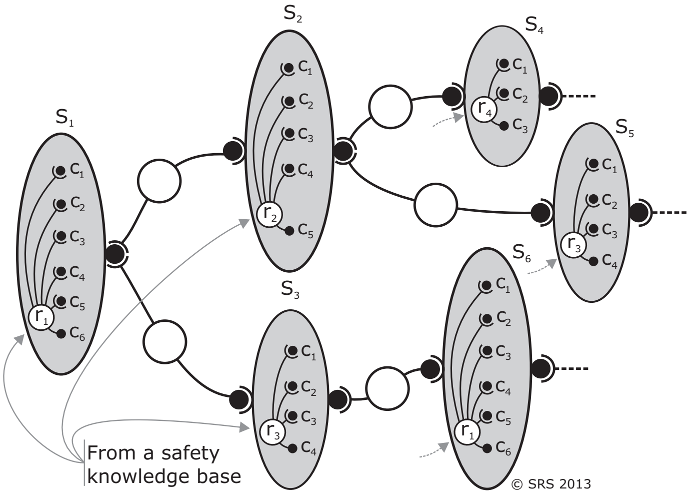


Figure 8.1: Realization of the dynamic risk assessment approach. Risk as a part of the situation: Situation risk-awareness.

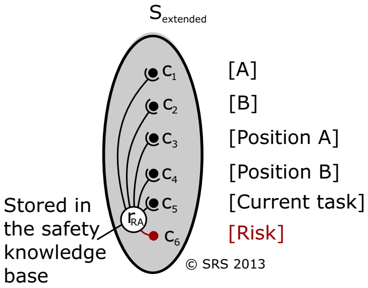


Figure 8.2: SOM-relation applied to denote the principle of the risk assessment process within a SOM-based approach.

risk 'Risk', then, the relation generating characteristic 'Risk' awaits the presence of characteristic 'A' and 'B'. The given example is illustrated in **Figure 8.2.**

Since the interaction of objects is particularly considered, some additional processing steps are required in order to establish the risk assessment. As already mentioned, the safety knowledge, expressed using Safety Principles is generalized in terms of referencing on object attributes instead on specific objects. Furthermore, the risk estimation can be based on already available or additional required situation characteristics. These dependencies are illustrated in **Figure 8.3.** Here, it is assumed that recognized and grasped objects are represented as kind of lists, describing the contained objects in a hierarchical structure, for instance, in XML, as it is proposed by Ertle et al. (2012a), or in a similar notation. On the one hand, there are data, describing currently measured aspects, such as position pose, size, and so forth. On the other hand, there are data that are inferred from a knowledge base containing declarative object knowledge, such as typical size, grasping data, center of mass, and further object attributes. In this connection, the object recognition (possibly in combination with a kind of object map in order to store and track already recognized objects) of the current environment provides the measured partition of the data, denoted in **Figure 8.3** as 'environment objects'. The grasping module in turn provides measured and stored data about the currently grasped

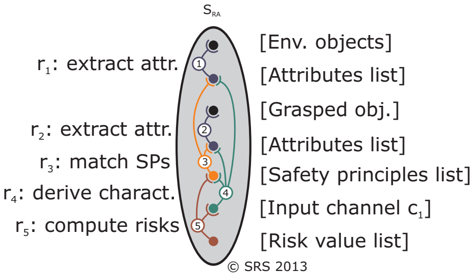


Figure 8.3: The situational object interaction risk assessment process denoted in detail in SOM notation.

objects, called ‘grasped objects’. Via a further processing step, object information is selectively inferred from a knowledge base. For considering safety aspects, in particular, these are the object attributes. This is realized with a relation $r_{1,2}$ that ‘extracts’ the object attributes object-wise in the ‘attributes list’. A further relation r_3 generates a list of Safety Principles, that become applicable due to the presence of respective object or object attributes. This relation could also be called the ‘hazard identification relation’.

It is specified within the Safety Principles which information channels (characteristics) are required in order to base the computation of the risk values on them. Hence, from the list of applicable Safety Principles, the list of required additional characteristics is known. In consequence, the additionally required characteristics can be generated by applying the specified relations, for instance, the relation r_4. In the following, the list of applicable Safety Principles is required to compute the risk function r_5 (‘specific risk assessment relation’) that is defined in the second part of each Safety Principle. In consequence, a risk value results from each specific risk assessment relation, and is added to the characteristic ‘risk value list’. In a further step, a resulting risk value can be computed in order to express the overall risk of a situation (not illustrated).

From the outlined process, the requirements for related subprocesses become visible. On the one hand, the requirement on the object recognition and a dynamic object environment map can be derived. On the other hand, the principle processing steps are explained for realizing the dynamic risk assessment approach. Both together provide a first basis for an implementation.

8.1.1 Risk-sensitive planning and perception

For ensuring safety of autonomous systems, the capability of the system to have situation awareness is important (Wardziński, 2006). As expressed by Endsley, situation awareness is *“the perception of the elements in the environment [...], and the projection of their status in the near future”* (Endsley, 1995). Consequently, two functions play a major role: Planning and perception. Unfortunately, neither their relation nor their internal relation with respect to internal structuring is detailed.

The first important function, the planning capability, is generally speaking the searching for a sequence of action in order to reach a given goal. Within the applied cognitive

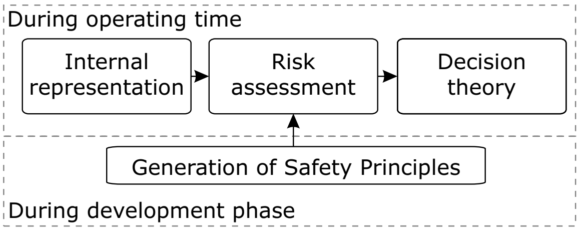


Figure 8.4: The dynamic risk assessment process based on initial safety knowledge (cf. Ertle et al., 2010c).

architecture, planning is realized with the help of its anticipation capabilities, see Section 4.2.4. Accordingly, the outcomes of applicable operators on a starting situation are predicted in a recursive fashion, so that a net of situation-operator-situation chains is generated, which represents the space of different (known) sequences of action, the so-called mental action space. If this action space contains the goal situation (once or multiple times), a plan for solving the problem can be found by successively transferring the system from the current to the desired goal situation. In this connection, the selection of the sequence of action should clearly depend on the risks that may be comprised in the different situations. Therefore, thresholds with regard to the benefit of the task have to be maintained, and an optimization should take place as well under consideration of risks.

The perception is seen as a second important component because the information selection is strongly related to it. In dependence on the current parameters, knowledge, intention, goal, and the like, the information processing of sensory data is adapted in order to provide the system with currently relevant information aspects. For considering safety, it must be ensured that potential risks in a situation are perceived and classified as relevant and required aspects to be considered, and hence, remain in the scope of the system's attention. Otherwise, risk information could be discarded as non-problem related information.

Since the main objective of integrating risk assessment capabilities in robotic systems is to realize safe autonomy, deliberative decision-making capabilities are required, at least at a higher systemic level. *"Rational decision-making requires, therefore, a clear and quantitative way of expressing risk so that it can be properly weighted, along with all other costs and benefits, in the decision process"* (Kaplan and Garrick, 1981). The term 'properly weighted' in relation to risks, costs and benefits requires comparable measures.

If the system has an internal representation of the sets of possible sequences of action available, for instance, in form of the mentioned action space, and this representation is equipped with quantitative measures such as benefits and costs, the decision theory becomes applicable in order to generate desired plans. The required quantitative measures for taking into account potential risks are provided by the dynamic risk assessment approach. Its principle integration is illustrated in **Figure 8.4.**

In order to realize the risk-sensitive planning approach, the mental action space is assessed situation per situation, and risk values are attached as further situation characteristics before the decision process takes place. In this connection, risk values can be assumed to be transition (operator) costs. As the action space is a directed graph,

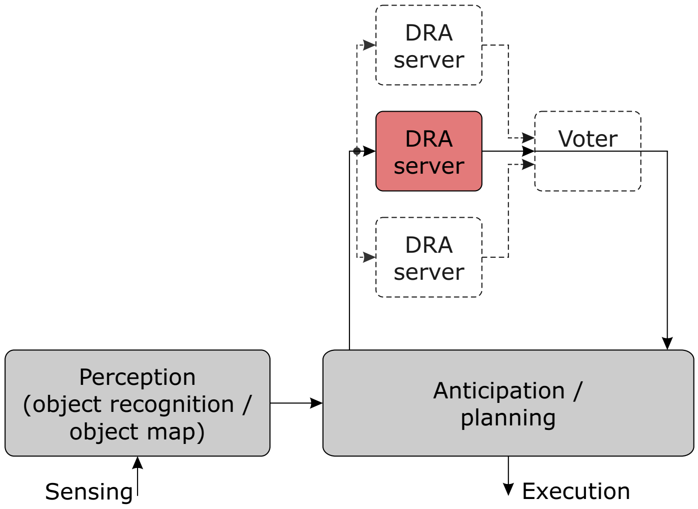


Figure 8.5: Sketch of the communication within the dynamic risk assessment (DRA) architecture, adopted from Ertle et al. (2012a).

and consequently, is equipped with respective weighted transitions, various methods are available in order to solve the optimization problem of weighted directed graphs.

8.1.2 The Risk assessment server

The latter section outlined the general steps for generating a knowledge base for object-object interaction hazards. In this section it is described more specifically how the safety knowledge is applied within the robotic system in order to put it into effect. Therefore, a client-server communication structure is chosen, since it provides some advantages, such as system interoperability, reduced workload, data integrity and the like (Yadava and Singh, 2009). So, the risk assessment module is realized as server, and the components of the robot control system as clients. Furthermore, a server-based architecture simplifies the realization of concepts such as distributed redundancy and software diversity (e.g. various compilers). The architecture highlighting the general integration of a dynamic risk assessment server into robotic architectures is sketched in **Figure 8.5.** Here, the perception includes the recognition of objects from the current scene. As object recognition processes are time-consuming, it is assumed that the presence of objects can be stored within, and in particular, recalled from a kind of object map. Activities for generation of the actions space, and the planning process are assumed to take place within the 'Anticipation/planning' module.

The dynamic risk assessment server module is connected to the anticipation/planning module, and might be realized as multiple instances, built via various compilers, or is executed on multiple redundant devices. A subsequent voter receives redundant responses from each dynamic risk assessment server instance, allowing detecting deviations due to software errors or failures in general. According to the dynamic risk assessment approach, the anticipation/planning module passes the set of possible situations to the dynamic risk assessment server(s), which in turn extends each of them with risk information in the described manner.

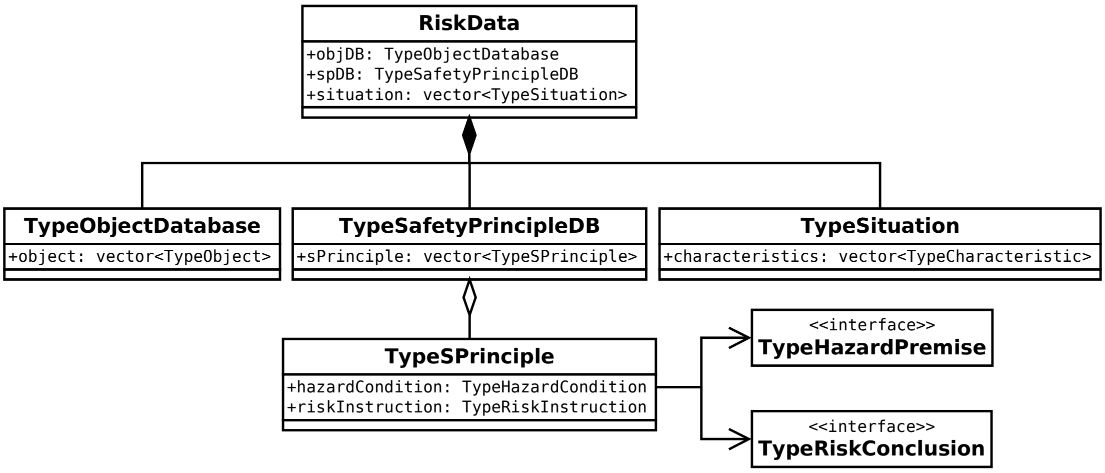

RiskData
+objDB: TypeObjectDatabase
+spDB: TypeSafetyPrincipleDB
+situation: vector<TypeSituation>
TypeObjectDatabase
+object: vector<TypeObject>
TypeSafetyPrincipleDB
+sPrinciple: vector<TypeSPrinciple>
TypeSituation
+characteristics: vector<TypeCharacteristic>
TypeSPrinciple
+hazardCondition: TypeHazardCondition
+riskInstruction: TypeRiskInstruction
<<interface>>
TypeHazardPremise
<<interface>>
TypeRiskConclusion

Figure 8.6: UML extract of the dynamic risk assessment server structure (Ertle et al., 2012a).

In general, the concept in this work favors the utilization of knowledge-based approaches. Besides the advantage to maintain knowledge-based updates, the separation of knowledge from the implementation is enforced (Beierle and Kern-Isberner, 2006). However, it is has to be considered that a part of the knowledge might be required for non-safety-related purposes and, therefore, must be accessible for other components of the system as well. For instance, this might be the case for the object knowledge base. In consequence, the object knowledge base and safety knowledge base are realized as a separate knowledge bases for this reason, and in addition, for providing better overview and maintainability. The utilization and integration of the safety and object knowledge for the dynamic risk assessment process is outlined in UML notation in **Figure 8.6.**

Here, the information of object attributes is inferred from the object knowledge base. The Safety Principles are obtained from the Safety Principle knowledge base. The dyadic form of the Safety Principles, the premise and the conclusion are implemented via interface structures. Interfaces might be comfortable to implement but, so far, they constitute a breach with the paradigm to separate the knowledge from the code implementation.

8.1.3 Safety clearance to critical situations

In the latter section, it is described how risks can be identified and quantified within a dynamic risk assessment approach and how this is integrated within the SOM approach. For realizing that a system safely operates, courses of action have to take place such that situations with unacceptable risks are sufficiently and reliably avoided. Additionally, the risk could be reduced, if the ‘distance’ to hazardous situations is maximized. This can be realized if the possible successor situations can be determined, on the one hand, and on the other hand, a measure for the distance to hazardous situations is defined. The anticipation capabilities of a cognitive technical system, described in Section 4.1, and the integration of the dynamic risk assessment approach in Section 8.1 denote the fulfillment of the first requirement. The realization of a safety clearance is described in the sequel.

Since it is difficult to formulate a distance measure at the description level of actions (operators) and situations, in form of spatial or temporal distances, the number of actions that are required to be performed to arrive at a hazardous situation are utilized as a distance measure. This can be realized by 'diffusing' the risk value of a hazardous situation to its prior situations. If a decision process is designed to find courses of action with minimized risks, an additional risk aversion can be realized with this approach. From this, it follows that a course of action with a safety clearance to hazardous situations can be favored over one that closely passes hazardous situations. Hence, the safety clearance depends on the number of operators that have to be executed until a hazardous situation is reached.

In order to realize this, a Markov transition matrix can be applied. In general, it can be used to compute the reachability of future states of a discrete Markov chain. The action space, representing the anticipation capabilities of a cognitive technical system (see Section 4.2.4), contains the current and all reachable future situations (depending on the amount or depth of the anticipation). Future situations are reachable by executing applicable operators (transitions), and each situation typically has one or a set of applicable operators. The applicability of an operator does only depend on its respective initial situation. Hence, the action space has Markov properties.

The Markov transition matrix is generated without considering any preference for action selection. Hence, the available successor situations are equally probable, consequently, their probability is inverse proportional to the available applicable operators. For instance, if there are '4' applicable operators in a situation, each has chance to be randomly chosen with a probability of '0.25'. The assumption of equally probable successor situations denotes the modeling of a task neutral reachability of future situations on the one hand. On the other hand, it represents the random action selection of an exploring agent (without any prior knowledge). Exploration is essential for learning, as a *"[...] learning agent must prefer actions that it has tried in the past and found to be effective in producing reward. But to discover such actions, it has to try actions that it has not selected before. The agent has to exploit what it already knows in order to obtain reward, but it also has to explore in order to make better action selections in the future. The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task"* (Sutton and Barto, 1998).

The required transition matrix has the dimension $n_{sit} \times n_{sit}$ for an action space consisting of n_{sit} situations. If, for instance, in the situation, $\mathrm{s_x}$ with $x \in \{1 \dots n_{sit}\}$, one operator is available, which transfers the situation, $\mathrm{s_x}$, to any situation, $\mathrm{s_y}$ with $y \in \{1 \dots n_{sit}\}$, of the action space, the transition probability of '1' is denoted at the respective position, x, y, in the transition matrix. In order to transform the complete action space into the transition matrix the inverse number of available operators (transition between two situations) in each situation, $\mathrm{s_x}$, is denoted at each position, x, y, in the matrix in which the application of respective operator transfers the situation, $\mathrm{s_x}$, into respective successor situation, $\mathrm{s_y}$. Assuming, that the situation, $\mathrm{s_x}$, is the initial situation, and the situation, $\mathrm{s_y}$, is the successor situation, the operator $\mathrm{o_{x,y}}$ transfers $\mathrm{s_x}$ into $\mathrm{s_y}$. Furthermore, the transition probability, $P(\mathrm{o_{x,y}})$, is the reciprocal of the number of available operators of each initial situation, $\mathrm{s_x}$. From this, it can be derived the quadratic transition matrix,

$$\boldsymbol{M} = \begin{bmatrix} P(\mathrm{o}_{1,1}) & \dots & P(\mathrm{o}_{\mathrm{x},1}) \\ \vdots & \ddots & \vdots \\ P(\mathrm{o}_{1,\mathrm{y}}) & \dots & P(\mathrm{o}_{\mathrm{x},\mathrm{y}}) \end{bmatrix}. \tag{8.1}$$

By raising the transition matrix $\boldsymbol{M}$ to the power of n, the transition probabilities of applying a number n of operators can be computed. Furthermore, each situation $\mathrm{s_y}$ comprises a number of risk values $risk_{k,\mathrm{y}}$ with $k = \{1 \dots n_{scenario}\}$ (from assessment of applicable out of $n_{scenario}$ Safety Principles, not applicable Safety Principles are denoted as $risk_{k,y} = 0$), therefore, it can be generated the $n_{sit} \times n_{scenario}$ risk matrix

$$\boldsymbol{RISK}_{SP} = \begin{bmatrix} risk_{1,1} & \dots & risk_{1,y} \\ \vdots & \ddots & \vdots \\ risk_{k,1} & \dots & risk_{k,y} \end{bmatrix}. \tag{8.2}$$

Assuming summable risks values, the cross product of transition probability matrix $\boldsymbol{M}^n$ and the transposed risk matrix $\boldsymbol{RISK}_{SP}^T$ generates the situation risk matrix

$$\boldsymbol{MRISK}_n = \boldsymbol{M}^n \times \boldsymbol{RISK}_{SP}^T, \tag{8.3}$$

for taking a number n of arbitrary applied but applicable operators into account. With regard to planning actions, the $n_{sit} \times n_{scenario}$ situation risk matrix $\boldsymbol{MRISK}_n$ can be used as a look-up table to find out which risks are available in a specific situation, including the risks that are diffused from its reachable neighbored situations. Depending on which number n of operators are considered in the risk matrix $\boldsymbol{MRISK}_n$, different safety clearances can be realized. The safety clearance can be seen as a measure how far the foresight to hazardous situations should be. The higher the safety clearance n the wider hazardous situations can be 'circumnavigated' (risk aversion). For instance, if a safety clearance $n = 1$ is considered in a situation $\mathrm{s_2}$ with three applicable operators, and one of them is leading to the 'lethal' situation $\mathrm{s_3}$ (severity $S_{Acc} = 1$), the transition from the situation $\mathrm{s_1}$ to the successor situation $\mathrm{s_2}$ will result to the diffused risk $mrisk_{1,1,2} = 33\%$ (see **Figure 8.7** left). If a safety clearance of $n = 2$ is considered, the transition from the initial situation $\mathrm{s_0}$ to the successor situation $\mathrm{s_1}$ already comprises the risk $mrisk_{2,0,1} = 0.33 \cdot 0.33 \approx 11\%$ (see **Figure 8.7** right). From this results that in situations with no risk (e.g. $mrisk_{n,x,y} = 0\%$) a number n of arbitrary operators could be applied without reaching a hazardous/lethal situation. In other cases, the application of a number n, of operators after transferring from one situation $\mathrm{s_x}$ to another situation $\mathrm{s_y}$, poses the $mrisk_{n,x,y}$.

Thus, information describing the distance to hazardous situations allows, at first, the reduction of the set of situations, which should be reachable by the system, and secondly, selecting operators considering a safety clearance.

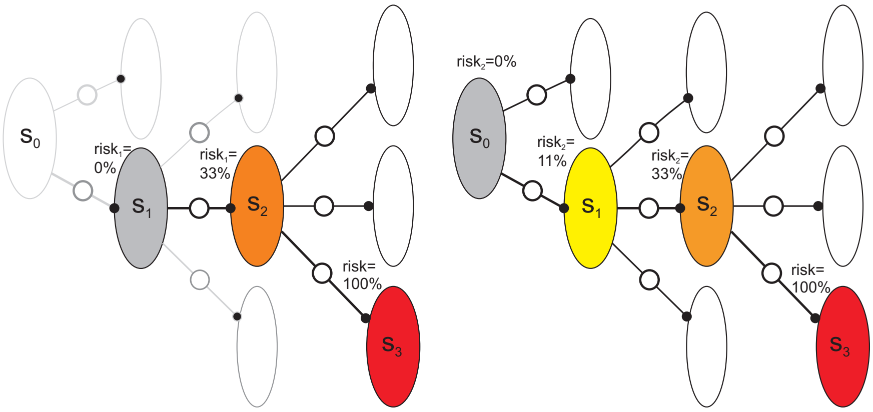


Figure 8.7: Safety clearance delivers risk value for risks with a distance of one (left part) or two (right part) operators. The 'future' risky situation s_3 can already be taken into account in s_2 or s_1 (Ertle et al., 2010a).

8.2 Simulation Experiments

8.2.1 Risk assessment server experiment

A small grid-world example is shown in **Figure 8.8** containing several potential hazardous object interactions. In all situations, the robot has gripped a 'coffee bowl' and is following a predefined path, so far not paying attention to any risks. A 'human' enters the scene drying hairs with a 'hairdryer'. The 'hairdryer' is deposited afterwards and the human crosses the scene. On the robot's path it approaches first a 'power plug', 'cooking plate' and later on the deposited 'hairdryer'. In general, the scene consists of ten situations that are passed consecutively to the dynamic risk assessment server.

The dynamic risk assessment server is realized within a SOAP[1] middleware offering a comfortable possibility to serialize and deserialize XML structures. Hence, the situational descriptions and the knowledge are realized via XML. The dynamic risk assessment server is prepared with initial object knowledge. The attribute-based generalization is realized by defining object attributes according to **Table 8.1**.

At first, the initial safety knowledge has to be loaded into the dynamic risk assessment server. This Safety-Principle knowledge base, shown in **Table 8.2**, contains four Safety Principles relating objects or their attributes to hazards as premises for the presence of respective hazards. The risk of each hazard is computed by the risk determination part. For illustration, all risks are interpolated as linear functions in dependence of respective object distances. The interpolation takes place between two extremes: Being a very hazardous constellation on the one hand, and on the other hand, being a constellation assumed to be safe. The last two columns of the table depict the two coefficients of a linear equation. As mentioned already, any functions or instructions can be utilized in this context.

[1] `http://www.cs.fsu.edu/~engelen/soap.html` [Online; accessed 08-December-2011]

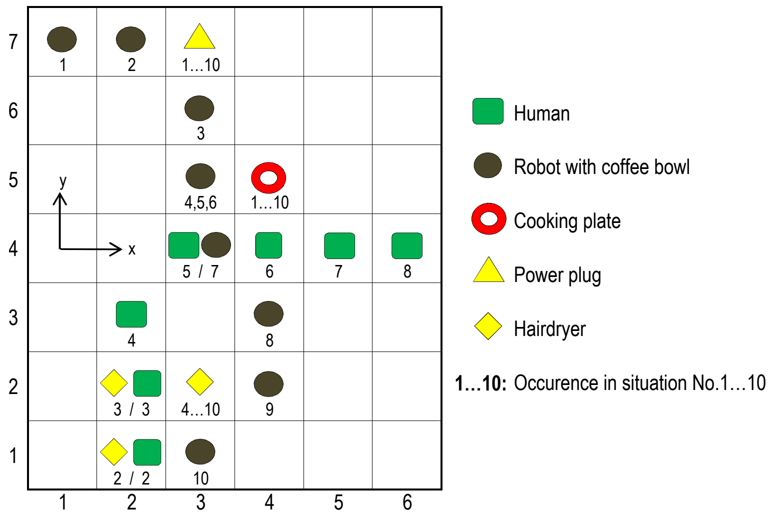


Figure 8.8: The Safety Principle simulator: Simulation comprising different hazardously interacting objects. The experiment is a 6*x*7 grid world and consists of 10 sequenced situations. Numbers below the symbols indicate respective object position at each sequence step (Ertle et al., 2012a).

Object knowledge base

Objects	Attributes
Coffee bowl	hot liquid, liquid, plastic
Human	obstacle
Cooking plate	obstacle, extreme heat
Power plug	obstacle, high voltage, grippable
Hairdryer	obstacle, high voltage

Table 8.1: The object knowledge base of the risk assessment server containing the attributes of the respective objects (Ertle et al., 2012a).

Safety knowledge base

Modeled risks	Gripped obj.	Scene obj.	Hazardous		Uncritical		Function type	Res. param.	
			Dist.	Risk	Dist.	Risk		a	b
Electric shock	liquid	high voltage	1	0,6	3	0	<linear>	-0,3	0,9
Scaling human	hot liquid	human	0	0,3	2	0	<linear>	-0,15	0,3
Collision risks	<any>	obstacle	0	0,1	2	0	<linear>	-0,05	0,1
Melting plastic	plastic	extreme heat	0	0,8	2	0	<linear>	-0,4	0,8

Table 8.2: Example of Safety Principles as risk computation models for object interaction accident (Ertle et al., 2012a).

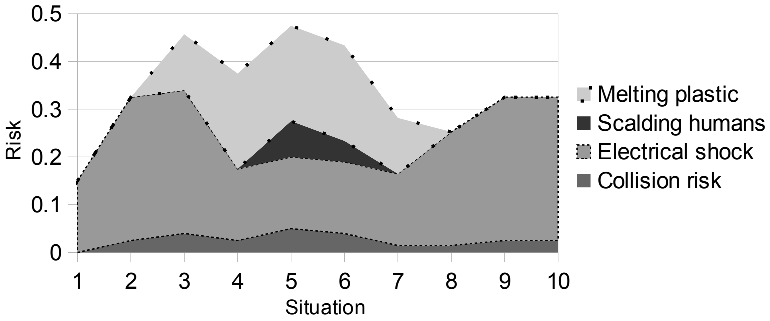


Figure 8.9: Graphical illustration of the output of the dynamic risk assessment: The risks appearing in the simulation example (Ertle et al., 2012a).

The results of the dynamic risk assessment server are illustrated in **Figure 8.9.** Risks are plotted situation-wise in a stack chart. The dynamic risk assessment server responds single risk values. The sum of these results in the overall situational risk value. Hence, it becomes apparent that the assessment of anticipated situations allows for balancing of single risks, overall risks with regard to comparisons of risks, risk thresholds or even modeled task benefits.

The values chosen for risk models are not realistic but for real world purposes they could be systematically determined with the help of the procedural model in the previous chapter.

8.2.2 Risk-sensitive planning experiment

As an example application, an arcade game[2] is chosen, in which an autonomous agent as softbot interacts within a grid-based environment. This application example is well suited in order to illustrate the proposed approach since it offers a simplified world focusing on the problems aimed in this contribution. Moreover, the developed functions can be easily transferred to real world applications since uncertainty, ambiguousness, faults, and perceptual limits can also be simulated by the arcade game in the figurative sense.

The environment consists of different kinds of fields and the agent can perform the actions ‘up’, ‘down’, ‘left’, and ‘right’. In general, the task consists of first picking up a certain number of ‘emeralds’, by entering related fields, and then finishing the level by leaving the scenario through an exit door, see **Figure 8.10.** Here, the agent is performing actions with help of the proposed cognitive architecture. Any situation s_i, as the input of the architecture, consists of the characteristics ‘x-position’ (integer), ‘y-position’ (integer), ‘type of the current field’ (string), and ‘collected points’ (integer). This example is now used to illustrate the action planning including safety aspects based on the mental action space. The agent has to leave the level by reaching the exit door in the lower right corner, which can be reached by following different paths. Furthermore, an emerald can be picked up to increase the number of collected points. This is no

[2] A. Entertainment, ‘Rocks’n’Diamonds’, `http://www.artsoft.org/rocksndiamonds/` [Online; accessed 08-Apil-2013]

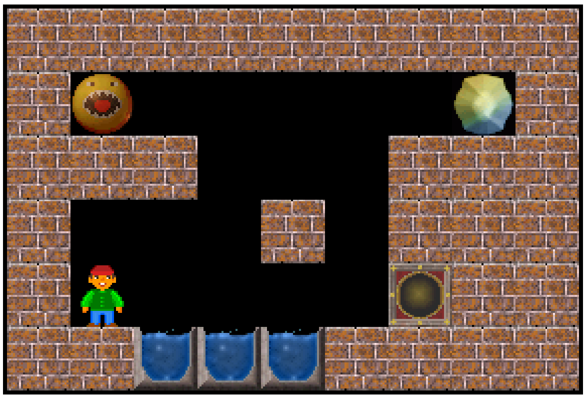


Figure 8.10: The simulation example - the arcade game (Ertle et al., 2010a).

necessary condition to finish the scenario, however, it can be used as an additional factor influencing the planning process. The considered scenario also contains three acid fields and a hostile monster agent performing horizontal movements. A collision with either the monster or the acid fields would lead to the undesired 'death' of the agent. In the considered application example a Safety Principle P_1 is defined.

$$SP_1 : Pos\,(player) = Pos\,(mortal\_field) \rightarrow risk = 1 \tag{8.4}$$

$$A : mortal\_field = \{acid\_field, monster\} \tag{8.5}$$

The conditional parts become true, if the position of the player is equal to the position of a '*mortal_field*' (which again could contain 'attributes A': '*acid_field*' or '*monster*'). The related risk is defined as 1 (accident severity $S_{Acc} = 1$ means 'death' of agent, accident probability $P_{Acc} = 1$ when the condition of Safety Principle is fulfilled and $risk = S_{Acc} \cdot P_{Acc}$).

The mental action space is generated based on the initial situation of the agent (lower left corner) and contains all possible future situations and actions. In **Figure 8.11,** the complete mental actions space as it results in this first simplified example is depicted using the SOM symbolic and colors in accordance to their present risk. Here, the hostile monster and the emerald in the upper right corner are not considered for simplification. After the generation of the mental action space, two different safety margins are evaluated. These can be identified by the indexes $risk_1, risk_2$. The agent's 'death' is assumed to have accident severity of '1' and therefore a risk 100% ('mortal'). Situations without any risks are described with 0% risk.

According to **Figure 8.11,** the shortest route will lead over situation $\mathrm{s}_{3,4,5,8}$ to the goal situation s_{21}. This path is hazardous because it is close to the acid fields ($\mathrm{s}_{9,10,11}$: colored red). Each decision error or exploration step will lead to the agent's 'death' with high probability. For planning the Dijkstra algorithm is used. It generates paths considering its costs. Thus, each transition from $\mathrm{s_i} \rightarrow \mathrm{s_j}$ costs

$$cost_{ij} = 1 + risk\left(\mathrm{s_j}\right),$$

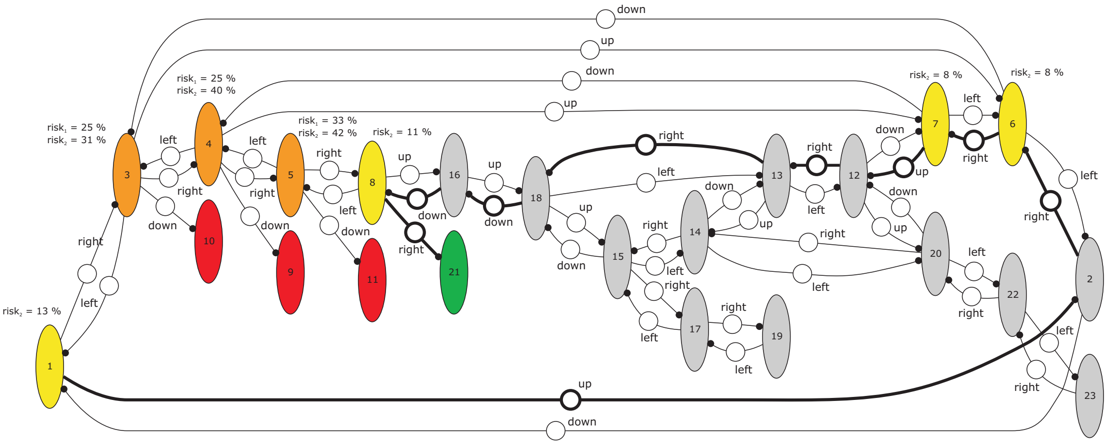


Figure 8.11: Graphical representation of the mental action space without considering the monster (Ertle et al., 2010a).

whereby the risk value is assumed to be expressed in percent $(0\dots100\%)$. The overall costs of the path $\vec{\mathrm{s}}$ with the elements $\mathrm{s}^0\dots\mathrm{s}^\mathrm{n}$ or n actions (with $n\in\mathbb{N}^*$) are calculated as

$$cost_{0,n} = n + \sum_{x=1}^{n} risk\,(\mathrm{s_x}).$$

This example cost formula is a very safety oriented strategy. The plan with lowest costs results to $costs(\vec{\mathrm{s}}:\mathrm{s}_{2,6,7,12,13,18,16,8,21})$. The goal is reachable without any risks, when considering a safety margin of one action. The plan does not change by considering safety margin of two actions but the maximum risk rises to $risk_2 = 11.1\%$ and the costs amount 37 $(\approx 11.1(\%)+8.3(\%)+8.3(\%)+9(steps))$.

By regarding the scenario (**Figure 8.10**), it can be seen that the protrusion directly above the player's starting position causes a bottleneck. Consequently, it is not possible to pass the acid containers with a greater safety margin.

When the dynamic elements (monster, emerald) are taken into account as well, the complexity is increasing significantly. The action space contains 816 situations (instead of 23), 1662 operators, and 22 different goal constellations. If the scenario is evaluated again with safety margin '1' then the goal can be reached without any risks. For shortest route 18 steps and 30 steps for the case that the emerald is collected are used. The planning process considering a safety margin '2' and collecting the emerald, results to

Path 11: $maxrisk_2(383) = 37{,}5\%$, $n = 42$ steps, $costs = 111$, $emeralds = 1$:
$o_{m:right}$, $o_{m:right}$, $o_{m:right}$, $o_{m:right}$, $o_{m:right}$, $o_{m:left}$, $o_{a:up}$, $o_{m:left}$, $o_{a:right}$, $o_{m:left}$, $o_{a:right}$, $o_{m:left}$, $o_{a:up}$, $o_{m:left}$, $o_{a:right}$, $o_{m:right}$, $o_{a:right}$, $o_{m:right}$, $o_{a:down}$, $o_{m:right}$, $o_{m:right}$, $o_{m:right}$, $o_{m:left}$, $o_{a:up}$, $o_{m:left}$, $o_{a:up}$, $o_{m:left}$, $o_{a:right}$, $o_{m:left}$, $o_{a:right}$, $o_{m:left}$, $o_{a:left}$, $o_{m:right}$, $o_{a:left}$, $o_{m:right}(38\%)$, $o_{a:down}$, $o_{m:right}(31\%)$, $o_{a:down}$, $o_{m:right}$, $o_{a:down}$, $o_{m:right}$, $o_{a:right}$.

When the emerald should be collected, then costs of 42 actions and maximum risk of 37.5% or overall costs of 111 have to be accepted. When the actions sequence is regarded, it can be seen, how the movement of the monster is taken into account to avoid collisions. The autonomous agent waits until the monster is in the left corner. This moment is used to pass the rock in the center of the scenario. If the monster again is in the left corner, the moment is used to collect the emerald. Performing this game without collecting the emerald is possible without any risks and with the cost of 22 (plan is not shown). Therefore, if the collecting of the emerald leads to a benefit of at least $111 - 22 = 89$ cost points then the 'emerald collecting' plan could be preferred. In **Figure 8.12** the respective action space is illustrated. The graph visualization software *Tulip*[3] is used to generate the graph. The *Graph Embedder* (GEM-Frick) was utilized to arrange the situations. Green-colored situations denote possible goal situations, red-colored situations denote lethal situations. Yellow, and orange-colored situations are risky, and grey-colored situations are without risk. The red-outlined situations indicate that the emerald was collected, in situations with black-outline, the agent did not pick up the emerald. The situations without emerald are located in the upper part of the graph, while the situations for which the emerald was collected are in the lower part. The connection of the both parts is visible as 'bottleneck' between the both parts. There apparently are few degrees of freedom with regard to the timing, denoting the problem to fetch the emerald.

In general, it appears that even a small problem with dynamic elements can result in a complex action space. Consequently, partial planning approaches might be required. However, since the risk assessment approach is not the reason for the complexity, it should be suitable as well for other planning techniques.

8.2.3 Interactive object manipulation simulation

For the demonstration of the concept a small simulation environment is implemented (Ertle et al., 2010c). The simulator generates a graphical environment in order to investigate the results of the described risk assessment module. Therefore, a scene in form of 2D-world is made available, containing a robot and several environment objects. The user can control the robot in order to manipulate the objects of the scene. The simulated scene is assumed to be the possible result of a powerful object recognition module. In this connection, it is assumed that the object recognition formulates different hypotheses for the identity of each object. The different hypotheses are assumed to be represented assigning different probabilities to different possible identities of the objects. The different hypotheses are specified for each object and remain fixed during the simulation, but they can be manually changed by the user. For the object identifiers, expressions in natural language are used. Further relevant information from the object recognition module, for instance, position, size of objects, and information about the internal state of the robot, for instance, its position and speed are collected in the situation description. The situation description serves as information basis for the risk assessment.

In order to realize the risk assessment approach, object and safety knowledge is required. Therefore, object information is declared in the object knowledge base. Here, one or more object attributes are assigned to each object in form of natural language description. The safety knowledge is represented in form of Safety Principles, consisting of the premise and conclusion part (see Section 7.5). The premise indicates the applicability of

[3] Tulip data visualization software `http://tulip.labri.fr/TulipDrupal/` [online; accessed 09. April 2013]

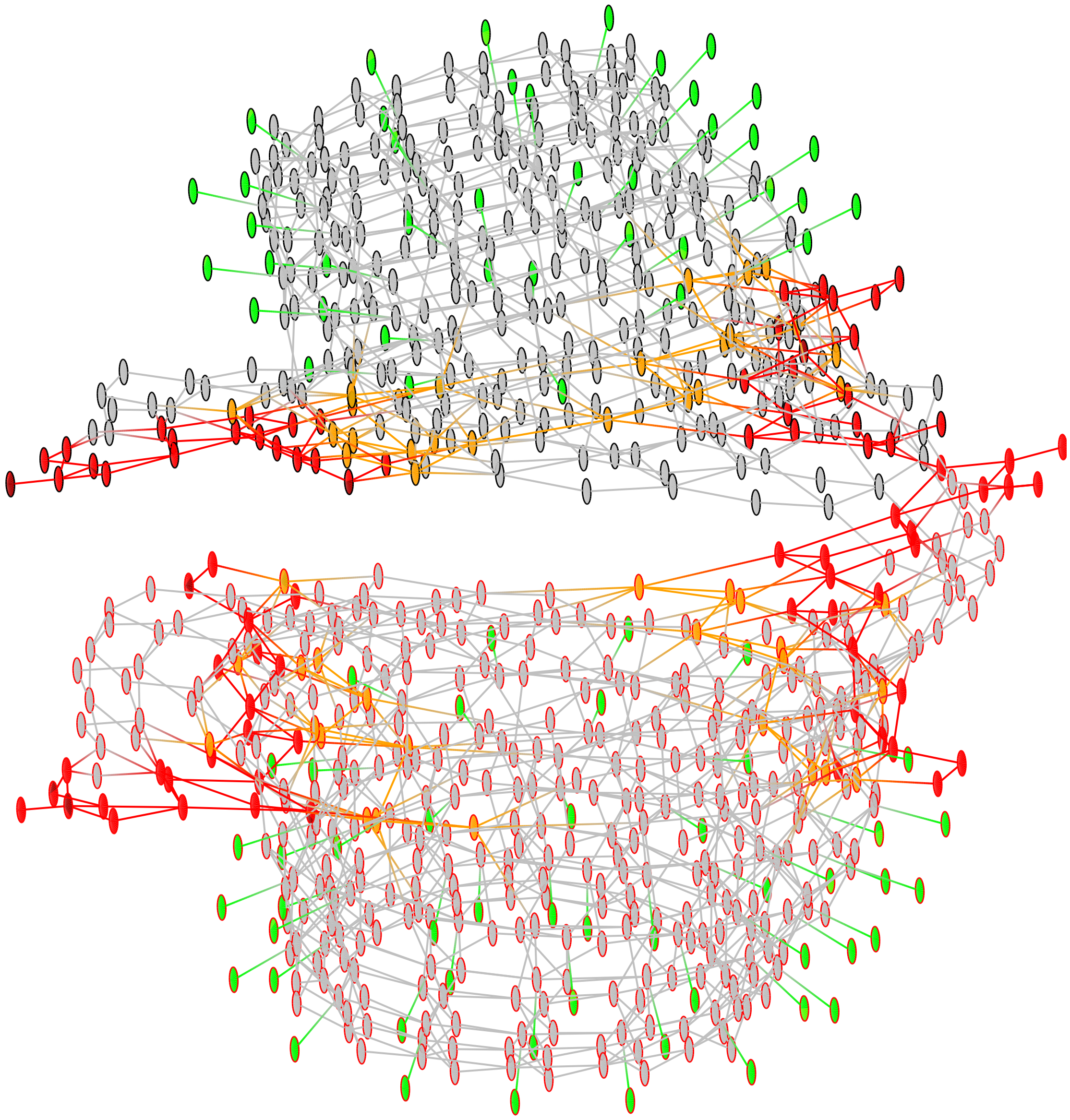
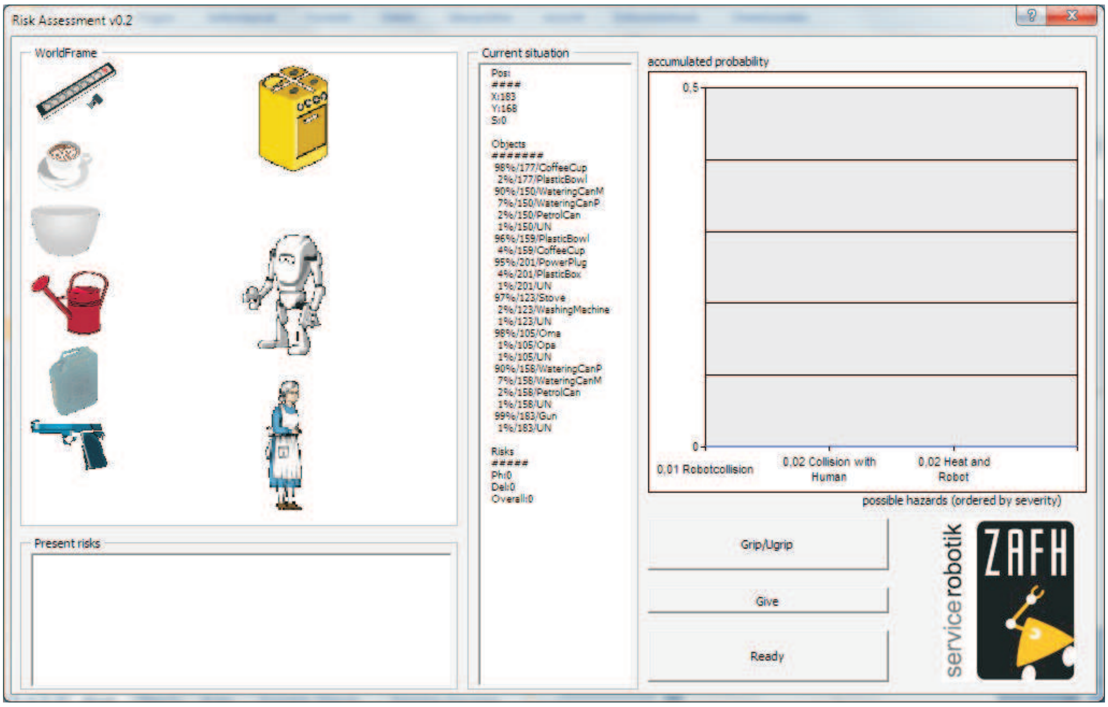


Figure 8.12: Graphical representation of the mental action space with safety clearance $n = 2$ including the monster. Red-colored situations denote mortal situations, orange-colored situations represent situations with medium risks. Red-outlined situations denote that the player fetched the emerald (lower part) and black-outlined situations denote that no emerald is collected (upper part). Green-colored situations represent goal situations.



```
Pos:
####
X:183
Y:168
S:0

Objects
#######
 98%/177/CoffeeCup
  2%/177/PlasticBowl
 90%/150/WateringCanM
  7%/150/WateringCanP
  2%/150/PetrolCan
  1%/150/UN
 96%/159/PlasticBowl
  4%/159/CoffeeCup
 95%/201/PowerPlug
  4%/201/PlasticBox
  1%/201/UN
 97%/123/Stove
  2%/123/WashingMachine
  1%/123/UN
 98%/105/Oma
  1%/105/Opa
  1%/105/UN
 90%/158/WateringCanP
  7%/158/WateringCanM
  2%/158/PetrolCan
  1%/158/UN
 99%/183/Gun
  1%/183/UN

Risks
#####
Ph:0
Del:0
Overall:0
```

Figure 8.13: The 2D-world, including typical objects and the robot[4](cf. Ertle et al., 2010c).

the respective principle, while the conclusion generates the numeric risk expression. The premise consists of logical conjunctions considering the presence of objects or attributes. For sake of simplicity, only Safety Principles are regarded that describe the interaction of two objects. In this connection, one object is grasped and ergo manipulated by the robot, the other object is located in the environment. In this connection, the confidence for the different identification hypotheses P_{ij} for each object are assumed to be $P_i = \sum P_{ij} \leq 1$. The remaining difference $P_{unknown,i} = 1 - P_i$ is assumed to decode that the object cannot be identified and could be an arbitrary object (either known or unknown). Thus, a Safety Principle is defined assuming unidentified objects to be lethal (see 8.3, last). Finally, the conclusion consists of a part estimating the probability, and the other, for estimating the severity of a threatening mishap or accident. The both multiplied denote the risk estimate concerning the respective hazard.

Experimental setup

The simulation scenario is shown in **Figure 8.13.** The scene objects are located in the 2D-world space at left-hand side. The situation description is located in the middle, the risk curve is drawn on the right part of the dialog. The object knowledge is realized with a table, see **Table 8.3**. Here, natural name attributes are assigned to the object identifiers. The prefix 'A:' indicates that the expression is an attribute. For instance, the object 'PlasticBowl' has associated attributes like 'graspable', 'plastic', and 'liquid_container'.

[4] Icons are from `http://openclipart.org/` [online; accessed 05. May 2013]

Object knowledge base

Identifier	Attributes			
OBJ_Human	A:human	A:moving		
OBJ_PlasticBowl	A:graspable	A:plastic	A:liquid container	
OBJ_CoffeeCup	A:graspable	A:liquid container	A:hot liquid	
OBJ_PowerPlug	A:graspable	A:electric	A:plastic	
OBJ_Stove	A:heat			
OBJ_Gun	A:graspable	A:lethal		
OBJ_WateringCanP	A:graspable	A:plastic	A:liquid container	
OBJ_WateringCanM	A:graspable	A:metal	A:liquid container	
OBJ_Oma	A:human			
OBJ_PetrolCan	A:graspable	A:flammable liquid	A:plastic	A:liquid container
OBJ_UN	A:lethal			

Table 8.3: The list of objects comprised in the object knowledge base. Each object has one or more attributes (Ertle et al., 2010c).

Scenario	Likelihood	Consequences	Cumulative probability
sc_1	p_1	x_1	$P_1=P_2+p_1$
sc_2	p_2	x_2	$P_2=P_3+p_2$
...	...	...	...
sc_{n-1}	p_{n-1}	x_{n-1}	$P_{n-1}=P_n+p_{n-1}$
sc_n	p_n	x_n	$P_n=p_n$

Table 8.4: List of ordered hazard scenarios, according to Kaplan and Garrick (1981).

The robot can be moved by the user via mouse interface, and it can be commanded to grasp or un-grasp objects close to the robot. While the user changes the scene, the risk assessment module continuously determines the present risk. The situation description consists of the robot position and velocity, and a list of 'recognized' objects. For each object it is listed the confidence of the recognition, the distance, and the object identifier. With the help of the object identifier, the object attributes can be inferred from the object knowledge base. The resulting information description denotes the basis for the risk assessment module. The output of the risk assessment module is a risk vector, containing the risks that are related to the respective hazardous object interactions. In accordance to Kaplan and Garrick (1981), risk can be represented by the risk triplet $risk = < sc_i, p_i, x_i >$, with $i = 1, 2, ..., n$. Each triplet represents a hazardous scenario which is also formalized by a Safety Principle. A list of scenarios is shown in **Table 8.4**. In order to generate meaningful illustration of the risk, a so-called risk curve can be realized (Kaplan and Garrick, 1981). For this reason, the scenarios are ordered with regard to their consequences $x_1 \leq x_2 \leq x_3 \leq \ldots \leq x_n$. The cumulated probability is computed by adding the likelihoods bottom-up, see **Table 8.4**. The resulting risk curve is immediately displayed in the field right-hand side of the dialog.

The applied Safety Principles are shown in **Table 8.5** being organized as a conditional part, the premise, the severity, and probability part as conclusion. The premise refers to the presence of either objects or object attributes. The severity and probability part can be parameterized either via fixed values or predefined functions in dependence of

Safety knowledge base

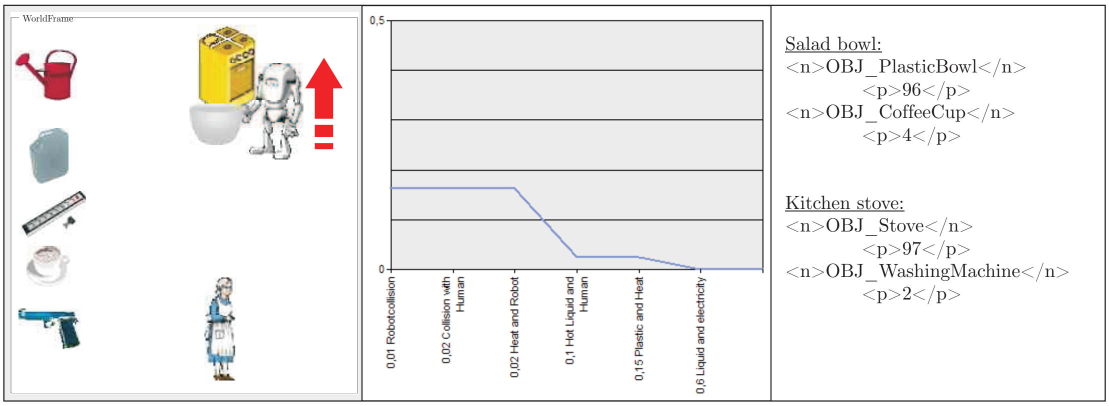
	Conditional part		Severity part				Probability part					
#	Object 1 (Robot/grasped)	Object 2 (in environment)	Function	Input	Sev. 1 (0...1)	@Val1 of Inp.	Function	Input	Prob. 1 (0...1)	@Val1 of Inp.	Prob. 2 (0...1)	@Val2 of Inp.
1	A:hot liquid	A:human	NONE	NONE	0.1		LinearFunction	distance	1	20	0	70
1	*		NONE	NONE			LinearFunction	speed	1	2	0	0.3
2	A:plastic	A:heat	NONE	NONE	0.15		LinearFunction	distance	1	10	0	50
3	A:chemical	A:human	NONE	NONE	0.6		StepMFunction	distance	1	30		
4	@	A:human	NONE	NONE	0.02		LinearFunction	distance	1	0	0.1	70
4	*		NONE	NONE			LinearFunction	speed	1	2	0	0
5	A:electric	A:liquid container	NONE	NONE	0.6		LinearFunction	distance	1	0	0	50
6	A:liquid container	A:electric	NONE	NONE	0.6		LinearFunction	distance	1	10	0	100
6	*		NONE	NONE			LinearFunction	speed	1	1	0	0.3
7	@	A:heat	NONE	NONE	0.02		LinearFunction	distance	1	10	0	60
8	A:heat	A:plastic	NONE	NONE	0.15		LinearFunction	distance	1	10	0	30
9	A:electric	A:human	NONE	NONE	0.6		LinearFunction	distance	1	50	0	100
10	@	_	NONE	NONE	0.01		LinearFunction	distance	1	10	0	40
10	*		NONE	NONE			LinearFunction	speed	1	2	0	0
11	A:flammable liquid	A:heat	NONE	NONE	0.7		LinearFunction	distance	1	20	0	100
12	A:heat	A:flammable liquid	NONE	NONE	0.7		LinearFunction	distance	1	20	0	100
13	A:flammable liquid	A:electric	NONE	NONE	0.7		LinearFunction	distance	1	0	0	50
14	A:electric	A:flammable liquid	NONE	NONE	0.7		LinearFunction	distance	1	0	0	50
15	OBJ_UN	_	NONE	NONE	1		StepMFunction	NONE	1	1		
16	OBJ_Gun	A:human	NONE	NONE	1		StepMFunction	NONE	1	1		

Table 8.5: List of the Safety Principles in the safety knowledge base (numbered in the first column), consisting of three parts, the conditional, severity estimate, and probability estimate part. The conditional part describes the condition in which each principle becomes applicable. The severity part and probability part describe how a quantitative probability or severity value can be defined in dependence of input channels and parametrizable standard functions (cf. Ertle et al., 2010c).

measured channels. For the case that two or more channels have to be considered the operator '*' indicates the multiplication of both sets of functions. The wildcard '@' represents the robot itself. As functions representing the dependence of a measured channel with the risk value, linear and step functions are applied. Here, the parameters represent data pairs for the linear function, or the threshold value and the magnitude of a step function.

Results

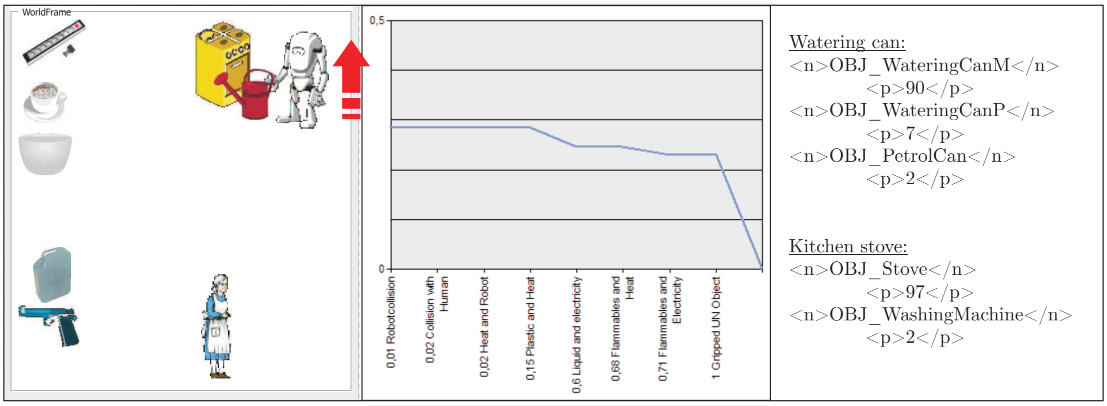
In the first experiment the robot was moved close to the kitchen stove while having grasped a plastic bowl, as illustrated in **Figure 8.14.** The object recognition module detects the objects to be a salad bowl and a kitchen stove with high confidence while there remains the chance that they are a coffee cup or a washing machine as well (see object description at the right-hand side). According to Safety Principle 8, there exists a hazard by approaching a plastic object toward a heat source. Additionally, Safety Principle 7 indicates another hazard by approaching the robot toward a heat source



Salad bowl:
<n>OBJ_PlasticBowl</n>
<p>96</p>
<n>OBJ_CoffeeCup</n>
<p>4</p>

Kitchen stove:
<n>OBJ_Stove</n>
<p>97</p>
<n>OBJ_WashingMachine</n>
<p>2</p>

Figure 8.14: The 2D-world showing the robot approaching the salad bowl toward the kitchen stove.



Watering can:
<n>OBJ_WateringCanM</n>
<p>90</p>
<n>OBJ_WateringCanP</n>
<p>7</p>
<n>OBJ_PetrolCan</n>
<p>2</p>

Kitchen stove:
<n>OBJ_Stove</n>
<p>97</p>
<n>OBJ_WashingMachine</n>
<p>2</p>

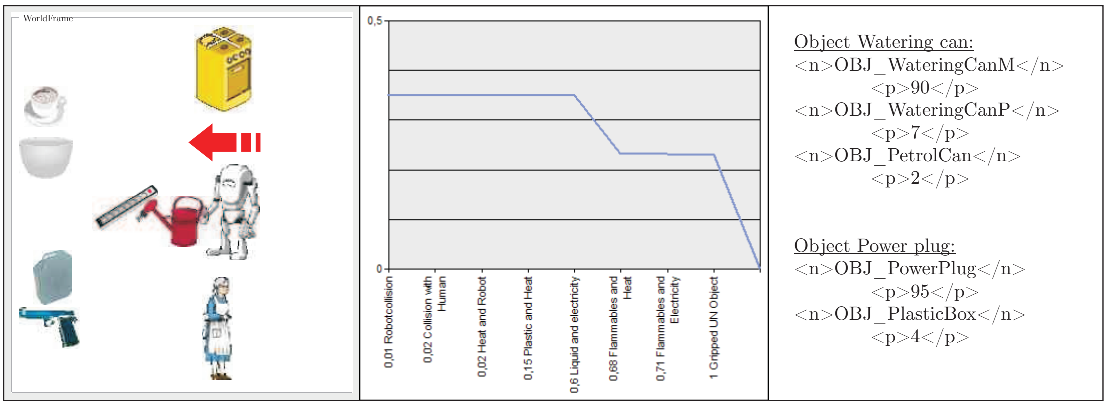
Figure 8.15: The 2D-world showing the robot approaching the watering can toward the kitchen stove.

as well. The resulting risk curve is shown in the chart, and yields an overall risk of ≈ 0.005, exactly according the parameters of the respective Safety Principle.

In the next example the robot is approached toward the kitchen stove having grasped a watering can, see **Figure 8.15.** In this case, it is assumed that the object recognition is not capable to unambiguously detect the object. There remains the uncertainty of 1%, indicated by sum of all confidence values for the identification hypotheses. From this can be concluded that there is the risk of handling an unknown object. The risks of known objects are according to the latter example.

Similarly, the effect of the unknown object remains for the example when the watering can is approached toward a power plug, see **Figure 8.16.** However, Safety Principle 6 takes effect, thus, the risk of electric shock appears. This Safety Principle is designed such that the robot approaching speed is taken into account, indicating that the risk may be increased due to possibility to spill the liquid over a huger distance.

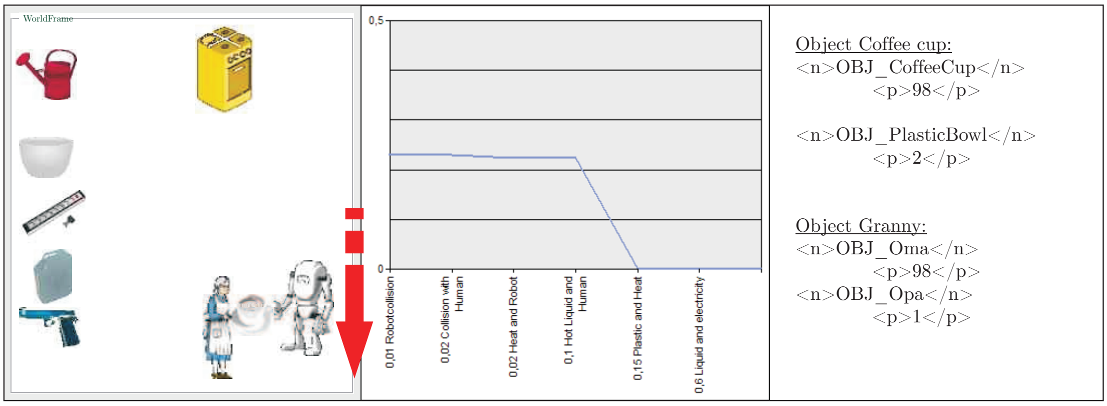
A similar effect is taken into account in the last example, were the robot quickly approaches a human while having grasped a cup of coffee, according to **Figure 8.17.** In this connection, the risk of scalding the human with spilled hot coffee is considered. This



```
Object Watering can:
<n>OBJ_WateringCanM</n>
        <p>90</p>
<n>OBJ_WateringCanP</n>
        <p>7</p>
<n>OBJ_PetrolCan</n>
        <p>2</p>

Object Power plug:
<n>OBJ_PowerPlug</n>
        <p>95</p>
<n>OBJ_PlasticBox</n>
        <p>4</p>
```

Figure 8.16: The 2D-world showing the robot approaching the watering can toward a power plug.



```
Object Coffee cup:
<n>OBJ_CoffeeCup</n>
        <p>98</p>

<n>OBJ_PlasticBowl</n>
        <p>2</p>

Object Granny:
<n>OBJ_Oma</n>
        <p>98</p>
<n>OBJ_Opa</n>
        <p>1</p>
```

Figure 8.17: The 2D-world showing the robot approaching the coffee cup quickly toward a human.

risk depends on the movement speed of the robot, as designed into the Safety Principle 1. Further examples are mentioned in more detailed research work (cf. Ertle et al., 2010c).

It is notable that a Safety principle for the collision risk is implemented as well. The Safety Principle 10 models a risk between the robot and all other objects (wildcard '_') in dependence of the robot the relative speed and the distance between robot and object. This collision risk could be modeled as well according to 'Danger Index' approach in Section 2.4.1, and shown in **Figure 2.2.**

8.3 Robotic Experiment

A small differential drive robot platform is used as an example application. The occurring of hazardous interactions of manipulated objects within its environment can be illustratively shown, and the proper functioning of the proposed approach as well.

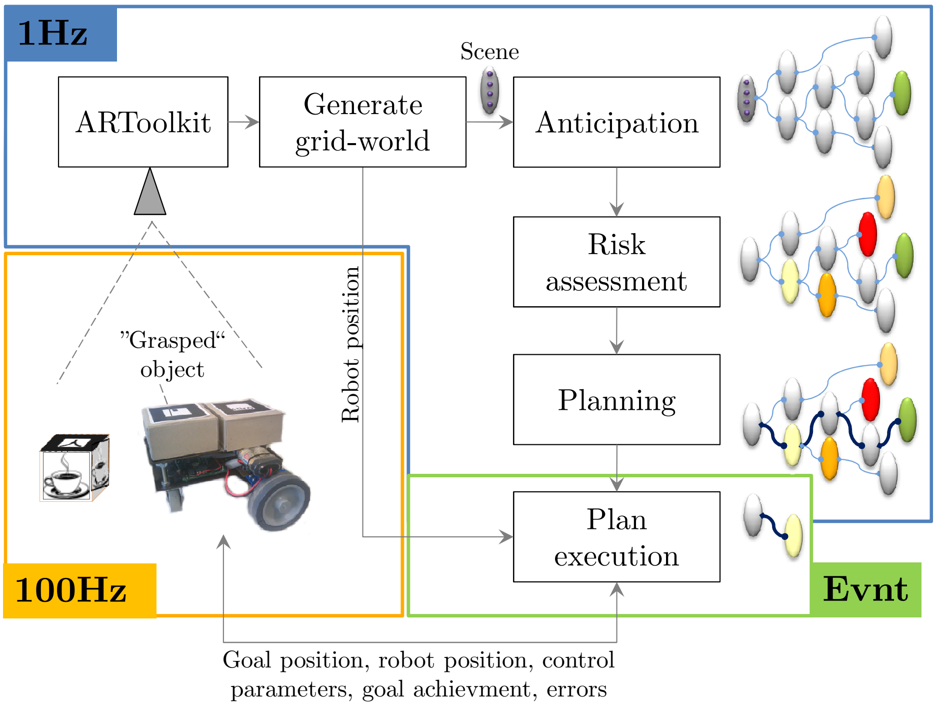


Figure 8.18: Overview over the setup and components of the differential drive robotic experiment (Ertle et al., 2012c).

8.3.1 Experimental scenario

The overall setting is illustrated in **Figure 8.18.** For sake of simplicity, robot localization and the capturing of the scene is externally realized by utilizing the (Augmented Reality) ARToolKit.[5] An overhead camera captures the plan view of the scene containing the robot and environment objects. In general, the ARToolKit is capable to recognize defined patterns within the camera picture, and provides their positions and poses in camera coordinates as output. In this experiment it is applied for detecting the robot and environment objects, which are supplied with respective patterns. Thus, the ARToolKit replaces object recognition, localization, and mapping, being not the scope of this contribution. The computation takes place at two different stages, in doing so, the term ‘robot’ refers to the whole system: 1) Perception, anticipation, assessment, and planning within the cognitive architecture takes place in a PC; 2) the reactive motion-control takes place on the robotic platform’s micro-controller. Data exchange is realized via Bluetooth connection. The position data of the scene objects (including the robot) are transformed and discretized in a 15×10 coordinate system. Object positions, poses, and their identifier represent the current perception of the robot. The robotic platform can change its position horizontally, vertically and diagonally to neighbored grid positions. Currently, the mere object positions are considered, estimation of user-related dynamics (anticipation of future positions of objects, moved by the human) is not modeled. The robot’s actions (SOM operators) are modeled such that the anticipated mental action space can be generated. The risk assessment module labels the action space with risk information. The extended action space is used to derive a safe plan.

[5] `http://www.hitl.washington.edu/artoolkit/` [online; accessed 27. September 2012]

Object knowledge base

Object name	Attributes			
human	moving	obstacle		
cleaning_solvent	flammable	liquid	toxic	obstacle
coffee_bowl	hot_liquid	liquid	plastic	obstacle
power_plug	high_voltage	plastic	obstacle	
robot	moving			
cooking_plate	extreme_heat	obstacle		
gun	mortal	obstacle		

Table 8.6: The object knowledge base comprising objects and related attributes (Ertle et al., 2012c).

This overall process is performed approximately once per second. The plan execution transmits the next valid command whenever a new plan is generated. Additionally, the robot's position, pose and present action command is send to the micro-controller. In the micro-controller, the slow global position and pose data are used for calibrating the fast reactive position data by odometry sensors. Fast position data are required to allow the adequate position control ($\approx 100Hz$).

The scene objects are small boxes with an ARToolKit pattern on the top and a picture of the represented real-world object at each side. Basically, the robot is not capable to grip scene objects by itself; the gripping procedure is performed manually by a user. Having the two modes, *object x gripped* or *no object gripped*, is sufficient for the scope of the present contribution. An object is recognized as being gripped if it is located at the platform of the robot. The robot's task is driving to the goal position, being also represented by a goal ARToolKit pattern. The environment objects can be arbitrarily positioned by the user. The current object positions and related changes are dynamically updated.

During the experiment, the robot dynamically generates plans for changing goal positions. Furthermore, the environment is dynamically changed by adding or removing objects to or from the environment, and the robot is confronted having different objects gripped. Due to hazardous interactions between the environment objects, the direct approaching of the goal might comprise unacceptable risks. An optimal path is generated by optimizing way costs ($costs_{way} = 0.1/operator$), risk costs ($costs_{risk} \in [0,1]$) and goal benefit ($costs_{goal} = -2$). Furthermore, entering of hazardous situations (defined via $risk_{intolerable} \geq 0.1$) is restricted. If no plan can be found the robot remains at its position. A risk reduction strategy is integrated in order to adequately react on suddenly appearing risks and to avoid getting stuck in hazardous situations. If the robot is in a hazardous situation, it is allowed to take this action that leads to a successor situation with the lowest possible risk (this strategy is prone to getting stuck in local minimums; actions leading to defined fail-safe states, a more sophisticated decision calculus, and the like would be required).

Several objects comprising potential hazardous interference with each other are used in the experiment. In particular, these are a human, a coffee bowl, a power plug, cleaning solvent and a cooking plate. The risk assessment is equipped with the following

Safety knowledge base

Conditional part		Severity part				Probability part			
Object 1 (grasped / carried)	Object 2 (in the environment)	Function	Input	Par. 1	Par. 2	Function	Input	Par. 1	Par.2
robot	A:obstacle	NONE	NONE	0,02		LinearF	distance	-0,25	1
A:hot_liquid	human	NONE	NONE	0,1		LinearF	distance	-0,2	1,2
A:plastic	A:extreme_heat	NONE	NONE	0,15		LinearF	distance	-0,167	1
A:liquid	A:high_voltage	NONE	NONE	0,6		LinearF	distance	-0,2	1,2
A:flammable	A:extreme_heat	NONE	NONE	0,7		LinearF	distance	-0,143	1,1429
A:mortal	human	NONE	NONE	1		LinearF	distance	-0,02	1

Table 8.7: The Safety Principles implemented within the safety knowledge base. Each row denotes a Safety Principle. The left side shows premises required to be fulfilled for activating a principle (based on the presence of respective objects or their attributes). The right side shows the parameters used for computing the corresponding risk (cf. Ertle et al., 2012c).

information: The environment objects are defined having the attributes in accordance to the object knowledge base, shown in **Table 8.6**. The Safety Principles are defined in the Safety Principle knowledge base, as presented in **Table 8.7**, being related to the presence of objects or object attributes. The presence of such object or attribute pair is the premise indicating that a Safety Principle becomes applicable. The conclusion contains the instructions for computing the related risk, consisting of probability and severity part. This computation is either based on parameterized data or measured data. Each Safety Principle which can be applied to the current environment, contributes to the overall risk of the situation (risks are summarized). For the experiment, the risk effecting variable is modeled being dependent of the measured distance of two objects (risk instruction based on input = distance [grid fields] see **Table 8.7**). The dependence of distances can be well illustrated within the 2D camera picture or within camera screen shots as well. The anticipated situation risk level is decoded by colors and drawn as indicator boxes in the video picture.

8.3.2 Experimental results

During the experimental run, multiple scenarios were tested. At first, the anticipated action space contains no risks; a straightforward plan can be generated (blue line), see **Figure 8.19.** In the second scene, a human is located at the center of the scene. As the human is also defined as an obstacle (‘A:obstacle’), certain (collision) risks appear around the human. The path is altered accordingly. Although risks under a certain threshold are not displayed, they are considered anyway. Hence, the risk- and way-cost balance induces a path keeping more distance.

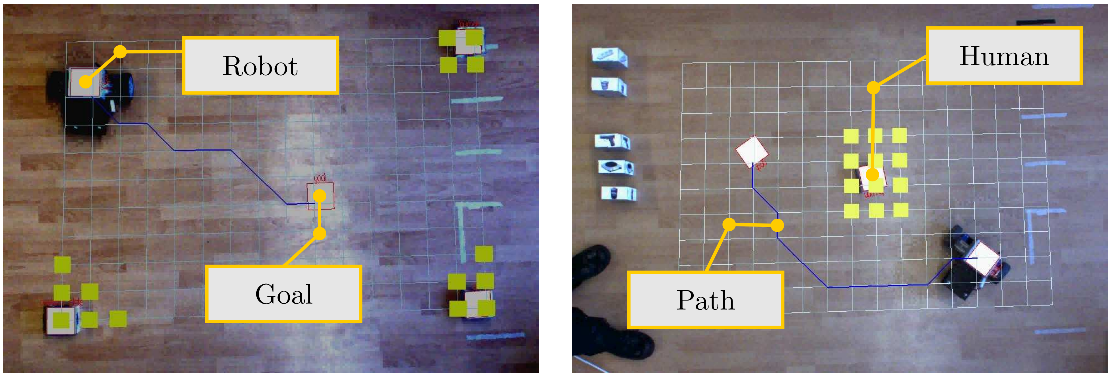


Figure 8.19: Snapshots during the experiment showing straightforward goal approach (left), and avoiding the obstacle 'human' (right) (cf. Ertle et al., 2012c).

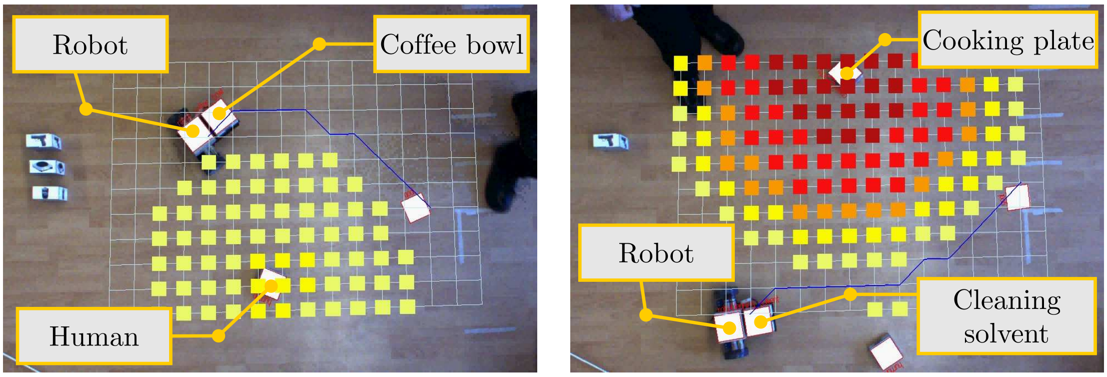


Figure 8.20: Snapshots during the experiment moving at full speed (only 0/1 available) being too high for approaching a human with a coffee bowl, (left), and hazardous interference between a potential hot cooking plate and the gripped cleaning solvent (right) (cf. Ertle et al., 2012c).

In the next scene, see **Figure 8.20** (left), the robot transports a coffee bowl, which potentially contains hot liquid ('A:hot_liquid'). The risk of scalding humans appears (the robot can move only at one fixed velocity and this is assumed to be too high). The scene at right-hand side, the relatively more hazardous interference between the cleaning solvent (being a flammable substance and gripped by the robot) and the cooking plate is shown.

Since a coffee bowl potentially contains liquid, an additional hazardous interference with the power plug appears. No risk-free plan can be found, as shown in **Figure 8.21** (left). In the scene at right-hand side, the situation is altered such that a risk free-path can be found. Any interference between the cleaning solvent and humans is not modeled, thus, the robot is allowed to closely pass the human.

In the last scene, the robot has grasped a gun. A very high risk appears when a human is equally present, according to the last Safety Principle in **Table 8.7**. The risk ranges according to the given parameters within the interval $[0.98, 0.64]$ (distance within the scenario $1 \ldots 18$). In consequence, all action alternatives are too risky ($risk_{intolerable} \geq 0.1$). If an action exists to reach a situation with low or no risk (e.g. a fail-safe state), this

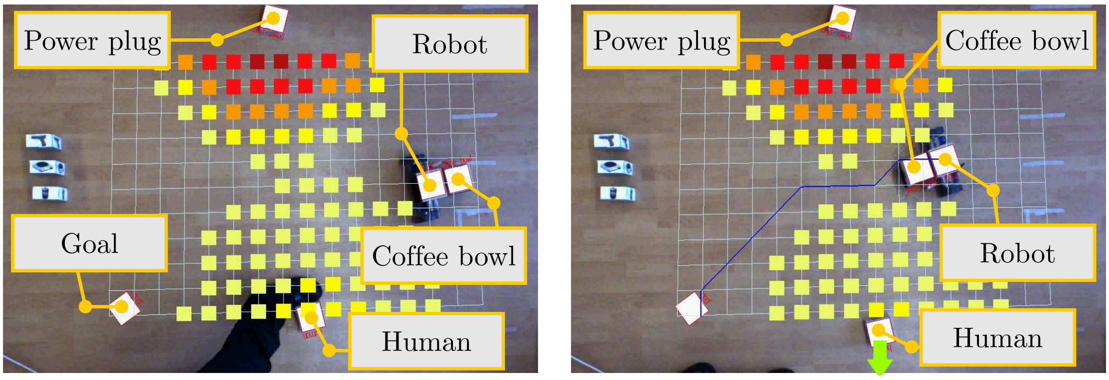


Figure 8.21: Snapshots during the experiment with risk scalding a human with hot coffee, and risk of electrical shock being provoked by approaching a coffee bowl toward a power plug. No path is available without risk (left), and changed positions of objects with a risk-free path (right) (cf. Ertle et al., 2012c).

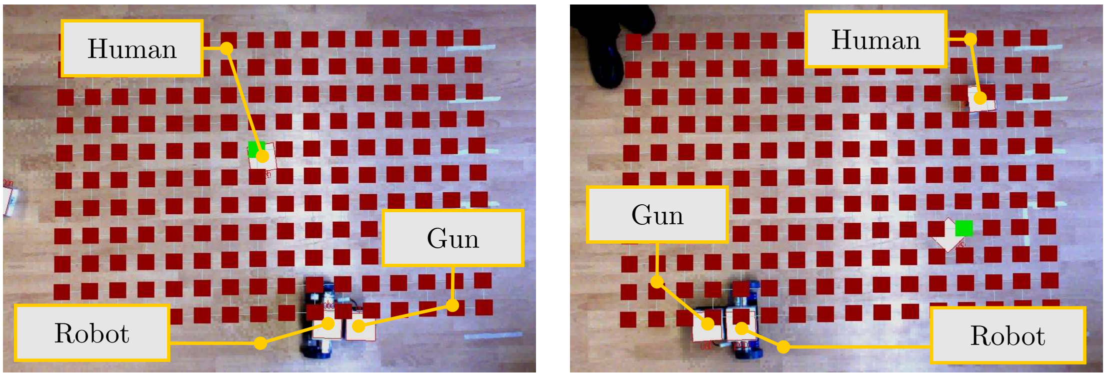


Figure 8.22: Snapshots during the experiment showing the robot having grasped a gun. The robot has no command to ungrasp the gun, only moving commands are available. According to an implemented risk reduction strategy, the robot is allowed to perform this action (out of the set of available actions) that results in a situation with the lowest possible risk. In consequence, the risk can be minimized by maximizing the distance to the human. The maximum distance is at the lower right corner for the scenario at left-hand side, and at the lower left corner for the scenario with the different position of the human at right-hand side. Red labels indicate very high risk; there are small differences of the risk values which are not visible by the color gradation.

action would be selected by the robot. However, there is no such situation, thus, the robot is allowed to take the action with the lowest risk in order to demonstrate a risk mitigating behavior. Since the robot is only provided with movement actions, and hence, it is not able to grasp or ungrasp an object (the robot is 'confronted' having the object grasped that is deposited by the experimenter), the robot can only take movement actions. With regard to the risk reduction strategy these are the actions that maximize the distance to the human. This scenario is illustrated in **Figure 8.22.** The risk of each situation is decoded with different colors. Due to the high risk and minimal differences of the available risks, the colors for all situations are equal. Since the risk values are (slightly) different, the robot can minimize the risk by maximizing the distance to the human.

9 The residual incompleteness of the safety-related knowledge: A perspective on learning approaches

9.1 Learning in robotics

An alternative to manually program robots denotes ‘imitation learning’, also known as ‘learning from demonstration’ or ‘programming by showing’. There are approaches were the human motion is recorded by motion capture tools in order to transfer the motion to the robot, yielding to the so-called ‘correspondence problem’ (Nehaniv and Dautenhahn, 2002). The correspondence problem originates from the differences between the mechanics of a human and a robot system. Hence, in other approaches the robot is taught by moving the robot, either directly or via a teleoperating interface. However, learning from demonstration requires for generalization mechanisms in order to allow adaptability to slightly different situations, otherwise, a mere reproduction of the demonstration may result.

Despite this, a robot might need the capability for self-improvement for the majority of problems, especially, if a robot behavior is required to be adapted to unforeseen situations, or a policy learned from demonstration needs to be improved, and the like. These problems are usually addressed in the reinforcement learning framework, which is inspired by the way animals and humans improve their behavior by trial-and-error (cf. Thorndike, 1932). In this connection, the well-known key problem is the exploration, exploitation dilemma. Since it is not known in advance which behavior is advantageous, and which turns out to be counterproductive, various different actions must be tried in order to discover an efficient strategy (Sigaud and Peters, 2010).

9.1.1 Implications of learning

If robots are required to be able to learn, this reveals additional questions with regard to the problem to ensure safe robot behavior. Learning capabilities implicate that the learning system is changed by the learning process. Hence, the system behavior is not anymore determined by its initial (designed) structure, and not only structure deviations due to occurring faults are of interest anymore. Learning changes the systems structure; thus, its behavior can as well be determined by the newly learned aspects. The essential consequence is that the system differs from its initially designed version. For this reason, it was suggested in the latter sections to integrate a safety knowledge base into the robot system in order to realize that the robot becomes aware of hazards (and as well of similar novel hazards), and to enable it appropriately to consider the information about hazards. In consequence, the robot system shall be allowed to autonomously adapting to novel situations, but within safe 'boundaries'. If the majority of hazards can be prevented by implementing a safety-related knowledge base, the question arises how residual hazards are treated that preferentially occur during the operating phase of the system.

From the safety-related point of view, it can be summarized that learning capabilities are basically questionable and in the first instance, not supportable. However, learning approaches can be helpful with regard to residual hazards. From a fundamental point of view, the general conditions in which learning can take place are the following.

At first, two different stages of learning must be mainly differentiated: Learning that takes place 1) during the development phase is to differentiate from learning that takes place 2) during the operation phase. Secondly, it is to distinguish which person teaches the system something. The system may learn based on the feedback of (robotic and safety) experts, by unqualified users, or both. Thirdly, the kinds of learning environments are to distinguish. Learning can take place in simulation (excluding real hazards), in real target operation environments (including the full range of hazards), or in laboratory environments, which might range from simple to almost target operation environments (including various real hazards but under surveillance conditions).

9.1.2 The problem of incomplete safety knowledge

To *"[...] overcome the practically impossible problem of preidentifying the full range of kinds of situations robots and other agents will get into during normal interaction with their environments, [...] we should [...] seek to build robots, and artificial agents in general, that are autonomous"* (Smithers, 1997). Of course, the author suggested this, having the complexity and NOT the safety problem in mind. Unfortunately, this statement also holds for safety considerations. Thus, the realization of learning-based autonomy of technical systems can imply new challenges for the system safety process, as not fully-specified (learning) systems can imply unconsidered hazards. Thus, the question arises what is the significance of the residual lack of safety-related knowledge.

Assuming two robots are confronted with a novel hazard. It can not be guaranteed that robot one, equipped with a safety knowledge base behaves safer than robot two without. All three options are possible: It might behave more safe, similar, or more risky. This depends on the specific interaction with the currently available safety knowledge. However, it can be generally assumed that a robot without learning capabilities is vulnerable for recurrent *rigid*[1] behaviors. This implies that a robot in identical conditions tends to repeatedly end in the identical (hazardous) situation.

[1] rigidity is described by Dörner (2000) as adhering to a strategy although external effects might require for changing the strategy in order to be more efficient or successful at all.

The refinement of the safety-related knowledge of the robotic system is mentioned as well in Section 7.2.5. Here, the two principle methods of knowledge refinement are mentioned being updating and learning. On the one hand, an update process implicates that safety engineers integrate new hazard aspects into the safety-related knowledge base and afterwards release a new version for the update process. On the other hand, robot learning capabilities imply that the robot integrates the new knowledge by itself. Generally speaking, if a system requires for knowledge updates this denotes a lack of autonomy because it requires system external information sources. In contrast thereof, learning improves the robot's autonomy in principle because the system itself can integrate and utilize missing aspects in order to accomplish its goals. For this and other reasons, capabilities to learn appear to be a significant aspect for autonomous robots. Thus, a perspective on trial-and-error learning (reinforcement learning) and learning from demonstration (supervised learning) in a safety-critical context is presented in the sequel.

9.2 Recent Work

9.2.1 An perspective on safety in reinforcement learning

Reinforcement learning was already considered to be applied as well to safety-critical applications, either in general, or focusing on the exploration and exploitation problem of reinforcement learning. Hence, several investigations are available. Some of these introduce safety aspects by giving stability guarantees for controllers (Perkins and Barto, 2003), even using arbitrary learning algorithms (Ng and Kim, 2004).

Perkins and Barto (2003) *use domain knowledge [...] to design the action choices available to the agent. An appropriately designed set of actions restricts the agent's behavior so that regardless of precisely which actions it chooses, desirable performance and safety objectives are guaranteed to be satisfied.* The relevant domain knowledge is designed as a *Lyapunov*-function. Pursuing toward minima of the Lyapunov-function means to approach a point of stability. Actions are restricted to those having the probability of a negative gradient in the Lyapunov-function. Thus, assuming correct and complete knowledge, the Lyapunov-function 'pushes' the system toward keeping the specified performance and safety objectives.

Geibel (2001) introduces a separate risk-cost function which allows for limiting and balancing risks. Risks are limited by classifying states as unsafe, when a certain cumulative risk is exceeded. For balancing, Geibel (2001) suggest a parameter weighting benefits and risk-costs between pure greedy and pure risk-optimal policies. Varying this parameter can be used to realize cautiousness, for instance, at the beginning of the learning process.

Exploration and safety

The exploration and exploitation problem of reinforcement learning (Sutton and Barto, 1998) inherently involves the safety problem. Basically, the exploration-exploitation problem is rather the decision to either chose an action in a more or less well-known

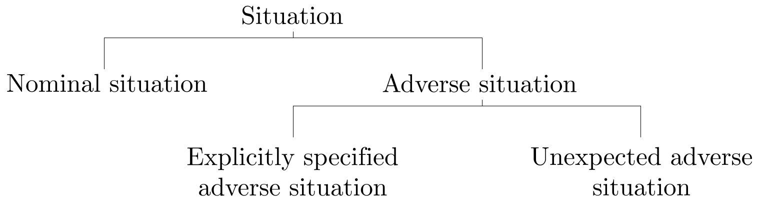

Situation
Nominal situation
Adverse situation
Explicitly specified adverse situation
Unexpected adverse situation

Figure 9.1: Classification of situations (Lussier et al., 2004).

state with more or less well-known consequences or to try something new. From a safety perspective, performing actions that lead to hazardous states are unacceptable. However, it has to be assumed that entering of such states can be acknowledged with high negative rewards.

Hans et al. (2008) focus on safe exploration. Besides introducing a *safety function*, giving pre-modeled safety information with respect to state-action pairs, the idea of a *backup policy* is mentioned. The backup policy shall be able to transfer the system from its current state to a safe state whenever an unsafe state occurs. The exploration itself is suggested to take place structured in a *level-wise* manner: Exploration is locally bounded to a state until respective exploration possibilities are exhausted.

However, Hans et al. assume that step-wise exploration from demonstrated data (apprenticeship learning) accompanied by a pre-specified backup policy ensures that the system performs safe. This approach relies on two assumptions that hold for their use-case, but might be wrong for others. On the one hand, they assume that a backup policy to avert upcoming hazards is known for every hazardous state. On the other hand, it is assumed that the exploration close to already known and nonhazardous states is acceptable if mentioned backup policy is available. This assumes that accidental states can only be reached by firstly enter hazardous states that surround accident states. Their safe exploration argument does not hold if there exist direct transitions from 'safe' to accident states and if there are hazardous states without backup policy.

Lussier et al. (2004) classifies situations as being either nominal or adverse. Adverse situations can be foreseen and specified or they might be unexpected as well (see Figure 9.1). The backup policy approach of Hans et al. might address specified adverse situations, but not necessarily unexpected (because unknown) adverse situations.

Thus, besides the challenge of principally limiting a system to safe states by specifying adverse situations, the question 'what if' remains, when the system approaches or enters unknown hazardous states, or a backup policy turns out to be inadequate.

Consequently, it is important to draw special attention to the design of safety measures that effectively avoid known adverse situations. The basic intention of such safety measures is to limit the system to safe states. On the other hand, safety measures can be used to provide the 'freedom' to explore within the given safety boundaries, for instance, to allow learning (cf. Perkins and Barto, 2003; Hans et al., 2008). With regard to considered robots, it is assumed that unknown adverse situations are potentially present due to complex tasks and environments. Thus, it is important to draw attention toward the possibility that the knowledge integrated into the system design may prevent hazards, however, in unknown adverse situations it can as well turn out to be the cause (the initiating mechanism) that transfers a hazard into an accident: Knowledge that is integrated into the system (design, knowledge base etc.) is the cause for an accident due to its inadequateness with regard to an unknown adverse situation. Because of the mentioned reasons, the following topic shall be focused:

- The design of safety-related knowledge by utilizing supervised learning methods in order to improve the safety knowledge design process, and
- learning with scope on the plasticity of knowledge, for instance, if systems are faced with unknown hazardous situations for which no fail-safe strategies (or backup policies) are available.

9.3 Safety Aspects in a Non-stationary Reinforcement Learning Problem

Temporal difference (TD) learning was inspired by cognitive and biological sciences, since it can be viewed as an extension of timing drawbacks of the models for conditioning (Rescorla-Wagner model of Pavlovian conditioning). Additionally, TD learning is also related to the neuroscience, as a similarity of the specific behavior in conditioning experiments and the behavior of midbrain dopamine neurons in the brain is found (Barto, 2007). The basic unenhanced TD learning algorithms are '$\boldsymbol{Q}$-learning' (Watkins, 1989) and 'Sarsa' (Rummery and Niranjan, 1994; Singh and Sutton, 1996), but by now, there are available multiple variances of these algorithms (cf. Kaelbling et al., 1996).

Since the following investigation shall draw attention to the exploration/exploitation dilemma, the basic TD learning algorithms $\boldsymbol{Q}$-learning and Sarsa are combined with a selection of methods to control the exploration/exploitation behavior. As representative selection of these exploration/exploitation policies, it is investigated $\boldsymbol{Q}$-learning and Sarsa in combination with the so-called '$\boldsymbol{\varepsilon}$-Greedy', 'Softmax Action Selection' (Softmax), 'Value-Difference Based Exploration' (VDBE), 'VDBE-Softmax', and 'Reinforce VDBE-Softmax', respectively. Indeed, more complex methods are available such as counter-based (Thrun, 1992), or confidence bound-based (Kaelbling, 1993) methods. However, the $\boldsymbol{\varepsilon}$-Greedy method is often hard to beat, also with regard to more complex methods (cf. Vermorel and Mohri, 2005; Tokic and Palm, 2011). Furthermore, Softmax, VDBE, and VDBE-Softmax are intended to improve the drawbacks of $\boldsymbol{\varepsilon}$-Greedy's parameter tuning, the Reinforce VDBE-Softmax method is a novel parameter-free variant.

The agent's exploration/exploitation policy is a formal description that defines when and to which extent the agent is explorative, or when it utilizes its knowledge (value function) for action selection. The control of the exploration/exploitation behavior can take place by implementing a constant exploration rate (the agent always explores to a certain extent), for instance, by using the $\boldsymbol{\varepsilon}$-Greedy policy. In this regard, the exploration can be improved by assigning high probability of selection to those actions with high expected rewards (e.g. Softmax policy). In contrast to the latter policies, the control can also be based on the TD-error. The TD-error is the difference between the expected and received reward (and discounted rewards, respectively). Generally spoken, a significant TD-error indicates that the value function does not match the observations the agent made by performing an action. Thus, a significant TD-error can be used to indicate that the value function should be rendered (learned) more precisely. The VDBE, VDBE-Softmax and the Reinforce VDBE-Softmax policy are such TD-error-based exploration/exploitation control policies. For the VDBE policy, a local exploration rate is assigned to each state. The VDBE-Softmax policy is a combined policy which behaves like ε-Greedy policy (greedily favors actions with high rewards, however, with regard to the VDBE-like local exploration rate) when the TD-error is low and selects actions according to a Softmax policy in case of a significant TD-error. The mentioned policies have to be adjusted with a parameter which globally effects the exploration/exploitation ratio. Reinforce VDBE-Softmax policy integrates a meta instance to adapt this parameter by comparing the current performance of the agent with a 'slow' performance baseline. Detailed information with regard to the policies can be found in the contributions of Sutton and Barto (1998); Vermorel and Mohri (2005); Tokic and Palm (2011), Tokic et al. (2012), Tokic and Palm (2012).

9.3.1 The non-stationary experiment

The cliff-walking problem proposed by Sutton and Barto (1998) is usually used to compare the characteristics of algorithms under uncertainty. The task is to pass a hazardous cliff in order to reach a goal position. This scenario is extended in order to investigate reinforcement learning in the focus of safety considerations by Tokic et al. (2012). In this connection, it is the basic concept to 'suddenly' change the scenario in order to simulate an upcoming hazardous situation which the system can not recognize as a distinct state (at least in the first instance). This may result due to system component failures, the inability to differentiate the states, and the like. Thus, the cliff-walking scenario is reformulated as a non-stationary experiment. The non-stationary experiment starts with phase a). Here, it comprises one hazard state. The scenario is changed in phase b). Here, the cliff is extended; hence, the scenario suddenly comprises more hazardous states, and forces the agent to alter the learned optimal path. In phase c) the problem becomes worse, as the scenario is changed again, and only one state remains as bottleneck to pass the hazardous states. The scenario and its different phases are shown in **Figure 9.2.**

Basically, the agent in the cliff-walking problem has the goal to learn a path from the start state S to the goal state G. The rewards for performing the path are the absolute costs. For each action taken the costs are increased, $r_{action} = -1$, and iff the goal state is reached the costs are reduced, $r = r + 1$. The scenario also comprises so-called cliff states, which, when entered, lead to a high negative reward, $r_{cliff} = -500$. In contrast to the original cliff walking problem, the entering of a cliff state does not reset the agent back to the start state S, instead, the episode is terminated. This can be compared to a scenario in which an agent is reset to a fail-safe state in case of entering a hazardous state (here, the fail-safe state is equal to its initial starting position) and the next learning episode starts.

An advantage of this experimental setup is that the 'action-less' transition from the cliff states into the starting state is avoided. This in turn offers the opportunity to model

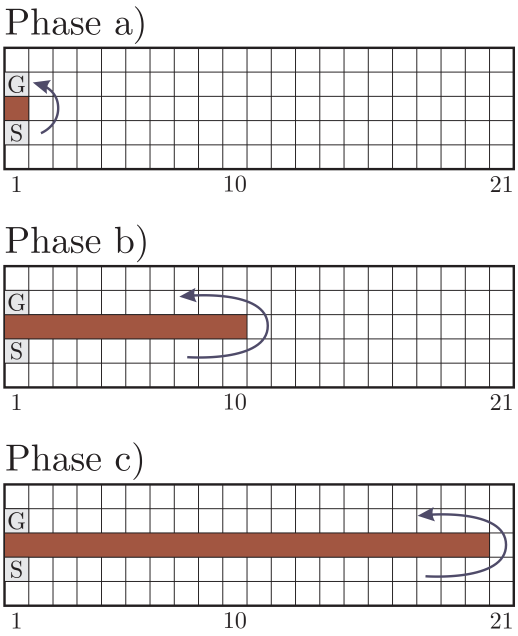


Figure 9.2: The non-stationary cliff-walking scenario for investigating safety-related performance of reinforcement learning approaches, according to Tokic et al. (2012).

the transition probabilities of the scenario with the help of a Markov transition matrix.[2] In consequence, the reachability of the cliff states of the scenario can be computed.

9.3.2 The hazard potential of the scenario

In former investigations (cf. Tokic et al., 2012) the performance of the learning approaches was investigated by taking into account the cliff-falls per action rate (entering of hazardous situations per number of taken actions) in a specific scenario. Consequently, the results are specific for the respective scenario. In order to improve the measure expressing the safety performance, the cliff-falls shall be considered to be relative to the cliff-fall probability. In other words, the failure of a learning agent is seen in the light of its chance to fail. The chance to fail can be computed with the Markov transition matrix and depends on the transitions probabilities of the scenario to reach a cliff state. Hence, the chance to fail in a scenario can be called the hazard potential of the scenario.

[2] In the original cliff-walking example and in the former experiment of Tokic et al. (2012), the agent was reset to its initial state in case of entering a cliff state without considering this transition as an action. This results problems if the performing of the agent over a whole episode shall be compared to the transition probabilities of the scenario, which are in turn computed with the help of a Markov transition matrix. It is difficult to model the scenario with a Markov transition matrix with 'action-less' transitions. A workaround could denote to model the scenario with direct transitions from the states that usually lead to a cliff state, to the start state. But this results problems analyzing why the starting state is entered (via normal movement actions or via cliff-fall). If the cliff states terminate an episode they appear as absorbing states in the Markov transition matrix.

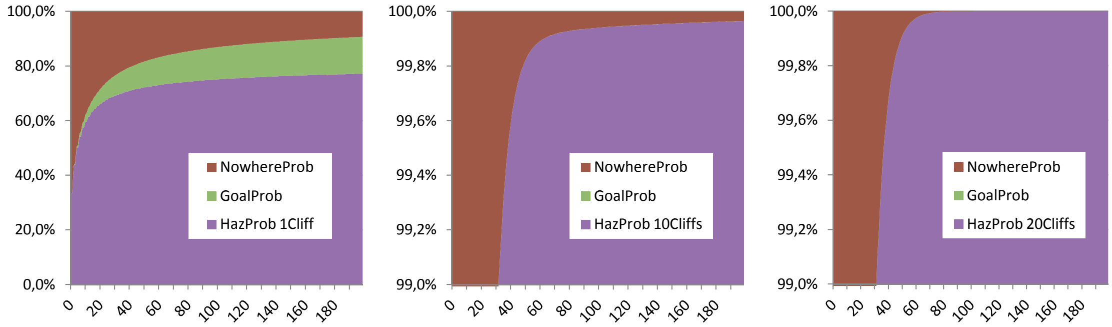
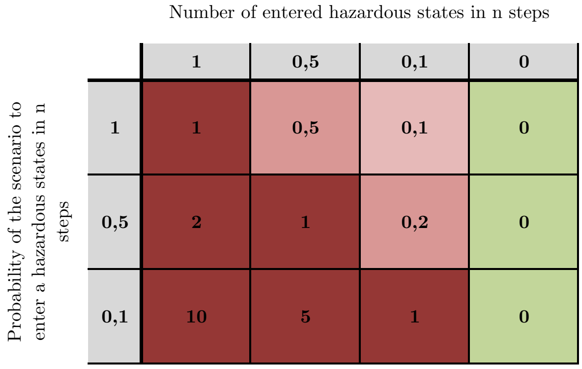


Figure 9.3: The transition probabilities in the scenario computed for a number, n, of taken actions at the abscissa. The ordinate shows the cumulated probabilities of the agent ending up after n actions in a hazardous state, in the goal state, or somewhere else when the episode is terminated due to the limitation of actions. Note that diagram two and three show the ordinate range $[0.99, 1]$.

Similar to Section 8.1.3, the Markov transition matrix, $\boldsymbol{M}$, of the scenario can be generated (goal and cliff states are absorbing states). By raising the transition matrix, $\boldsymbol{M}$, to the power of n, the transition probabilities of applying a number, n, of actions can be computed. For each number of considered actions, the transition probability from the starting point to a cliff state can be computed. In case of multiple cliffs, the transition probability to the cliffs can be cumulated. Hence, the probability to end up in a hazard state (cliff state) can be computed taking into account a specific number, n, of actions. The maximum number of actions in the scenario is limited to 200 steps, thus, a look-up table for $n = 1 \ldots 200$ can be generated for each phase of the experiment. The look-up tables are shown in **Figure 9.3** as diagrams. For phase a), it is visible that there exists an increasing chance to end up in a hazardous state if the number of performed actions increases. For phase b) and c) this becomes worse. The probability to reach the goal state is not visible anymore in the diagram, and the probability to enter a hazardous state is converging toward 1 (for $n = 200$ actions the probability to reach a hazardous state is $P_{haz\_state,c}(200) = 0.999999999984357$, and probability for reaching the goal state $P_{goal,c}(200) = 6,677 \cdot 10^{-12}$). Hence, the scenario in phase b) and c) is quite hazardous with a low chance to reach the goal. This knowledge about the scenario can be used to make assertions like the absolute number of entered hazardous states more general. Absolute assertions about the performance of the learning approaches can be relativized with regard to used experiment scenario. Hence, a learning algorithm entering many hazardous states in a more or less uncritical environment performs worse in comparison to an algorithm in a comparatively safety-critical environment. This relation is illustrated in **Figure 9.4.** In consequence, the hazard performance measure is the quotient of the average number of entered hazardous states and the probability in the scenario to enter hazardous states. If this quotient is around 1, the learning algorithm performs similar to a randomly walking agent. Numbers above 1 indicate that it performs worse than randomly taking actions (rigid behavior; hazardous states are preferred), numbers close to 0 are desired. From the safety point of view, the number of performed actions does not matter if no hazardous states are entered. The number of required actions is of interest for performance considerations. Hence, the hazard performance is computed with regard to the hazard potential for each episode, i, for the phases, $p \in \{a, b, c\}$,



Number of entered hazardous states in n steps

Probability of the scenario to enter a hazardous states in n steps	1	0,5	0,1	0
1	1	0,5	0,1	0
0,5	2	1	0,2	0
0,1	10	5	1	0

Figure 9.4: Hazard performance measure for learning algorithms taking into account the experiment scenario characteristics. The number of cumulated hazardous states is set into relation with the probability to enter hazardous states in the scenario.

without taking into consideration the optimality of the found solutions. In consequence, it is the number of entered hazardous states ($\in \{0,1\}$) divided by the hazard probability,

$$HP_i = \begin{cases} 0 & : n_{hazstates} = 0 \\ \frac{n_{entered\_hazstates,i}}{P_{hazstate,p}(n_i)} & : n_{hazstates} \geq 1 \end{cases} . \tag{9.1}$$

The hazard probability is based on the number of applied actions, n, which bear a specific probability to enter a hazardous state, for a scenario that contains a number, $n_hazstates \geq 1$, of hazardous states. The scenario specific hazard probability comes from the aforementioned Markovian look-up table. In consequence, the hazard performance measure is low if the chosen courses of action comprise a low number of entered hazardous states. If the scenario is uncritical with single hazardous states that are difficult to reach, the hazard performance measure becomes higher for the identical number entered hazardous states. Hence, in a scenario with 10-times smaller chance to enter a hazard, the equally ‘safe’ or ‘hazardous’ courses of action are expected to result in entering as well 10-times less hazardous states.

9.3.3 Experimental setup

As the episodes are terminated more quickly, multiple learning episodes might become necessary in order to achieve a comparable learning success. Therefore, the scenario phases consist of 500 episodes for phase a), 1500 for phase b), and 3000 episodes for the phase c). The agent should learn how to reach the goal state G given the start state S. For each action taken the costs are increased, $r_{action} = -1$, and iff the goal state is reached the agent receives an award that compensates the minimal required way costs $(4;22;42)$. The cliff states lead to a high negative reward, $r_{\text{cliff}} = -500$, when entered.

9.3.4 Results

The hazard performance for Q-learning is shown in **Figure 9.5,** at which the graphs that correspond to the different phases are vertically arranged, and the best parameter setting is shown in the left, the worst parameter setting in the right column. Since the Reinforce VDBE-Softmax strategy is parameter-free, the results are identical for both cases, and are visualized in both categories. For the case of best parameter settings, most policies converge comparably fast toward hazard-free paths. For Reinforce VDBE-Softmax, it takes a significantly longer timer to converge, especially if the cliff is extended to 20 cliff states in phase c). For the case of worse parameter settings, ε-Greedy and Softmax keep constantly exploring. In phase b) this remains unchanged, whereas the VDBE policy shows as well an identical exploring behavior. VDBE-Softmax seems to converge in phase b), and possibly in phase c) as well. Here, ε-Greedy, Softmax, and VDBE policy remain in a significant explorative state, while Reinforce VDBE-Softmax converges toward zero within 2000 episodes.

The hazard performance for Sarsa-learning is shown in **Figure 9.6;** the arrangement is identical to the latter. For comparison, the results of Reinforce VDBE-Softmax strategy are as well identical for both parameter categories. For the case of best parameter settings, most policies as well converge comparably fast toward hazard-free paths. For Reinforce VDBE-Softmax, it takes a significantly longer timer to converge, especially if the cliff is extended to 20 cliff states in phase c). For worst-case parameter settings, ε-Greedy and Softmax keep constantly exploring, Softmax at a significantly higher rate than ε-Greedy. The VDBE-Softmax strategy seems to learn quickly, but remains in an exploring behavior to a certain extent. This does not change in general for the phase b). Here, all strategies remain in an exploring behavior except Reinforce VDBE-Softmax. The VDBE strategy seems to converge, but after 200 episodes it starts to ‘unlearn’. The characteristics do not change in phase c) but both, ε-Greedy and VDBE-Softmax, appear to have slight convergence toward zero.

In general, the hazard performance of Q-learning and Sarsa are comparable. An overview of the averaged hazard performances is provided in **Figure 9.7.** Here, the averaged hazard performance for each phase and for all phases together are shown, Q-learning in the left, and Sarsa in the right column, best-case parameter settings in the upper, worst-case parameter settings in the lower diagram.

For properly chosen parameters, there seems to be no significant difference, neither between Q-learning and Sarsa, nor between the different exploration/exploitation strategies, except for Reinforce VDBE-Softmax. However, the parameter-free Reinforce VDBE-Softmax strategy performs with a lower averaged hazard performance in comparison to the other worst-case parameter settings and with a significantly lower hazard performance for Sarsa in comparison with Q-learning. For all considered cases, VDBE-Softmax performs with a comparable or lower hazard performance than the other strategies, except Reinforce VDBE-Softmax.

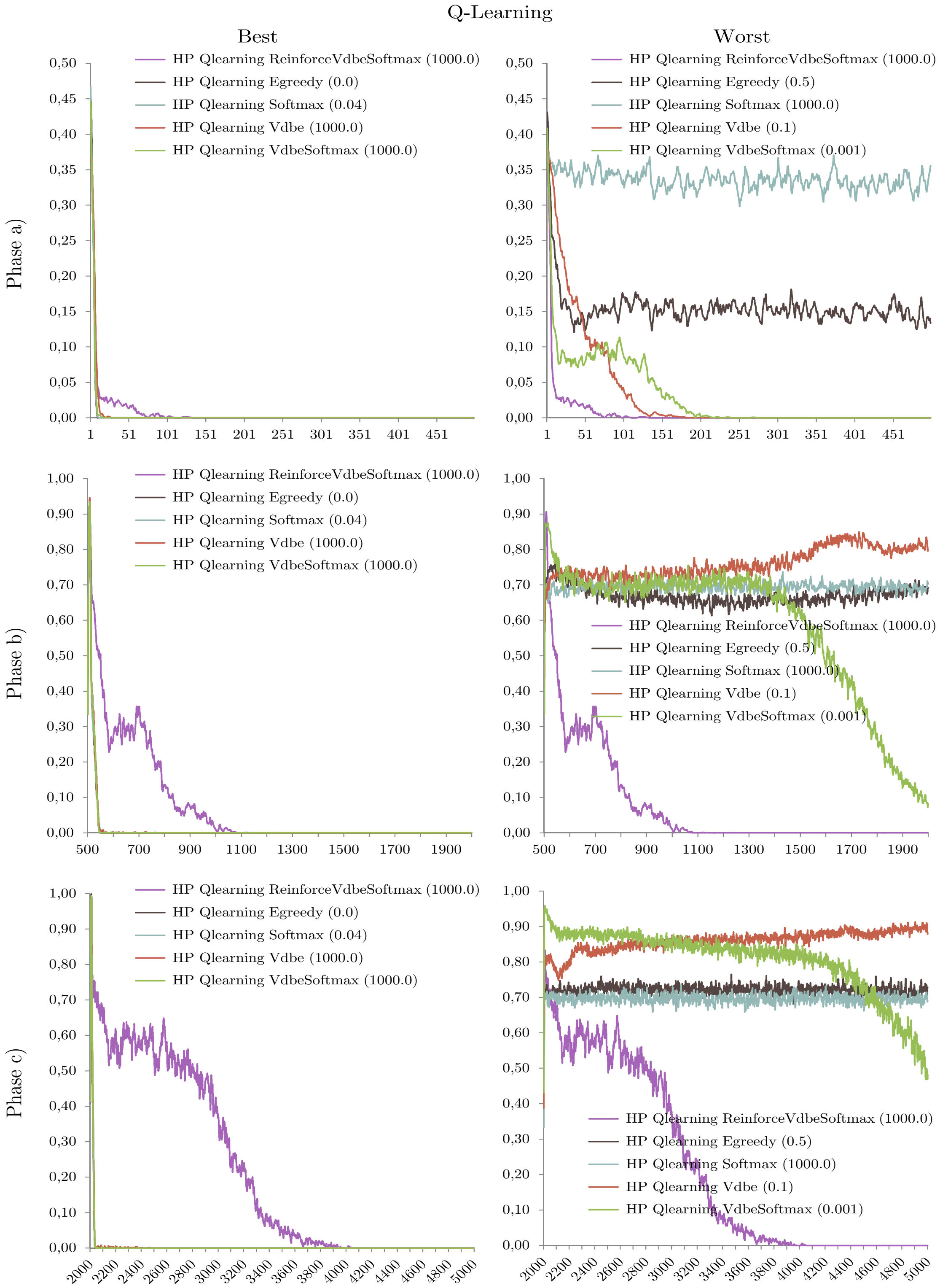


Figure 9.5: Hazard performance of the Q-learning algorithm in combination with different exploration-exploitation strategies with best-case and worst-case parameter settings. The abscissa comprises the number of learning episodes, the ordinate the hazard probability measure. The distinct phases are vertically arranged, the respective best/worst parameter setting is horizontally arranged. The best-/worst-case parameters ($\varepsilon : [0,1]; \tau : [0.001, 1000]; \sigma : [0.001, 1000]$) are denoted in brackets.

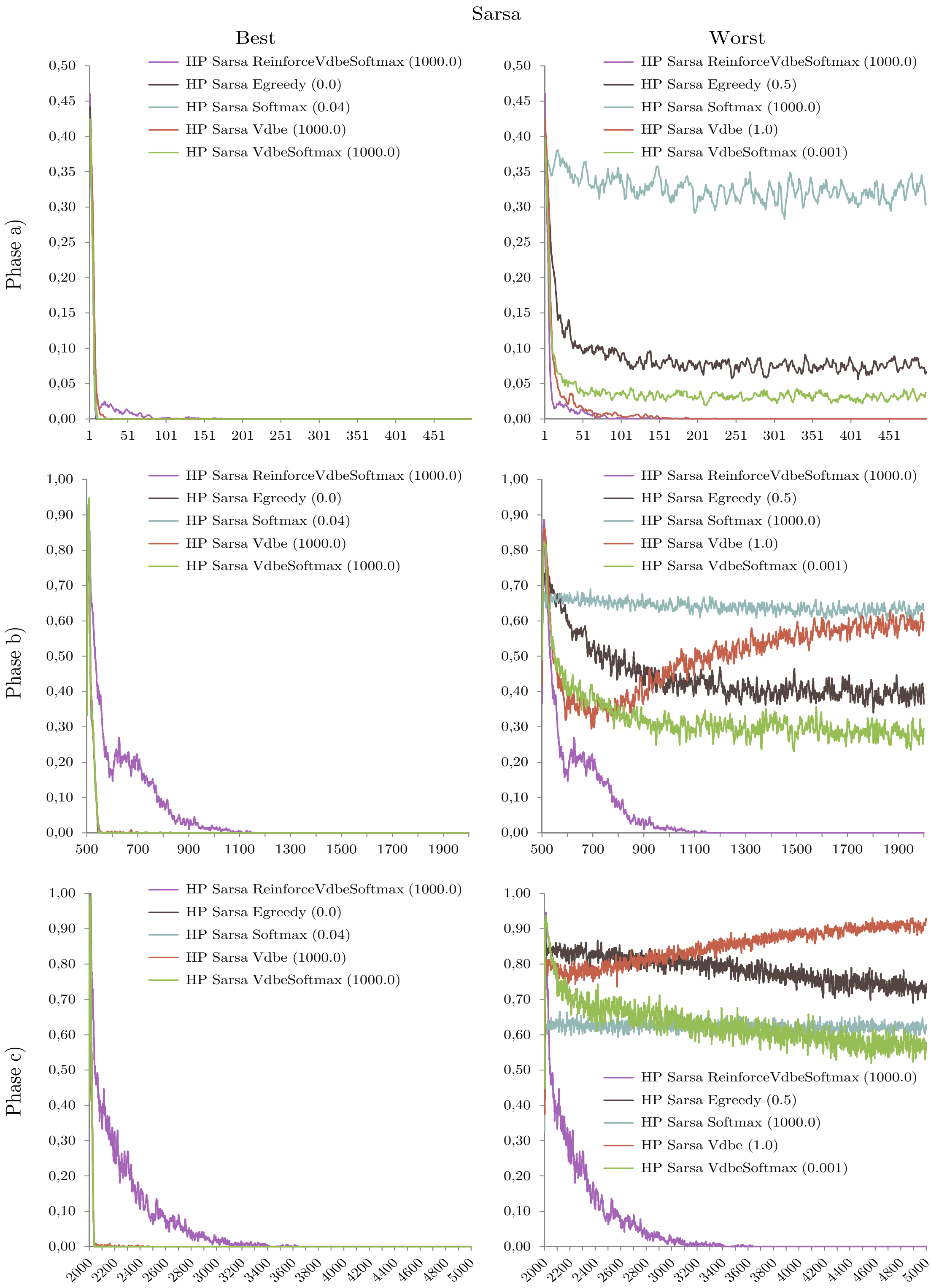


Figure 9.6: Hazard performance of different Sarsa-learning algorithms in combination with different exploration-exploitation strategies with best-case and worst-case parameter settings. The distinct phases are vertically arranged, the respective best/worst parameter setting is horizontally arranged. The best-/worst-case parameters ($\varepsilon : [0,1]; \tau : [0.001, 1000]; \sigma : [0.001, 1000]$) are denoted in brackets.

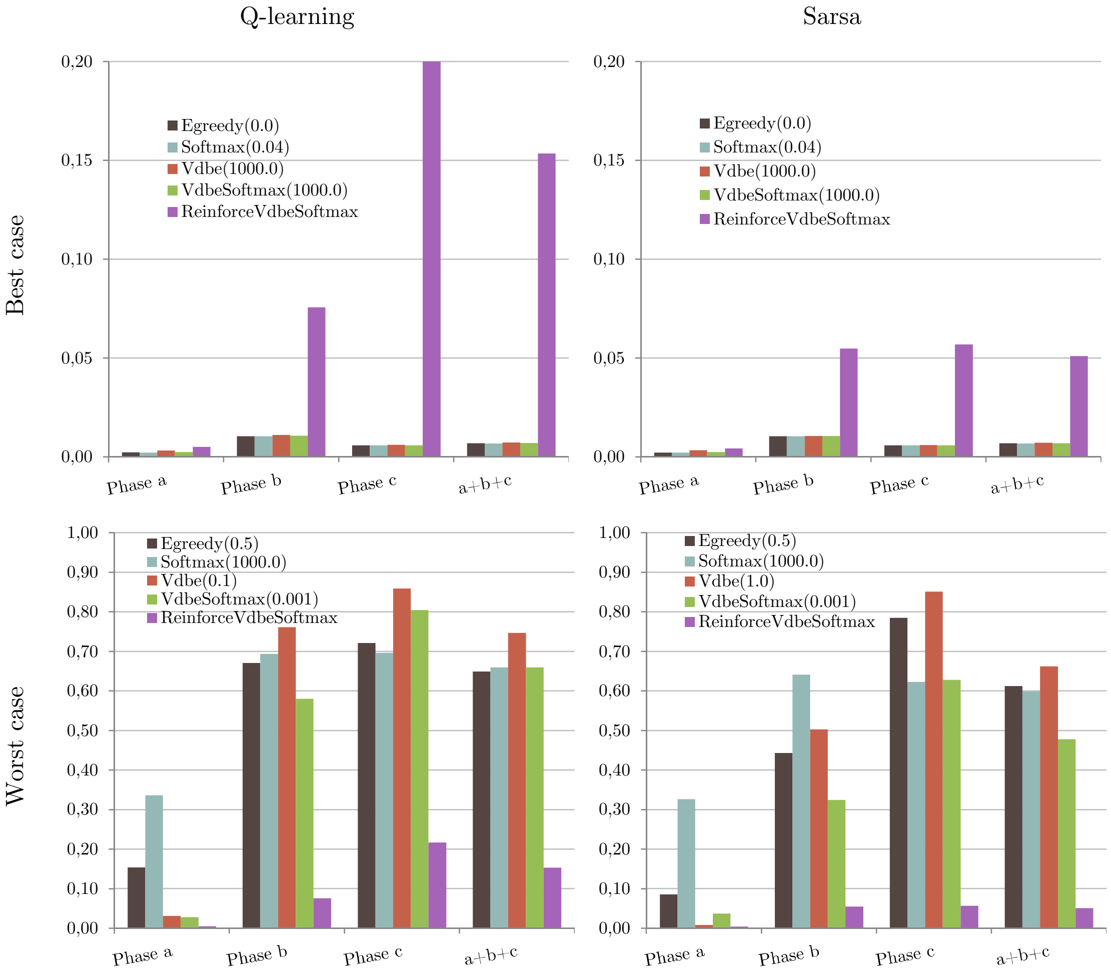


Figure 9.7: Hazard performance (ordinate) of Q-learning, and Sarsa-learning in combination with different exploration and exploitation strategies averaged with regard to the single phases a), b), c), and averaged over all phases (abscissa) of the non-stationary cliff-walking scenario with absorbing cliff states.

9.3.5 Conclusions

In general, exploration/exploitation strategies that combine Softmax and Greedy behaviors perform with a lower hazard performance than applying them directly. This is due to VDBE-Softmax strategies do not select exploration actions equally distributed but in case of fluctuating values exploration actions are selected value sensitively according to Softmax, and in case the value function has converged it acts greedily (Tokic et al., 2012). Practically, it seems that both reinforcement learning approaches in combination with the mentioned exploration/exploitation strategies lack the required plasticity with regard to an altered operation environment, and inadequately chosen parameters, except Reinforce VDBE-Softmax.

Since all constellations of investigated algorithms require entering hazardous states multiple times before the safe behavior is leaned, it is obvious that they should not be applied in safety-critical applications without taking into account this issue. However, it is notable that many of the considered constellations perform significantly better than a randomly walking agent. From this, it can concluded that if an agent in a potential hazardous environment lacks any strategy to act (due to unknown situations, lack of specification, miss-design, specified fail-safe state turns out to be inadequate, etc.), it would be theoretically better to equip the agent with learning capabilities, for instance, Reinforce VDBE-Softmax. However, the information has to be provided to the agent that a situation is undesired or even hazardous. Possibly, this could be realized if hitting the emergency button gives as well a high negative reward.

9.4 Learning Safety from Demonstration for Robot Skills

So far, rarely literature is available focusing on learning from demonstration for safety-related concerns. Kuter et al. (2007), for instance, describe the learning of safety constraints. Here, a domain ontology is developed in which domain concepts define sets of entities in the world, belonging together, and sharing some properties. Their so-called Constraint Learner procedure checks for demonstrations of specific properties `p` from which new lower or upper bounds can be observed, `if upper_bound(p)<value_observed(p) then upper_bound(p)=value_observed(p)`, and the like for lower bounds.

An approach for learning spatial constraints is given by Gips et al. (1998). A decision tree is learned from labeled data. In a theoretic example it is learned how a rectangle is located at a specific (at right-hand side of the observer) spatial position. The learning of safety constraints is not explicitly mentioned. Consequently, so far no approach exists to infer safety measures for robots by a learning from demonstration approach.

Robot skills

The majority of robotic architectures are realized with three layers. At the lowest level, skills are managed by the superordinate layers (cf. Schlegel and Wörz, 1999). These skills are combined to more complex behaviors. The behaviors do not need representational knowledge Arkin (1998); thus, they can basically be regarded as structurally invariants during the operation phase. Moreover, this is also valid for cognitive-oriented robotic architectures, for instance, described by Ahle (2007).

In order to learn complex skills, the second field of learning approaches, the learning from demonstration plays an important role. Here, behavior cloning approaches are influential, in which the robot directly derives a policy from a teacher's demonstration in order to apply it for reproducing the task (Sammut and Webb, 2010). When robots shall learn from demonstrations, the predominant questions are: What about safety? Are safety aspects already comprised in the demonstration?

Usually, these approaches lack explaining why an applied policy is a good or bad one (Sammut and Webb, 2010), which implicates that they lack recognizing safety relevant aspects. Safety might be implicitly considered, but it is definitely not systematically taken into account. With regard to pick and place tasks, this becomes clear. On the one hand, new learned skills might be observed and limited by already known safety rules. But new skills may be accompanied by new hazards not considered so far. For instance, as outlined in the latter chapters, various new hazards may arise with regard to object interactions. If the robot learns to manipulate objects (pick and place), it is questionable for good reason whether the knowledge to safely handle objects is also appropriately learned. Hence, if robot should be taught to become more skilled, the general problem is to teach them how new skills are safely performed. It is assumed that an additional learning step is required therefore.

As outlined at the beginning of this section, for learning during the operation phase should be paid more attention, especially, if this takes place under supervision of unqualified users. Thus, the proposed procedural model represents an approach for learning safety knowledge in order to simplify the safety engineering process. The essential aspect is that the learned safety knowledge can (and has to) be checked and revised after the learning process and before the system is finally put into operation. However, the basic concept can also be applied to learn (post-design) during the operating time in order to try to consider residual hazards.

9.4.1 The extended safety procedure

As initially mentioned, the general system safety process is integrated in the development process and takes place throughout the complete life cycle of a system. The general steps take place in a cyclic manner: Hazard identification, hazard risk assessment, risk control, and risk verification. The risk control is sufficient since all risks are mitigated to an acceptable level (Ericson, 2005).

With regard to autonomous systems, it is suggested (in the field of systems of systems) to structure the hazard analysis with regard the capabilities a system provides: *"Each capability will present a variety of possible hazards, stemming from a failure to provide the capability, an incorrect implementation of the capability, or from unexpected side-effects of employing the capability"* (Alexander et al., 2008). This is in line with the standard ISO/NP 13482 for robots in (non-medical) personal care (see Section 2.1), as reported by Harper and Virk (2010), where a list of tasks needs to be identified and specified with regard to functional and non-functional requirements.

The proposed approach is to modify the system safety process so that risks are reduced by learned countermeasures (safety functions). In the first instance, safety functions basically label perceived situations as either risky or not. How far the generated labels are utilized for 'limiting' the system depends on a decision process which should take into account as well the benefits of a task. Te binary labels ('safe', 'risky') can be transformed into risk values if the overall risk of the top level hazard is determined with regard to the specific task. The modified safety process is shown in **Figure 9.8.** At first, hazards, which are related to a specific behavior, are required to be identified and assessed. This is according to the conventional procedure for generally determining the hazards which are required to be mitigated. Four steps are identified in order to control the identified and unacceptable risks: The definition of the requirements for the demonstration, the demonstration itself, the revision of the learned safety functions, and

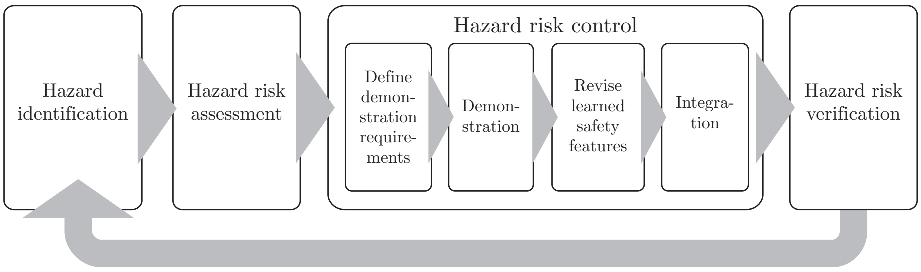


Figure 9.8: System safety process for learning safety features from demonstration (Ertle et al., 2012b).

finally, their integration into the robot control system. The product of the demonstration task finally is either a learned or a learned and refined safety function. The safety function labels the perceived (measured) situation, which can be in turn considered by a subsequent decision or planning process (integration).

Demonstration requirements

The definition of the requirements plays a central role. First, required data must be determined. If relevant data is not provided to the learning approach, the learning problem can not be classified. If relevant aspects can not be measured or derived from prior system knowledge, the demonstration approach is consequently not reasonable. Secondly, demonstrated data has to be labeled. An adequate user interface must be provided. Thirdly, the identified hazards have to be demonstrated without provoking accidents. For the most cases, it is assumed that it is possible to demonstrate an approaching toward a hazard, in terms of the hazard is not endangered to be actuated. The limitations learned in consequence, are still in a safe range; therefore, they include a specific safety clearance. The indicating of a hazard plays a central role. Therefore, the demonstration sequence must be precisely defined. In order to make the safety learning problem classifiable, sufficient desired and undesired constellations must be demonstrated. Consequently, aspects that are not comprised in the demonstration can not be learned. The hazards must be ‘encircled’ by the indications of the undesired approaching so that it becomes clear which area (hyper space) is undesired. The problem is illustrated in **Figure 9.9.**

Demonstration

During the demonstration it must be ensured that the demonstration specifications are maintained. Additionally, it is important to be as precise as possible when indicating undesired/risky situations. Otherwise, too many ambiguities may let the learning problem become unclassifiable. Several repetitions of a demonstration might be necessary. In this regard, classification quality measures could be used to guide the demonstration process.

Revise Safety Features

Since there are many factors that contribute to good or bad learning results, an intensive

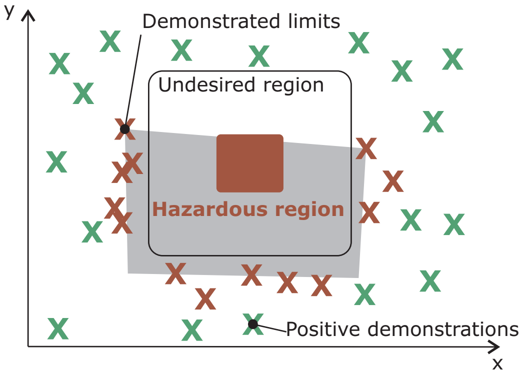


Figure 9.9: Need for carefully planned demonstrations in order to ensure sufficient representation of features, according to Ertle et al. (2012b).

revising of the learning success is assumed to be mandatory, at least for severe hazards. In this case, the learning from demonstration approach can be seen as a design aid for generating safety functions. The learned safety functions have to be additionally checked. Furthermore, they have to be available in a readable form what is either given (learning of decision trees) or what could be realized by additional transformation steps, for instance, via extraction of rules from neural networks (Hruschka and Ebecken, 2006), or from reinforcement learning functions (Vogiatzis and Stafylopatis, 2002). Nevertheless, a learning from demonstration approach remains helpful and may reduce faults. As shown in the following experiments, the considering of spatial relations between objects in 3D-space often requires a set of coordinate transformations between different reference frames, what sometimes is error-prone.

The revision step can either take place via safety engineers if the learning from demonstration approach is used prior to the operation time, or it applied during the operating time in order to consider arising residual hazards. In this connection, either the revision step is rejected at all, or automated revision methods are developed and integrated.

Integration

The outcome of the learned safety function depends on the applied learning algorithm. In general, the outcome might consist of binary, any continuous or probabilistic prediction values. In case the severity S of an accident A is numerically expressed, a risk value $R = P(A) \cdot S(A)$ can be computed and assigned to the labels. The expression of hazards in risk measures is important because it can be considered in a subsequent decision process, as outlined in Chapter 8. In this connection, the decision making is equipped with the additional risk information in order to be balanced with other costs and benefits for finding out the optimal strategy. The proper consideration of risks, thus, is part of the decision process, whose testing and verification can take place separately without taking into consideration to correctness of the learned safety functions.

9.4.2 Object interaction risk experiment

In order to test the described approach, two different scenarios were chosen. It is assumed that for each scenario a specific robotic behavior would be required. Each scenario comprises simple hazards to be learned. The goal is to learn a safety function for each behavior, which can be applied for detecting unsafe/undesired situations.

The approach is applied considering real world conditions with noisy sensor data. Furthermore, those risks are focused which appear when environment and robot manipulated objects interact with each other. These objects are represented by markers, whose position and orientation are detected by the ARToolKit.[3] The camera for observing the scene was mounted at the demonstrator's head in order to simulate the consistently different positions a robot will be relatively located in real world applications. The data is labeled as risky if a key on the keyboard is hit (e.g. representing an emergency button or voice command module).

The demonstration videos are first captured and stored. Afterwards, the videos are presented to a *C++* program utilizing the ARToolKit library in order to store the recognized position and pose data vectors. Finally, the stored data is analyzed. As outlined by Ertle et al. (2012b), the data-mining tool KNIME[4] in combination with Weka[5] extensions are utilized for extracting functional representations of the demonstrated safety knowledge. Three supervised learning techniques are evaluated: 'Radial Basis Function Network' (RBFN), J48 'Decision Tree' (DT), and 'Multilayer Perceptron' (MLP). Each of these classifiers is manually hand-tuned, according to **Figure 9.10,** for obtaining reasonable results.

The performance of these classifiers is measured with the recall and F1-measure (positive means indication of risk, true/false positive/negatives being tp, fp, tn, fn),

$$\text{Recall} = \frac{tp}{tp+fn} , \tag{9.2}$$

$$\text{F1} = \frac{2 \cdot tp}{tp+fn+tp+fp} . \tag{9.3}$$

The recall reflects the sensitivity with regard to correct classification of risks. The F1-measure represents the overall accuracy.

9.4.3 Results of the ironing task

The first scenario is the ironing scenario. In this task, the hazard of fire is potentially comprised when an iron remains too long at the same position. Therefore, it is demonstrated that the standstill of the iron on the ironing board is not desired. The placing of the iron at its backside (in upright position) or in the provided deposit is uncritical. The scene is shown in **Figure 9.11.**

[3] `http://www.hitl.washington.edu/artoolkit/` [online; accessed 17-March-2013]

[4] `http://www.knime.org` [online; accessed 17-March-2013]

[5] `http://www.cs.waikato.ac.nz/ml/weka/` [online; accessed 17-March-2013]

	Parameter	Ironing	Stacking
RBFN	Clustering seed	1	1
	MaxIts	unl.	unl.
	MinStdDev	0,1	0,1
	NumClusters	20	1
	Ridge	1,00E-08	1,00E-08
MLP	Data	normalized	normalized
	Max.Num.Iterations	1000	1000
	Hidden layer	2	2
	Neurons per hid. layer	10	10
J48 DT	Conf. Factor	1	1
	MinNumObj	100	62
	NumFold	3	3
	Seed	1	1
	SubtreeRiasing	true	true
	Pruning	true	true
	Use laplace	false	false

Figure 9.10: Parameterization of the used learning algorithms (cf. Ertle et al., 2012b).

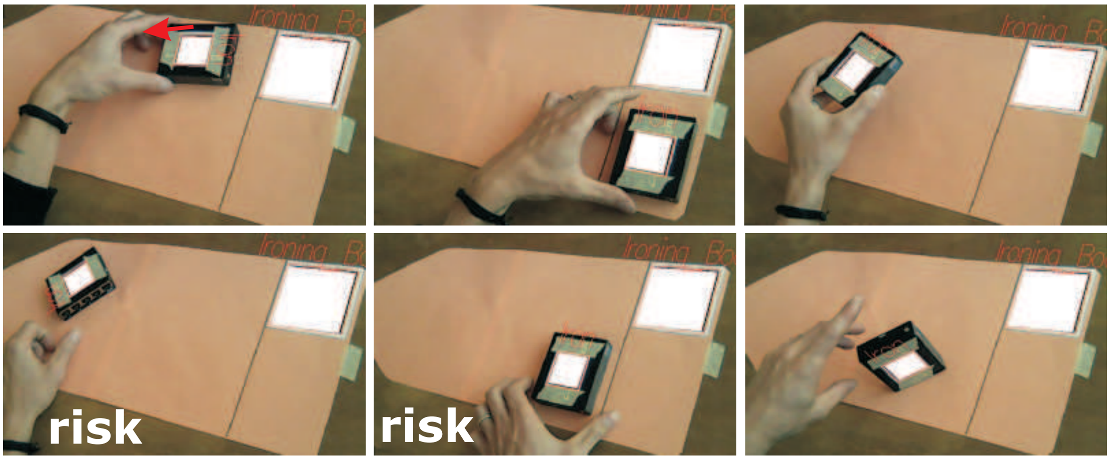


Figure 9.11: The ironing task: Pictures extracted from the demonstration video captured from changing positions. The object positions and poses are recognized with ARToolKit. The labeled pictures show *risk* situations (stand-still of the iron). The others show *normal* situations (iron positioned at the iron storage, iron turned at the back side or iron moved over the surface) (Ertle et al., 2012b).

The relevant data for the safety function to be learned is measured. These consist of the relative positions x,y,z and angles α,β,γ from the coordinate system of 'object 0' (ironing board) and 'object 1' (iron), respectively. Additionally, the absolute values of the relative velocities v are computed. Observed data are transformed into vector form with predefined order. Thus, the situation input vector s of a typical ironing scene consists of the following 14 continuous inputs, as

$$s = [x_0^1, y_0^1, z_0^1, v_0^1, \alpha_0^1, \beta_0^1, \gamma_0^1, x_1^0, y_1^0, z_1^0, v_1^0, \alpha_1^0, \beta_1^0, \gamma_1^0] \ . \tag{9.4}$$

Data points are labeled as commanded by the user. Initially, 'normal' operation is assumed, which is the default label for all data vectors. In case hazardous situations are demonstrated, data is labeled to 'risk' as long as the emergency button is pressed.

In total three demonstrations have been presented with a total of 3633 training-data vectors, where the duration of each demonstration varies between 114 to 142 seconds. These comprise 2909 vectors (80%) belonging to the 'normal' class, and 724 vectors (20%) to the 'risk' class. A fourth demonstration is used as test data consisting of 1100 vectors, 900 vectors (81%) belonging to the *normal* class, and 200 vectors (19%) to the 'risk' class.

The diagrams in **Figure 9.12** show the classifications of the test data according to the trained classifiers. The classification recall and F1 performance of the overall task is shown in **Figure 9.16** (upper). The parameters are computed for classification of the test data after learning of 1, 2, and 3 demonstrations. In the last demonstration, 4*, the test data is additionally learned. The performance of the complete demonstration scenario is computed with regard to the same test data.

The generated decision tree reflects the effect of the respective measurements well comprehensible: Sufficient velocity is desired, via relative x-position the deposit of the ironing board is detected, via the γ-angle it is detected whether the iron is turned upright, and if the iron is above the ironing board there is no risk as well. The branch for angle α denotes a fragment. It is notable that the relative x-position is considered from the perspective of the ironing board (object 0) and γ-angle from perspective of the iron (object 1). Otherwise neither the position at the iron storage nor the depositing of the iron at its backside could be easily distinguished.

9.4.4 Results of the stack-it-safely task

Inspired by the 'Safe-To-Stack' approach by Mitchell et al. (1986), a scenario is designed for learning the safe stacking of two objects. The hazard identified for this scenario is the toggling and falling down of the narrow *object* 1 that is stacked upwards on 'object 0'. The flat stacking is assumed to be uncritical. Indeed, modeling of spatial relations as 'on, upwards' and so forth may simplify the problem description, but finally, these relations also rely on similar relative coordinates as they were utilized in the demonstration. A flag whether an object is gripped or released is not considered in the scenario, the relative velocities of the objects becoming zero indicate a similar

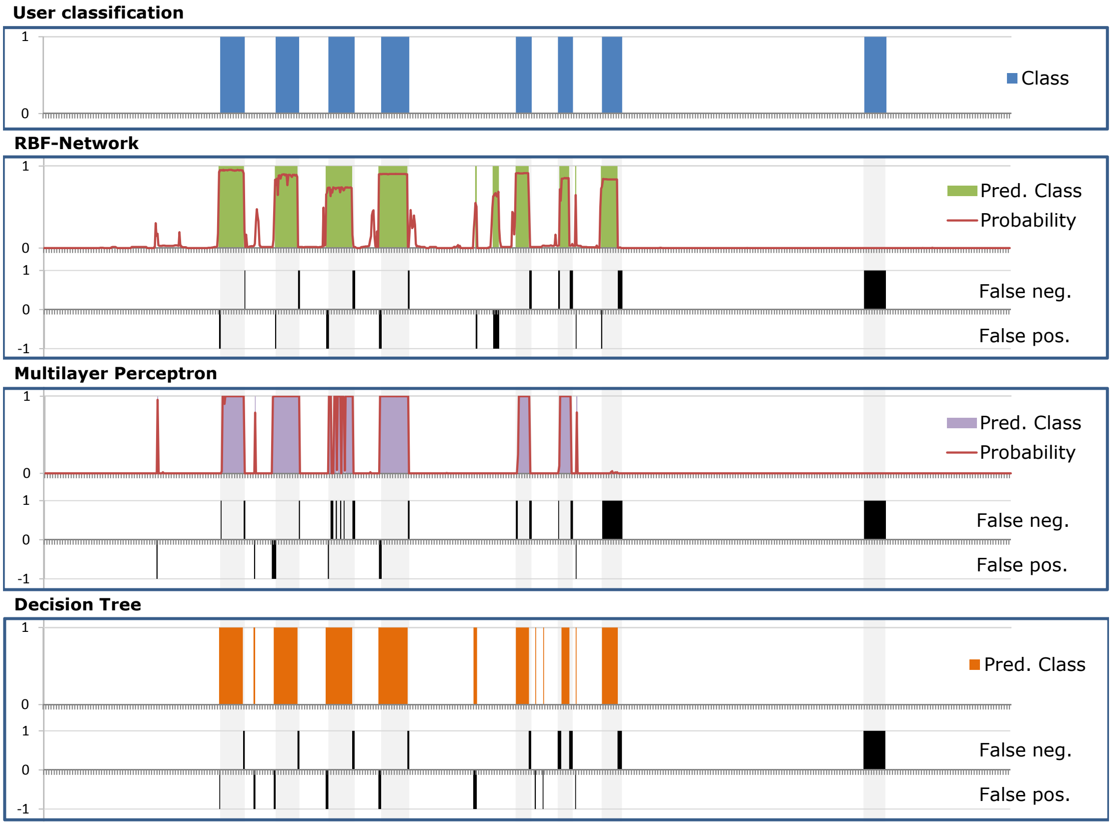


Figure 9.12: Time plot showing the classification results of the experiment test data for the ironing task (Ertle et al., 2012b). The section shows all classification failures of the experiment. The first diagram indicates when the user has demonstrated risks (1 corresponds to risk/positive). The following double diagrams each illustrate the results of the respective learning method. The first of the double diagrams shows when risks are predicted, the second illustrates when fn (1), correct prediction (0), and fp (-1) occur. The curve in the RBFN and MLP diagrams represent the predicted probability for risk, respectively.

```
$v_0^1$ ≤ 2.4362
|   $x_0^1$ ≤ -93.2441
|   |   $\gamma_1^0$ ≤ 21.063
|   |   |   $z_1^0$ ≤ -39.851
|   |   |   |   $\alpha_1^0$ ≤ 54.86 : normal (128.0 / 46.0)
|   |   |   |   $\alpha_1^0$ > 54.86 : risk (182.0 / 41.0)
|   |   |   $z_1^0$ > -39.851 : risk (453.0 / 28.0)
|   |   $\gamma_1^0$ > 21.063 : normal (211.0)
|   $x_0^1$ > -93.2441 : normal (397.0)
$v_0^1$ > 2.4362 : normal (2262.0 / 112.0)
```

Figure 9.13: Learned decision tree for the ironing task (v in [cm/s]; x, y, z in [mm]; α, β, γ in [°]) (Ertle et al., 2012b).

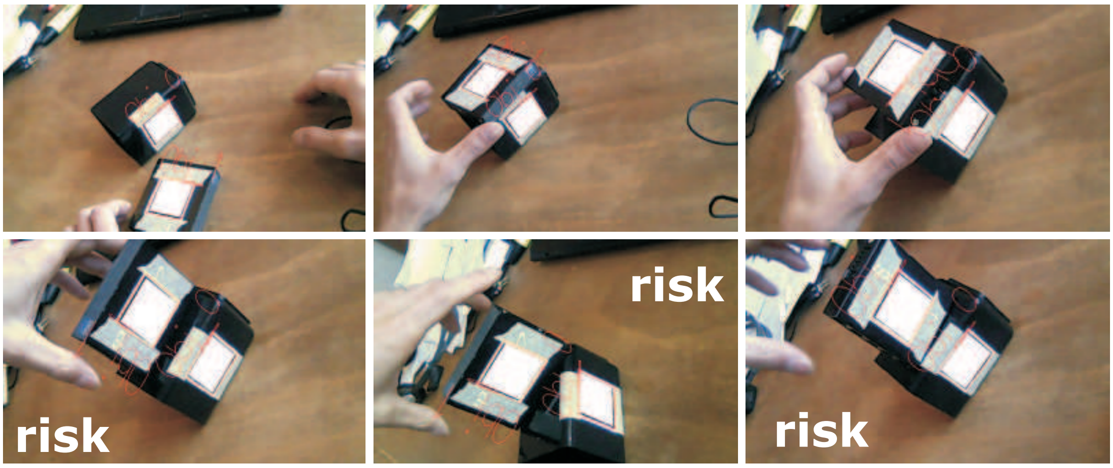


Figure 9.14: The stack-it-safely task: The labeled pictures show *risk* situations (upward stacking of the narrow object). The others show *normal* situations (stacking objects located somewhere else or in correct position at the top of the other object) (Ertle et al., 2012b).

state as the release of a gripped object. The vector remains the same as described in Section 9.4.3. The scene is shown in **Figure 9.14.**

In total three demonstrations have been made as well, with a total amount of 3184 training-data vectors, where the duration of each demonstration varies between 114 to 145 seconds. These comprise 2452 vectors (77%) belonging to the *normal* class, and 732 vectors (23%) to the *risk* class. A fourth demonstration is used as test data consisting of 772 vectors, 666 vectors (86%) are belonging to the *normal* class, and 106 vectors (14%) to the *risk* class. The diagrams in **Figure 9.15** show the classifications of the test data according to the trained classifiers. The classification recall and F1 performance of the overall task is shown in **Figure 9.16** (lower). The parameters are computed for classification of the test data after learning of 1, 2, and 3 demonstrations. In the last demonstration, 4*, the test data is additionally learned as well.

9.4.5 Conclusions

The experiments show that the 'decision tree'-algorithm appears to provide good performance to learn safety functions in object manipulation scenarios besides its advantage to be readable, and at need, also adjustable by safety engineers. The results indicate that the proposed safety procedure provides a significant potential for supporting the safety engineering process. Since the quality and completeness of the learned hazard model significantly relies on the demonstration quality, it is questionable to apply this concept to post-design phases. For safety-critical application with severe consequences, this concept should be limited to the application during the design phase of the system, and combined with verification procedures. For behaviors that have to be taught during the operating phase, it might be pondered to apply such a learning concept for tasks with minor consequences. Principally, it holds the same for this learning from demonstration approach as for the reinforcement learning investigation: In

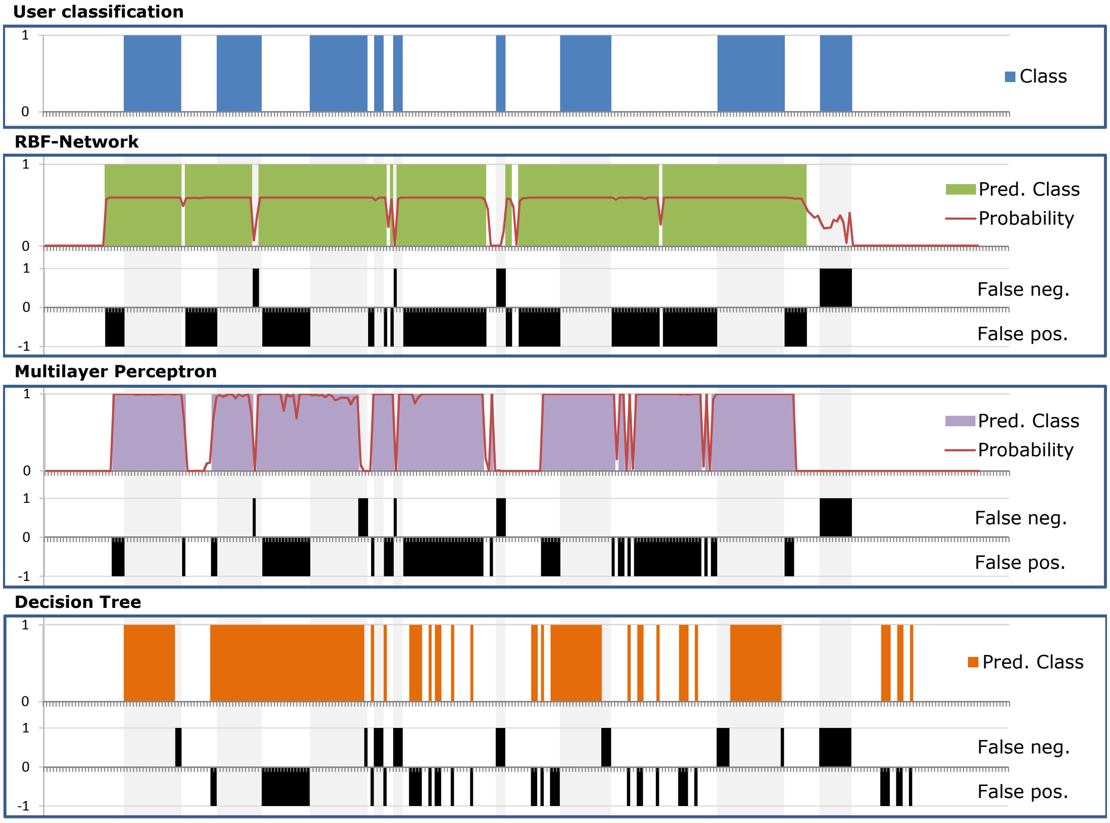


Figure 9.15: Time plot showing the classification results of the experiment test data for the stack-it-safely task (Ertle et al., 2012b). For explanation, see **Figure 9.12.**

Nr.	F1			Recall		
	RBFN	MLP	DT	RBFN	MLP	DT
			Inoning task			
1	**0,76**	0,29	0,73	0,74	0,80	**0,88**
2	0,72	0,70	**0,75**	0,84	0,78	**0,86**
3	**0,82**	0,77	0,80	**0,88**	0,79	0,85
4*	0,80	0,70	**0,88**	0,74	0,74	**0,90**
			Stack-it-safely			
1	**0,60**	0,19	0,54	0,47	0,19	**0,64**
2	0,58	0,54	**0,75**	0,45	**0,76**	**0,75**
3	0,58	**0,73**	0,66	0,44	**0,62**	**0,61**
4*	0,61	**0,68**	**0,67**	0,45	0,53	**0,63**

Figure 9.16: The classification performance in terms of the F1 and recall measure. The performance is measured against the test data after 1, 2, and 3 demonstrations. *The test data are additionally integrated into the learning process in a fourth step, the performance is measured according to the same test data. Best F1/recall measure is highlighted, respectively (Ertle et al., 2012b).

case of appearing unknown situations, lack of specification, miss-design, or the specified fail-safe state turns out to be inadequate, learning capabilities can also be seen as a chance to enable the robot to overcome a rigid unsafe behavior.

9.5 Concluding Remarks

At first, it can be concluded that learning can in principle be permitted from the safety perspective, for simulations, and laboratory conditions if the hazards occurrence and consequences are under supervision of qualified persons. Furthermore, it can be permitted if it takes place during the development phase, and iff it can be shown afterwards that the learned aspects can not lead to hazardous situations during the system's operating time. Because it might be difficult or impossible to prove the safety, it can be deliberated if systematic and intensive testing is sufficient as proof for safe operation.

Secondly, it can be stated that learning under influence of unqualified users remains critical, since the unqualified users may lack estimating the consequences of the learned aspects, on the one hand, and on the other hand, the intentions of the users are unknown.

Thirdly, learning capabilities of robots can be permitted in general, iff it can be ensured that hazards are known by the robot system, and their control is designed such that hazardous situations are sufficiently avoided. The concept to equip the robotic system with the knowledge about hazards is pursued by the safety-knowledge-based dynamic risk-assessment approach outlined in Chapter 8. In this regard, the dynamic risk assessment process can be considered as a basis for robot safety. Learning approaches might be integrated within the engineering process of the safety knowledge, or may operate complementary in order to minimize the residual lack of safety knowledge. Consequently, a systematically engineered system safety process during the design phase, and during the operating phase via safety knowledge updates, can not be replaced by learning approaches, but learning can be used to improve the safety of an already comparably safe system, for instance, realized with a concept as proposed in Chapter 7.

10 Summary and Future Work

This thesis draws attention to the safety problem of autonomous robots. A procedural model to systemically generate the initial safety knowledge, required for realizing the dynamic risk assessment approach is developed and detailed. Particularly, it is drawn attention to generate the safety knowledge of the so far not considered hazardous object-object interactions. Furthermore, learning approaches are focused against the background of the potential lack of safety-related knowledge. The resulting main achievements and related benefits are summarized in the next section. Additionally, further steps and new ideas, which were identified in conjunction with this thesis, are briefly described in the final section of this chapter.

10.1 Summary

The contributions to robotic safety applications are spread over a wide spectrum of research fields due to the interdisciplinary character of robotic research. As interdisciplinary as the robotics research field is, the diverse are the contributions to safety-related topics. Hence, a thorough review of present robotic and cognitive paradigms is from particular interest in order to point out the present termini and perception of these topics. Furthermore, it is spent particular effort structuring the extensive literature review in a meaningful way.

It is observed that there is available a significant variety of research work focusing collision safety with humans and obstacles for both, robotic hardware design and reactive behaviors. In this connection, the robot's kinetic energy, emanating from the robotic system itself is considered as hazardous energy. The attitude to perceive the hazardous energy source as a part of the system is typical for the majority of the research work. Not later than the robots become capable to grasp and manipulate objects, further hazard origins have to be taken into account. A taxonomy for hazard origins is one of the essential achievements of this work. A group of hazards with notable harm potential is formed by hazardous energy sources that are located in the robot environment. The robot itself is primarily involved in turning the hazards into a mishap or accident by manipulating objects that interact with respective environment objects. This kind of hazards was so far neither perceived in present research work, nor is it considered in relevant standards.

The autonomy feature, often touched on and also often tried to be realized in mobile robotics, is investigated in detail as one point worthy to point out in this contribution. Especially, the impact of machine autonomy on safety aspects is explicated in detail. Here, it is derived that the desired degree of autonomy inherently requires the capability to learn. Additionally Freud's structure model of the human psyche is identified as a useful blueprint to approach toward a solution of the autonomy-safety problem.

If robots can not be ruled out to be such safety-critical systems, it is obvious from the safety perspective that safety measures have to be implemented before the system is put into operation. It is also obvious that robots can not solely learn to safely behave because this implicates that hazardous situations have to (quasi) occur in order to learn from them. However, knowledge about hazards at the design stage has to be assumed to be potentially incomplete for operating environments of high complexity. In consequence, it seems that both, a good safety engineering practice, and learning approaches have to be applied in order to achieve a convergence toward safety. Therefore, the system safety process is understood as an active iterative process that reaches as well into the operation phase. Hence, it is part of the overall concept of this work to realize an adjustable safety knowledge base on-board of the robotic system that is applicable and expandable for upcoming new situations during the operating time.

The application of the safety knowledge is intended to take effect within the so-called dynamic risk assessment approach which aims on enabling the robotic system itself to assessing the risks of upcoming situations. The utilization of gradual risk descriptions is essential because a binary classification of risks (as safe or not) entails problems. The situation awareness of the system itself plays as well a central role in this connection, as it denotes the capability to perceive and comprehend the environment, and to foresee situations in the near future. In this regard, it is firstly described an approach which realizes situation risk-awareness on basis of a cognitive architecture and a knowledge-based situation risk assessment function. Therefore, the hazards are formalized as safety knowledge in form of so-called 'Safety Principles'. A Safety Principle has a dyadic structure and denotes a model to formalizing hazards: At first, it comprises information for detecting the presence of a hazard, and secondly, instructions for computing its respective numeric risk value. In general, other concepts such as adaptive collision avoidance strategies, adaptive compliant actuation or injury knowledge-based control etc. can be connected to context-awareness by Safety Principles as meta-structure.

The underlying cognitive architecture providing cognitive functions such as perception, learning, planning and anticipation is based on the so-called Situation-Operator-Modeling (SOM) approach. The SOM approach is used as meta-modeling technique for structuring and formalizing complex environments of systems. The safety knowledge and the risk assessment server as well are conceptualized with the SOM notation; hence, the both can be integrated within the cognitive architecture. For this reason, the capabilities of the cognitive architecture can be fused with the dynamic risk assessment approach, or rather risk awareness can be realized within the cognitive architecture, since risks become perceivable for the system. Consequently, the outlined concept provides the capability to perceive and anticipate risks in order to basically realize risk-sensitive planning. Additionally, it is introduced the concept of a safety clearance into the risk-sensitive planning-approach; hence, it is already avoided to get closer to hazardous situations. Since the safety knowledge is realized via SOM notation, it remains to be practically changeable by the system itself which structurally enables the extension and refinement of the safety knowledge base during the operating time of the robot, for instance, via learning.

As already mentioned, it is problematic to apply learning approaches for safety-critical applications. Therefore, a separate perspective on this topic is given. In this regard, a non-stationary scenario, comprising different hazardous states is designed in order to evaluate reinforcement learning. In the first instance, a hazard performance measure is defined which also takes into account the hazard potential of the scenario. Hence, the count of transitions into hazardous states relative to the scenario-hazard potential is evaluated (comparison to a randomly walking agent). It is found that learning capabilities can as well provide significant benefit for worst-case conditions, in which the algorithms are sub-optimally parameterized in situations without adequate fail-safe strategies.

In the following, a further learning paradigm is investigated - imitation learning, or more precisely, learning from demonstration. In principle, learning from demonstration can be applied for extending the safety knowledge during the operating time of the robot. However, this can raise multiple issues, since the validation of the learned safety knowledge is difficult to accomplish. Thus, a learning from demonstration procedure is described which is integrated into the system safety process in order to support the generation process of risk models. The risk models can be integrated into the dynamic risk assessment approach when the verification of the learned contents took place. Notably, the utilization of decision trees shows interesting results, besides the advantage that the decision trees are directly readable.

10.2 Future Work

This work was focused on the realization of a risk-aware autonomous system. Since little research work was realized in this area, this work represents a first approach toward a novel topic and, hence, many directions of future work have to be pursued. In the following, some points are mentioned which are considered to be from major interest.

It is described within the realized concept how risks can be formalized in order to enable the system itself to recognize and estimate them. During the execution of reactive (low-level) skills, the higher system level may be involved in planning further steps in order to approach toward a goal. But if the performing of a skill is risky (potentially close to an acceptable risk threshold), the higher system levels may be required to track the execution, in terms of draw special attention to the progression of the current action's outcome. Thus, the question arises how the resources of the higher systemic levels should be controlled and distributed. Therefore, a control may be required which deliberates the assignment of the high-level resources. Resources may be gradually distributed among precise short-term planning and monitoring and long-term planning.

Furthermore, the system is aware of risks at higher system levels. The knowledge about the risks should as well have effect on the execution layer in order to enable risk-sensitive skill execution without special attention of the higher system level. For instance, specific collision risks may be formulated as a Safety Principle; thus, the system is aware of it at higher system level, but not the movement controller. In consequence, the awareness about risks is an important aspect but often depends on complex world knowledge.

According to the hybrid paradigm in robotics, complex world knowledge-related aspects are typically processed at higher system levels, and have to become rare at lower levels in order to maintain reactivity. Thus, the question arises how the execution of skills under risky conditions should be parameterized or modulated in order to take into account the risks the system is aware at higher system levels. In this connection, the risk information has to be somehow decomposed into adequate parameters, high-performance functions or code segments to be processed by the execution of low-level skills. If the skills itself are considered to be atomic units at some point, the information of risks has finally to be mapped into adequate execution parameters.

In addition, the handling of multiple risks in complex situations is perceived as challenging problem. The outlined risk planning strategy is logic and analytic. Humans typically operate differently in complex situations, not analytically but intuitively according to an experience-involving 'affect heuristic' (Finucane et al., 2000). In this regard, two fundamental systems are involved, which again need for a third, being altogether 1) the 'analytic system', which uses algorithms and normative rules, 2) the 'experiential system', which relies on experiences based on images or associations and 3) the 'political system', which mediates the two latter system's outcome (Slovic et al., 2004). For the judgment in complex decision or decision with limited resources (e.g. time pressure), the affect heuristics can be more effective than to analytically deliberate the pros and cons. Especially, decisions in the scope of risks and benefits are made with regard to emotions stemming from former experiences (Finucane et al., 2000). The utilization of concepts like emotions can denote an advantageous strategy as well for robots (Lee-Johnson and Carnegie, 2007). 'Fear' as basic emotion (cf. Lee-Johnson and Carnegie, 2007) within an affect heuristic denotes in interesting concept to integrate results from the dynamic risk assessment, for instance, as it is described by the psychodynamic structure model in Section 3.5.2.

With scope on the refinement of the safety knowledge, user dialogs could be suitable, in particular with new objects. A Marvin-like agent may ask question in order to extend its knowledge (cf. Anderson et al., 1986). Since, unknown objects should appear to comprise unpredictable risks, this might trigger a user dialog.

Finally, the work at hand comprises a procedural model for systematically generating the safety knowledge for object interaction. This systematic approach denotes a first milestone of a safety strategy. In further steps, a strategy to verify the safety-related knowledge and the risk assessment server is required. Furthermore, the consistency of the knowledge base is of interest, but in particular, the verification of the overall robotic system, see as well Alexander et al. (2007, 2008, 2009, 2010).

Literature

Ahle, E., 2007. Autonomous systems: A cognitive-oriented approach applied to mobile robotics. Ph.D. thesis, University of Duisburg-Essen, Duisburg, Germany.

Ahle, E., Söffker, D., 2006. A cognitive-oriented architecture to realize autonomous behavior - part II: application to mobile robotics. In: IEEE International Conference on Systems, Man and Cybernetics SMC. pp. 2221–2227.

Alami, R., Simeon, T., Krishna, K. M., 2002. On the influence of sensor capacities and environment dynamics onto collision-free motion plans. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. Vol. 3. Lausanne, Switzerland, pp. 2395–2400.

Albus, J., 1991. Outline for a theory of intelligence. IEEE Transactions on Systems, Man, and Cybernetics 21 (3), 473–509.

Alexander, R. D., Gorry, B., Kelly, T. P., 2010. Safety lifecycle activities for autonomous systems development. In: Proceeding of the 5th SEAS DTC Technical Conference. Edinburgh, UK.

Alexander, R. D., Hall-May, M., Kelly, T. P., 2007. Certification of autonomous systems. In: Proceedings of the 2nd SEAS DTC Technical Conference, July 2007. Edinburgh, UK.

Alexander, R. D., Herbert, N. J., Kelly, T. P., 2008. Structuring safety cases for autonomous systems. In: Proceedings of the 3rd IET International Conference on System Safety. Birmingham, U.K., pp. 1–6.

Alexander, R. D., Herbert, N. J., Kelly, T. P., 2009. Deriving safety requirements for autonomous systems. In: Proceedings of the 4th SEAS DTC Technical Conference. Edinburgh, UK.

Althoff, D., Kuffner, J., Wollherr, D., Buss, M., 2012. Safety assessment of robot trajectories for navigation in uncertain and dynamic environments. Autonomous Robots 32 (3), 285–302.

Anderson, J., Michalski, R., Michalski, R., Carbonell, J., Mitchell, T., 1986. Machine Learning: An Artificial Intelligence Approach. No. Bd. 2 in Machine Learning. Morgan Kaufman Publishers Incorporated.

Arkin, R. C., 1998. Behavior-based robotics. MIT Press, Cambridge, Mass.

Asimov, I., 1950. I, Robot. Gnome Press.

Barto, A. G., 2007. Temporal difference learning. Scholarpedia 2 (11), 1604.

Baumann, M., Krems, J. F., 2007. Situation awareness and driving: A cognitive model. In: Cacciabue, P. C. (Ed.), Modelling Driver Behaviour in Automotive Environments. Springer London, pp. 253–265.

Beavers, G., Hexmoor, H., 2004. Types and limits of agent autonomy. In: Nickles, M., Rovatsos, M., Weiss, G. (Eds.), Agents and Computational Autonomy. Vol. 2969 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 95–102.

Beer, M., Ferson, S., Kreinovich, V., 2013. Imprecise probabilities in engineering analyses. Mechanical Systems and Signal Processing 37 (1-2), 4–29.

Beierle, C., Kern-Isberner, G., 2006. Methoden wissensbasierter Systeme. Computational Intelligence. Vieweg+Teubner.

Bekris, K. E., Kavraki, L. E., 2007. Greedy but safe replanning under kinodynamic constraints. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. Rome, Italy, pp. 704–710.

Bertschinger, N., Olbrich, E., Ay, N., Jost, J., 2008. Autonomy: An information theoretic perspective. Biosystems 91 (2), 331–345.

Bicchi, A., Peshkin, M. A., Colgate, J. E., 2008. Safety for physical human-robot interaction. In: Siciliano, B., Khatib, O. (Eds.), Springer Handbook of Robotics. Springer, Berlin / Heidelberg, pp. 1335–1348.

Bischoff, R., Graefe, V., 2004. HERMES - a versatile personal robotic assistant. Proceedings of the IEEE Special Issue on Human Interactive Robots for Psychological Enrichment 92 (11), 1759–1779.

Blythe, J., 1999. Decision-theoretic planning. AI Magazine 20 (2), 37–54.

Börcsök, J., 2007. Functional Safety: Basic Principles of Safety-related Systems. Hüthig, Heidelberg.

Bornstein, R. F., 2003. Psychodynamic models of personality. In: Weiner, I. B. (Ed.), Handbook of Psychology. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Brooks, R. A., 1986. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation 2 (1), 14–23.

Brooks, R. A., 1991. Intelligence without representation. Artificial Intelligence 47 (1-3), 139–159.

Buller, A., 2005. Building brains for robots: A psychodynamic approach. In: Pal, S. K., Bandyopadhyay, S., Biswas, S. (Eds.), Pattern Recognition and Machine Intelligence. No. 3776 in Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 70–79.

Burgman, M., 2005. Risks and Decisions for Conservation and Environmental Management. Ecology, Biodiversity and Conservation. Cambridge University Press.

Cacciabue, P. C., Hollnagel, E., 1995. Simulation of cognition: applications. In: Hoc, J.-M., Cacciabue, P. C., Hollnagel, E. (Eds.), Expertise and technology. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, pp. 55–73.

Cameron, I., Raman, R., 2005. Process Systems Risk Management. Process Systems Engineering. Academic Press.

Carabelea, C., Boissier, O., Florea, A., 2004. Autonomy in multi-agent systems: A classification attempt. In: Nickles, M., Rovatsos, M., Weiss, G. (Eds.), Agents and Computational Autonomy. Vol. 2969 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 59–70.

Carlowitz, B., 1995. Kunststoff-Tabellen. Hanser.

Castelfranchi, C., 1995. Guarantees for autonomy in cognitive agent architecture. In: Wooldridge, M., Jennings, N. (Eds.), Intelligent Agents. Vol. 890 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 56–70.

Castelfranchi, C., Falcone, R., 2003. From automaticity to autonomy: The frontier of artificial agents. In: Hexmoor, H., Castelfranchi, C., Falcone, R., Weiss, G. (Eds.), Agent Autonomy. Vol. 7 of Multiagent Systems, Artificial Societies, and Simulated Organizations. Springer, Berlin / Heidelberg, pp. 103–136.

Collier, J., 2002. What is autonomy? International Journal of Computing Anticipatory Systems: CASY 2001 - Fifth International Conference CASYS'01 on Computing Anticipatory Systems, 13–18.

Collier, J. D., 1999. Autonomy in anticipatory systems: Significance for functionality, intentionality and meaning. AIP Conference Proceedings 465 (1), 75–82.

Collier, J. D., Hooker, C. A., 1999. Complexly organised dynamical systems. Open Systems & Information Dynamics 6 (3), 241–302.

Collins, S., Ruina, A., Tedrake, R., Wisse, M., 2005. Efficient bipedal robots based on passive-dynamic walkers. Science 307 (5712), 1082–1085.

Conrad, M., 1993. Adaptability theory as a guide for interfacing-computers and human society. Systems Research 10 (4), 1–23.

Coradeschi, S., Ishiguro, H., Asada, M., Shapiro, S., Thielscher, M., Breazeal, C., Mataric, M., Ishida, H., 2006. Human-inspired robots. IEEE Intelligent Systems 21 (4), 74–85.

Damme, M. V., Beyl, P., Vanderborght, B., Versluys, R., Ham, R. V., Vanderniepen, I., Daerden, F., Lefeber, D., 2010. The safety of a robot actuated by pneumatic muscles-a case study. International Journal of Social Robotics 2 (3), 289–303.

Dario, P., Guglielmelli, E., Laschi, C., 2001. Humanoids and personal robots: Design and experiments. Journal of Robotic Systems 18 (12), 673–690.

Das, S. K., Fox, J., Elsdon, D., Hammond, P., 1997. A flexible architecture for autonomous agents. Journal of Experimental and Theoretical Artificial Intelligence 9, 407–440.

Dautenhahn, K., Walters, M., Woods, S., Koay, K. L., Nehaniv, C. L., Sisbot, A., Alami, R., Siméon, T., 2006. How may i serve you?: a robot companion approaching a seated person in a helping context. In: Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction. ACM Press, Salt Lake City, Utah, USA, pp. 172–179.

Di Paolo, E. A., Iizuka, H., 2008. How (not) to model autonomous behaviour. Bio Systems 91 (2), 409–423.

DIN EN ISO 10218, 2009. Robots for industrial environments - Safety requirements. Beuth Verlag, Berlin.

DIN EN ISO 12100, 2004. Safety of machinery - Basic concepts, general principles for design. Beuth Verlag, Berlin.

DIN EN ISO 8373, 2010. Manipulating industrial robots - Vocabulary. Beuth Verlag, Berlin.

Dörner, D., 2000. Die Logik des Mißlingens. Strategisches Denken in komplexen Situationen. Rowohlt, Reinbek / Hamburg.

Dworkin, G., 1976. Autonomy and behavior control. The Hastings Center Report 6 (1), 23–28.

Endsley, M. R., 1995. Toward a theory of situation awareness in dynamic systems. Human Factors: The Journal of the Human Factors and Ergonomics Society 37 (1), 32–64.

Ericson, C., 2005. Hazard analysis techniques for system safety. Wiley-Interscience, Hoboken NJ.

Ericson, C. A., 2011. Concise Encyclopedia of System Safety: Definition of Terms and Concepts. Wiley, Hoboken, NJ.

Finucane, M. L., Alhakami, A., Slovic, P., Johnson, S. M., 2000. The affect heuristic in judgments of risks and benefits. Journal of Behavioral Decision Making 13 (1), 1–17.

Fischer, Y., Bauer, A., Beyerer, J., 2011. A conceptual framework for automatic situation assessment. In: 2011 IEEE First International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA). pp. 234 –239.

Fox, J., Das, S., 2000. Safe and sound: artificial intelligence in hazardous applications. MIT Press, Cambridge, MA, USA.

Fraichard, T., 2007. A short paper about motion safety. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. Rome, Italy, pp. 1140–1145.

Fraichard, T., Asama, H., 2004. Inevitable collision states - a step towards safer robots? Advanced Robotics 18, 1001–1024.

Franklin, S., Graesser, A., 1997. Is it an agent, or just a program?: A taxonomy for autonomous agents. In: Müller, J., Wooldridge, M., Jennings, N. (Eds.), Intelligent Agents. Vol. 1193 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 21–35.

Freud, S., 1923. Das Ich und das Es. Internationaler Psycho-analytischer Verlag (available at `www.gutenberg.spiegel.de/buch/932/3` [online; accessed 22-November-2012]).

Fu, X., Söffker, D., 2011. Concept for SOM-based computer supported cooperative work. In: Proceedings of the15th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD). pp. 261–267.

Gamrad, D., 2011. Modeling, simulation, and realization of cognitive technical systems. Dissertation thesis, University of Duisburg-Essen, Duisburg, Germany.

Gamrad, D., Oberheid, H., Söffker, D., 2009. Automated detection of human errors based on multiple partial state spaces. In: Proc. 6th Vienna Conference on Mathematical Modeling on Dynamical Systems MATHMOD 2009. Vienna, Austria, pp. 651–659.

Gamrad, D., Söffker, D., 2009a. Architecture for cognitive technical systems allowing learning from interaction with unknown environments. In: 7th Workshop on Advanced Control and Diagnosis. Zielona Góra, PL.

Gamrad, D., Söffker, D., 2009b. Reduction of complexity for the analysis of human-machine-interaction. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics. pp. 1263–1268.

Gamrad, D., Söffker, D., 2009c. Simulation of learning and planning by a novel architecture for cognitive technical systems. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics. San Antonio, Texas, USA, pp. 2302–2307.

Gamrad, D., Söffker, D., 2010. Learning from conflicts in real world environments for the realization of cognitive technical systems. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics. Istanbul, Turkey, pp. 1995–2002.

Gao, D., Wampler, C. W., 2009. Head injury criterion. Robotics Automation Magazine 16 (4), 71 –74.

Geibel, P., 2001. Reinforcement learning with bounded risk. In: Proceedings of the 18th International Conference on Machine Learning. ICML '01. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 162–169.

Gill, A., 1962. Introduction to the theory of finite-state machines. McGraw-Hill, Boston, MA.

Gips, C., Wiebrock, S., Wysotzki, F., 1998. Learning of constraints and logical implications in the spatial domain. In: Beiträge zum Treffen der GI-Fachgruppe 1.1.3 (Maschinelles Lernen), TR 98/11 in Technischer Bericht des FB Informatik. TU.

Goertzel, B., Pennachin, C., 2007. Contemporary approaches to artificial general intelligence. In: Goertzel, B., Pennachin, C. (Eds.), Artificial General Intelligence. Springer, Berlin / Heidelberg, pp. 1–30.

Goldman, R. P., Boddy, M. S., 1994. Epsilon-Safe planning. In: Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence. Seattle, WA, USA, pp. 253–261.

Gray, P. D., Salber, D., 2001. Modelling and using sensed context information in the design of interactive applications. In: Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction. Springer, London, UK, pp. 317–336.

Grosan, C., Abraham, A., 2011. Intelligent systems a modern approach. Springer, New York.

Haddadin, S., Albu-Schaffer, A., Hirzinger, G., 2008. The role of the robot mass and velocity in physical human-robot interaction - part i: Non-constrained blunt impacts. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. pp. 1331–1338.

Haddadin, S., Albu-Schäffer, A., Hirzinger, G., 2010. Safety analysis for a human-friendly manipulator. International Journal of Social Robotics 2 (3), 235–252.

Haddadin, S., Haddadin, S., Khoury, A., Rokahr, T., Parusel, S., Burgkart, R., Bicchi, A., Albu-Schäffer, A., 2012. On making robots understand safety: Embedding injury knowledge into control. The International Journal of Robotics Research 31 (13), 1578–1602.

Ham, R., Sugar, T., Vanderborght, B., Hollander, K., Lefeber, D., 2009. Compliant actuator designs. Robotics and Automation 16 (3), 81–94.

Hans, A., Schneegaß, D., Schäfer, A. M., Udluft, S., 2008. Safe exploration for reinforcement learning. In: Proceedings of the 16th European Symposium on Artificial Neural Networks ESANN'08. pp. 143–148.

Harper, C., Virk, G., 2010. Towards the development of international safety standards for human robot interaction. International Journal of Social Robotics 3 (2), 229–234.

Hayes-Roth, B., 1995. An architecture for adaptive intelligent systems. Artif. Intell. 72 (1-2), 329–365.

Heinzmann, J., Zelinsky, A., 1999. A safe-control paradigm for human-robot interaction. Journal of Intelligent and Robotic Systems 25 (4), 295–310.

Heinzmann, J., Zelinsky, A., 2003. Quantitative safety guarantees for physical human-robot interaction. International Journal of Robotics Research 22 (7-8), 479 –504.

Hertzberg, J., Chatila, R., 2008. AI reasoning methods for robotics. In: Siciliano, B., Khatib, O. (Eds.), Springer Handbook of Robotics. Ch. 9. Springer, Berlin / Heidelberg, pp. 207–223.

Hollnagel, E., 1998. Cognitive Reliability and Error Analysis Method (CREAM). Elsevier Science Ltd.

Hollnagel, E., 1999. Accident and barriers. In: Proceedings of the 7th European Conference on Cognitive Science Approaches to Process Control. Villeneuve, France, pp. 175–180.

Hollnagel, E., Pederson, O. M., Rasmussen, J., 1981. Notes on human performance analysis. Tech. Rep. Risø-M-2285, Risø National Laboratory, Roskilde, Denmark.

Hooker, C., 1995. Critical discussion of j. h. holland, adaptation in natural and artificial systems. Philosophy and Psychology 8, 287–299.

Hruschka, E. R., Ebecken, N. F., 2006. Extracting rules from multilayer perceptrons in classification problems: A clustering-based approach. Neurocomputing 70 (1–3), 384–397.

Ikuta, K., Ishii, H., Nokata, M., 2003. Safety evaluation method of design and control for human-care robots. International Journal of Robotics Research 22 (5), 281 –297.

ISO/DIS 13482, 2011. Robots and robotic devices - Safety requirements for non-industrial robots - Non-medical personal care robot. Beuth Verlag, Berlin.

Jonas, H., 1985. The imperative of responsibility: in search of an ethics for the technological age. University of Chicago Press.

Kaelbling, L. P., 1993. Learning in embedded systems. MIT Press.

Kaelbling, L. P., Littman, M. L., Moore, A. W., 1996. Reinforcement learning: A survey. Journal of Artificial Intelligence Research 4 (1), 237–285.

Kandel, E. R., 1999. Biology and the future of psychoanalysis: A new intellectual framework for psychiatry revisited. American Journal of Psychiatry 156 (4), 505–524.

Kant, I., Kirchmann, J., 1869. Immanuel Kant's Kritik der praktischen Vernunft. Philsophische Bibliothek. L. Heimann.

Kaplan, S., 1997. The words of risk analysis. Risk Analysis 17 (4), 407–417.

Kaplan, S., Garrick, B. J., 1981. On the quantitative definition of risk. Risk Analysis 1 (1), 27–37.

Karlsson, B., Karlsson, N., Wide, P., 2000. A dynamic safety system based on sensor fusion. Journal of Intelligent Manufacturing 11 (5), 475–483.

Khatib, O., 1985. Real-time obstacle avoidance for manipulators and mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation. pp. 500–505.

Kim, B., Song, J., 2010. Hybrid dual actuator unit: A design of a variable stiffness actuator based on an adjustable moment arm mechanism. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. pp. 1655–1660.

Kintsch, W., 1998. Comprehension: A Paradigm for Cognition. Cambridge University Press.

Klein, P., 1991. The safety-bag expert system in the electronic railway interlocking system elektra. Expert Systems with Applications 3 (4), 499–506.

Koenig, S., Simmons, R. G., 1994. Risk-sensitive planning with probabilistic decision graphs. In: Doyle, J., Sandewall, E., Torasso, P. (Eds.), KR'94: Principles of Knowledge Representation and Reasoning. Morgan Kaufmann, San Francisco, CA, USA, pp. 363–373.

Kulić, D., Croft, E., 2006. Real-time safety for human-robot interaction. Robotics and Autonomous Systems 54 (1), 1–12.

Kulić, D., Croft, E., 2007. Pre-collision safety strategies for human-robot interaction. Autonomous Robots 22 (2), 149–164.

Kurzweil, R., 2005. The Singularity Is Near: When Humans Transcend Biology. A Penguin Book: Science. Viking.

Kuter, U., Levine, G., Green, D., Rebguns, A., Spears, D., Dejong, G., 2007. Learning constraints via demonstration for safe planning. Technical report WS-07, AAAI Press.

Lacevic, B., Rocco, P., 2010. Kinetostatic danger field - a novel safety assessment for human-robot interaction. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. pp. 2169–2174.

Laird, J. E., 2009. Toward cognitive robotics. Proceedings of SPIE 7332, 73320Z–73320Z–11.

Langley, P., 2005. An adaptive architecture for physical agents. In: Proceedings of the IEEE International Conference on Web Intelligence WIC. pp. 18 – 25.

LaValle, S. M., 2006. Planning algorithms. Cambridge University Press, Cambridge; New York.

Lee-Johnson, C. P., Carnegie, D. A., 2007. Emotion-based parameter modulation for a hierarchical mobile robot planning and control architecture. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007). San Diego, CA, USA, pp. 2839–2844.

Leveson, N., 2003. White paper on approaches to safety engineering. Design, 1–10.

Leveson, N. G., 2012. Engineering a Safer World: Systems Thinking Applied to Safety. MIT Press, Cambridge, Massachusetts.

Lew, J., Jou, Y., Pasic, H., 2000. Interactive control of human/robot sharing same workspace. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. pp. 535–540.

Lussier, B., Chatila, R., Ingrand, F., Killijian, M. O., Powell, D., 2004. On fault tolerance and robustness in autonomous systems. In: 3rd IARP - IEEE/RAS - EURON Joint Workshop on Technical Challenges for Dependable Robots in Human Environments. Manchester, UK, pp. 7–9.

Madhavakrishna, K., Alami, R., Simeon, T., 2006. Safe proactive plans and their execution. Robotics and Autonomous Systems 54 (3), 244–255.

Mahadevan, S., 1996. Machine learning for robots: A comparison of different paradigms. In: Proceedings of the Workshop on Towards Real Autonomy, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Osaka, Japan.

Matsumoto, Y., Heinzmann, J., Zelinsky, A., 1999. The essential components of human-friendly robot systems. In: International Conference on Field and Service Robotics. pp. 43–51.

Matthias, A., 2008. Automaten als Träger von Rechten: Plädoyer für eine Gesetzesänderung. Logos, Berlin.

Maturana, H. R., Varela, F. J., 1998. The tree of knowledge: the biological roots of human understanding. Shambhala, Boston, Mass.

McCarthy, J., 1963. Situations, actions and causal laws. Technical report, Stanford University.

Mitchell, T. M., Keller, R. M., Kedar-Cabelli, S. T., 1986. Explanation-based generalization: A unifying view. Machine Learning 1 (1), 47–80.

Mosemann, H., Wahl, F. M., 2001. Automatic decomposition of planned assembly sequences into skill primitives. IEEE Transactions on Robotics and Automation 17 (5), 709–718.

Murphy, R. R., 2000. Introduction to AI Robotics, 1st Edition. MIT Press, Cambridge, MA, USA.

Najmaei, N., Kermani, M. R., 2011. Applications of artificial intelligence in safe human-robot interactions. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 41 (2), 448–459.

Nam, K., Kim, B., Song, J., 2010. Compliant actuation of parallel-type variable stiffness actuator based on antagonistic actuation. Journal of Mechanical Science and Technology 24 (11), 2315–2321.

Nehaniv, C. L., Dautenhahn, K., 2002. The correspondence problem. In: Dautenhahn, K., Nehaniv, C. L. (Eds.), Imitation in animals and artifacts. MIT Press, Cambridge, MA, USA, pp. 41–61.

Newell, A., Simon, H., 1976. Computer science as empirical inquiry: Symbols and search. Communications of the ACM 19 (3), 113–126.

Ng, A. Y., Kim, H. J., 2004. Stable adaptive control with online learning. In: Advances in Neural Information Processing Systems. pp. 13–18.

Nilsson, N. J., 1984. Shakey the robot. Tech. Rep. 323, AI Center, SRI International, CA, USA.

Oberheid, H., Gamrad, D., Söffker, D., 2008. Closed loop state space analysis and simulation for cognitive systems. In: Billington, J., Duan, Z., Koutny, M. (Eds.), 8th International Conference on Application of Concurrency to System Design (ACSD 2008), Xi'an, China, June 23-27, 2008. pp. 39–44.

Pace, C., Seward, D. W., 2005. A model for autonomous safety management in a mobile robot. In: International Conference on Computational Intelligence for Modelling, Control and Automation. Vol. 1. pp. 1128–1133.

Park, J., Kim, B., Song, J., Kim, H., 2007. Safe link mechanism based on passive compliance for safe human-robot collision. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. pp. 1152–1157.

Park, J., Song, J., 2009. Collision analysis and evaluation of collision safety for service robots working in human environments. In: International Conference on Advanced Robotics ICAR. pp. 1–6.

Perkins, T. J., Barto, A. G., 2003. Lyapunov design for safe reinforcement learning. Journal of Machine Learning Research 3, 803–832.

Petti, S., Fraichard, T., 2005. Safe motion planning in dynamic environments. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. Alberta, Canada, pp. 2210–2215.

Pfeifer, R., Scheier, C., 2001. Understanding intelligence. MIT Press, Cambridge, Mass.

Pinker, S., 1994. The Language Instinct: How the Mind Creates Language. HarperCollins, New York.

prEN ISO 8373, 2010. Manipulating industrial robots - Vocabulary. Beuth Verlag, Berlin.

Proske, D., 2008. Catalogue of risks natural, technical, social and health risks, 16th Edition. Springer, Berlin / London.

Rasmussen, J., 1983. Skills, rules, and knowledge: signals, signs, and symbols, and other distinctions in human performance models. IEEE Transactions on Systems, Man and Cybernetics 13 (3), 257–266.

Rasmussen, J., 1986. Information processing and human-machine interaction: an approach to cognitive engineering. Elsevier Science Ltd.

Reiter, R., 2001. Knowledge in action: logical foundations for specifying and implementing dynamical systems. MIT Press, Cambridge, Mass.

Ridley, M., 2003. Nature Via Nurture: Genes, Experience, and What Makes Us Human. HarperCollins, New York.

Rohde, M., Stewart, J., 2008. Ascriptional and 'genuine' autonomy. Bio Systems 91 (2), 424–433.

Rokeach, M., 1973. The nature of human values. Free Press, New York.

Rummery, G. A., Niranjan, M., 1994. On-line Q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166, Cambridge University.

Russell, S., Norvig, P., 2010. Artificial Intelligence: A Modern Approach, 3rd Edition. Prentice Hall Series in Artificial Intelligence. Prentice Hall.

Rychlak, J. F., 1990. The Psychology of Rigorous Humanism, 2nd Edition. University Press New York.

Sammut, C., Webb, G. I., 2010. Encyclopedia of machine learning, XXVI Edition. Springer, New York / London.

Schlegel, C., Wörz, R., 1999. Interfacing different layers of a multilayer architecture for sensorimotor systems using the object-oriented framework SMARTSOFT. In: Proceedings of the Third European Workshop on Advanced Mobile Robots. Zurich, Switzerland, pp. 195–202.

Seward, D., Pace, C., Agate, R., 2007. Safe and effective navigation of autonomous robots in hazardous environments. Autonomous Robots 22 (3), 223–242.

Seward, D., Pace, C., Morrey, R., Sommerville, I., 2000. Safety analysis of autonomous excavator functionality. Reliability Engineering & System Safety 70 (1), 29–39.

Siciliano, B., Khatib, O., 2008. Introduction. In: Springer Handbook of Robotics. Springer, Berlin / Heidelberg, pp. 1–4.

Sigaud, O., Peters, J., 2010. From motor learning to interaction learning in robots. In: Sigaud, O., Peters, J. (Eds.), From Motor Learning to Interaction Learning in Robots. No. 264 in Studies in Computational Intelligence. Springer, Berlin / Heidelberg, pp. 1–12.

Simon, H., 1969. The Sciences of the Artificial. Karl Taylor Compton Lectures. MIT Press.

Singh, S., Sutton, R. S., 1996. Reinforcement learning with replacing eligibility traces. Machine Learning 22, 123–158.

Slovic, P., Finucane, M. L., Peters, E., MacGregor, D. G., 2004. Risk as analysis and risk as feelings: Some thoughts about affect, reason, risk, and rationality. Risk Analysis 24 (2), 311–322.

Smithers, T., 1997. Autonomy in robots and other agents. Brain and Cognition 34, 88–106.

Smits, R., 2010. Robot skills - design of a constraint-based methodology and software support. Ph.D. thesis, Universiteit Leuven, Leuven, Belgium.

Söffker, D., 2001. From human-machine-interaction modeling to new concepts constructing autonomous systems: A phenomenological engineering-oriented approach. Journal of Intelligent and Robotic Systems 32 (2), 191–205.

Söffker, D., 2008. Interaction of intelligent and autonomous systems - part I: Qualitative structuring of interaction. Mathematical and Computer Modelling of Dynamical Systems 14 (4), 303–318.

Söffker, D., Ahle, E., 2008. Interaction of intelligent and autonomous systems - part II: realization of cognitive technical systems. Mathematical and Computer Modelling of Dynamical Systems 14 (4), 319–339.

Stephans, R., 2004. System Safety for the 21st Century: The Updated and Revised Edition of System Safety 2000. Wiley-Interscience.

Strube, G., 1998. Modelling motivation and action control in cognitive systems. In: Schmid, U., Krems, J. F., Wysocki, F. (Eds.), Mind Modeling. Papst, Berlin, pp. 111–130.

Strube, G., Habel, C., Konieczny, L., Hemforth, B., 2003. Kognition. In: Handbuch der Künstlichen Intelligenz, G. Görz, C.-R. Rollinger und J. Schneeberger (ed.). Vol. 4. Oldenbourg Wissenschaftsverlag, pp. 19–72.

Suita, K., Yamada, Y., Tsuchida, N., Imai, K., Ikeda, H., Sugimoto, N., 1995. A failure-to-safety "Kyozon" system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. Vol. 3. pp. 3089–3096.

Sutton, R. S., Barto, A. G., 1998. Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA.

Thorndike, E., 1932. The Fundamentals of Learning. Books for college libraries. Teachers college, Columbia university.

Thrun, S. B., 1992. Efficient exploration in reinforcement learning. Technical Report CMU-CS-92-102, Carnegie Mellon University, Pittsburgh, PA, USA.

Tokic, M., Palm, G., 2011. Value-difference based exploration: Adaptive exploration between epsilon-greedy and softmax. In: KI 2011: Advances in Artificial Intelligence. Springer, Berlin / Heidelberg, pp. 335–346.

Tokic, M., Palm, G., 2012. Adaptive exploration using stochastic neurons. In: Artificial Neural Networks and Machine Learning - ICANN 2012 - Part II. Vol. 7553 of Lecture Notes in Computer Science. Springer, Lausanne / Switzerland, pp. 42–49.

Traver, V., del Pobil, A., Perez-Francisco, M., 2000. Making service robots human-safe. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. pp. 696–701.

Turkle, S., 1988. Artificial intelligence and psychoanalysis: A new alliance. Daedalus 117 (1), 241–268.

Van Den Berg, J., Miller, S., Goldberg, K., Abbeel, P., 2011. Gravity-based robotic cloth folding. Algorithmic Foundations of Robotics IX, 409–424.

Varela, F. J., 1979. Principles of biological autonomy. Prentice Hall PTR.

Veres, S. M., Lincoln, N. K., Molnar, L., 2011. Control engineering of autonomous cognitive vehicles - a practical tutorial. Tech. rep., Faculty of Engineering and the Environment, University of Southampton.

Verhagen, H., 2004. Autonomy and reasoning for natural and artificial agents. In: Nickles, M., Rovatsos, M., Weiss, G. (Eds.), Agents and Computational Autonomy. Vol. 2969 of Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 21–40.

Vermorel, J., Mohri, M., 2005. Multi-armed bandit algorithms and empirical evaluation. In: Machine Learning: ECML 2005. pp. 437–448.

Vernon, D., Metta, G., Sandini, G., 2007. A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. IEEE Transactions on Evolutionary Computation 11 (2), 151 –180.

Vlahavas, I., Vrakas, D., 2005. Intelligent Techniques For Planning. IGI Global.

Vogiatzis, D., Stafylopatis, A., 2002. Reinforcement learning for rule extraction from a labeled dataset. Cognitive Systems Research 3 (2), 237–253.

von Goldammer, E., Paul, J., 1995. Autonomie in biologie und technik. In: Ziemke, A., Kaehr, R. (Eds.), Jahrbuch für Komplexität in den Natur-, Sozial- und Geisteswissenschaften. Vol. 6 of Realitäten und Rationalitäten. Duncker & Humblot, Berlin, pp. 277–298.

Wand, Y., Wang, R. Y., 1996. Anchoring data quality dimensions in ontological foundations. Communication of the ACM 39 (11), 86–95.

Wang, P., 1995. Non-axiomatic reasoning system: exploring the essence of intelligence. Ph.D. thesis, Indiana University.

Wang, P., 2006. Rigid Flexibility: The Logic of Intelligence. Applied Logic Series. Springer, Berlin / Heidelberg.

Wang, P., 2007. The logic of intelligence. In: Goertzel, B., Pennachin, C. (Eds.), Artificial General Intelligence. Springer, Berlin / Heidelberg, pp. 31–62.

Wardziński, A., 2006. The role of situation awareness in assuring safety of autonomous vehicles. In: Górski, J. (Ed.), Computer Safety, Reliability, and Security. No. 4166 in Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, pp. 205–218.

Wardziński, A., 2008. Safety assurance strategies for autonomous vehicles. In: Proceedings of the 27th International Conference on Computer Safety, Reliability, and Security SAFECOMP. Springer, Berlin / Heidelberg, pp. 277–290.

Wassink, M., Stramigioli, S., 2007. Towards a novel safety norm for domestic robotics. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS. San Diego, CA, USA, pp. 3354–3359.

Watkins, C., 1989. Learning from delayed rewards. Ph.D. thesis, University of Cambridge, England.

Weng, Y., Chen, C., Sun, C., 2009. Toward the Human–Robot Co-Existence society: On safety intelligence for next generation robots. International Journal of Social Robotics 1 (4), 267–282.

Wimpenny, J. H., Weir, A. A. S., Clayton, L., Rutz, C., Kacelnik, A., 2009. Cognitive processes associated with sequential tool use in new caledonian crows. PLoS ONE 4 (8).

Wolf, S., Hirzinger, G., 2008. A new variable stiffness design: Matching requirements of the next robot generation. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. pp. 1741–1746.

Yadava, S. C., Singh, S. K., 2009. An introduction to client/server computing. New Age International (P) Ltd., New Delhi.

Yamada, Y., Hirasawa, Y., Huang, S., Umetani, Y., Suita, K., 1997. Human-robot contact in the safeguarding space. IEEE/ASME Transactions on Mechatronics 2 (4), 230–236.

Yampolskiy, R., Fox, J., 2013. Safety engineering for artificial general intelligence. Topoi 32 (2), 1–10.

Yampolskiy, R. V., 2013. Artificial intelligence safety engineering: Why machine ethics is a wrong approach. In: Müller, V. C. (Ed.), Philosophy and Theory of Artificial Intelligence. No. 5 in Studies in Applied Philosophy, Epistemology and Rational Ethics. Springer, Berlin / Heidelberg, pp. 389–396.

Zeilinger, H., Deutsch, T., Muller, B., Lang, R., 2008. Bionic inspired decision making unit model for autonomous agents. In: IEEE International Conference on Computational Cybernetics. pp. 259 –264.

Zinn, M., Khatib, O., Roth, B., Salisbury, J., 2004. Playing it safe. IEEE Robotics and Automation Magazine 11 (2), 12–21.

Zollo, L., Siciliano, B., Laschi, C., Teti, G., Dario, P., 2002. Compliant control for a cable-actuated anthropomorphic robot arm: an experimental validation of different solutions. In: Proceedings of the IEEE International Conference on Robotics and Automation ICRA. Vol. 2. pp. 1836–1841.

Own Publications

In der vorliegenden Dissertation vorgestellte Konzepte und Ergebnisse sind Teil dieser Originalarbeiten. Teile von gemeinschaftlichen Originalarbeiten wurden ausschließlich verwendet 1) unter Zustimmung der Koautoren und 2) sofern diese vom Autor der vorliegenden Arbeit selbst verfasst wurden. Verwendete Teile darüber hinaus wurden ordnungsgemäß referenziert.

Ertle, P., Gamrad, D., Voos, H., Söffker, D., 2010a. Action planning for autonomous systems with respect to safety aspects. In: Proceedings of the IEEE International Conference on Systems Man and Cybernetics (SMC) 2010. Istanbul, Turkey, pp. 2465–2472.

Ertle, P., Söffker, D., 2010. Towards risk analysis to enable safe service robotics. In: Baloian, N., Luther, W., Söffker, D., Urano, Y. (Eds.), Interface and Interaction Design for Learning and Simulation Environments. DAAD-German Summer Academy at the University Duisburg-Essen; revised contributions. Logos, Berlin, pp. 33–35.

Ertle, P., Tokic, M., Tobias, B., Ebel, M., Voos, H., Söffker, D., 2012a. Conceptual design of a dynamic risk-assessment server for autonomous robots. In: Proceedings of the 7th German Conference on Robotics. VDE Verlag, pp. 250–254.

Ertle, P., Tokic, M., Voos, H., Söffker, D., 2012b. Towards learning of safety knowledge from human demonstrations. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012). Vilamoura, Algarve, Portugal, pp. 5394–5399.

Ertle, P., Voos, H., Söffker, D., 2010b. Development of safe autonomous mobile service robots using an active integrated approach. In: International Symposium on Robotics ISR 2010. Munich, Germany, pp. 1121–1128.

Ertle, P., Voos, H., Söffker, D., 2010c. On risk formalization of on-line risk assessment for safe decision making in robotics. In: Proceedings of the 7th IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environments. Toulouse, France, pp. 15–22.

Ertle, P., Voos, H., Söffker, D., 2012c. Utilizing dynamic hazard knowledge for risk sensitive action planning of autonomous robots. In: Proceedings of the IEEE International Symposium on Robotic and Sensors Environments ROSE. Magdeburg, Germany, pp. 162–167.

Tokic, M., Ertle, P., Palm, G., Söffker, D., Voos, H., 2012. Robust Exploration/Exploitation trade-offs in safety-critical applications. In: Proceedings of the 8th International Symposium on Fault Detection, Supervision and Safety of Technical Processes. IFAC, Mexico City, Mexico, pp. 660–665.

Voos, H., Ertle, P., 2009. Online risk assessment for safe autonomous mobile robots - a perspective. In: 7th Workshop on Advanced Control and Diagnosis. Zielona Góra, PL.

Supervised Student's Work

Im Rahmen von Forschungs- und Projektarbeiten im 'ZAFH auonome Mobile Serviceroboter' wurden von Philipp Ertle und Prof. Dr.-Ing. Holger Voos die nachstehenden Studien-, Bachelor-, Projekt- und Masterarbeiten inhaltlich betreut, wobei Bestandteile und Ergebnisse aus den Forschungs- und Projektarbeiten sowie den studentischen Qualifikationsarbeiten wechselseitig in die jeweiligen Arbeiten und somit auch in diese Promotionsarbeit eingeflossen sind.

Bystricky, T., 2010. Regelbasierte Risikoanalyse-Konzeption und Realisierung. Master thesis, University of Applied Sciences Ravensburg-Weingarten, ZAFH autonome Mobile Servieroboter, Weingarten, Germany.

Ebel, M., 2012. Development of a graph-based database for safe service robots. Master thesis, University of Applied Sciences Ravensburg-Weingarten, ZAFH autonome Mobile Servieroboter, Weingarten, Germany.

Hondorf, T., 2010. Evaluierung von Methoden zur Entwicklung sicherheitskritischer Softwaresysteme. Studienarbeit, University of Applied Sciences Ravensburg-Weingarten, ZAFH autonome Mobile Servieroboter, Weingarten, Germany.

Nuber, M., 2010. Entwicklung eines Mikrocontroller-Systems zur sicheren Abschaltung eines Roboterantriebs. Bachelorarbeit, University of Applied Sciences Ravensburg-Weingarten, ZAFH Autonome Mobile Servieroboter, Weingarten, Germany.

Ramachandran, R., 2010. Case study of the software development for safety-critical systems. Research project, University of Applied Sciences Ravensburg-Weingarten, ZAFH autonome Mobile Servieroboter, Weingarten, Germany.

Keyword Register

A

ALARP . . . 16
Antagonistic-controlled stiffness . . . 20
Automated supervision . . . 32
Autonomy . . . 35
  Adaptability . . . 40
  Anticipation . . . 41
  Control of control . . . 39
  Robot autonomy . . . 36
  Selectors for change . . . 39
  Structural determinism and coupling37
  Structural drift . . . 37
  System ontogeny . . . 39

B

Binary strategy . . . 22

C

Categorization of objects . . . 83
Cognitive robots . . . 11
Cognitive science . . . 10
  Information processing system . . . 10
  Knowledge-based behavior . . . 11
  Macro-cognition . . . 10
  Micro-cognition . . . 10
  Physical symbol system hypothesis . 10
  Rule-based behavior . . . 11
  Skill-based behavior . . . 11
Cognitive technical system
  Need for 13
Cognitive technical systems . . . 11
Compliant actuation . . . 19
Conservative approach . . . 78

D

Danger Index . . . 23
Definitions
  Human-originated and robot-caused hazards . . . 63
  Object-originated and robot-caused hazards . . . 64
  Robot-originated and human-caused hazards . . . 62
  Robot-originated and object-caused hazards . . . 63
  Robot-originated and robot-initiated hazards . . . 61
  Safety Principle . . . 93
Design safety feature . . . 15
Distributed Macro-Mini Actuation . . 19, 20
Dynamic risk assessment . . . 25, 31

E

Elusive robot . . . 23, 24
Equilibrium-controlled stiffness . . . 19
Evidence-based approach . . . 78

F

Force limited actuation . . . 20

G

Gradual strategy . . . 22
Guardian agents . . . 29

H

Hazard analysis
  Hazard description . . . 83
  Preliminary risk estimation . . . 85
Hazard identification matrix . . . 82
Hazard matrix analysis . . . 81
Hazard performance measure . . . 138
Hazard Theory . . . 58
  Hazard triangle . . . 59
  Hazardous element . . . 59
  Initiating mechanism . . . 59
  Target and threat . . . 59
Head Injury Coefficient . . . 18
Heteronomy . . . 35
High-level planning and reasoning . . . 27
HRI . . . 6

I

ILCA . . . 53

Action planning 55
Architecture 53
Learning 54
Mental action space 55
Perception 55
Inevitable Collision States 25
Information quality evaluation 95
Coverage level 97
Effort level 95
Reliability level 96
Injury knowledge 18
Intelligent robots 7
Assumption of Insufficient Knowledge and Resources 7
Efficient intelligence 8

K

Kinetostatic danger field 23
Knowledge-based level 11, 53

L

Learning
Implications for system safety 131
Learning in robotics 131
Non-stationary cliff-walking problem136
Performance according to hazard potential 138
Revision of safety features 146
The extended safety procedure 145
The ironing task 148
The stack-it-safely task 150
learning from demonstration 131
Lightweight design 18

M

Mechanically controlled stiffness 20

N

Non-stationary cliff-walking problem . . 136

O

Observations
Autonomy requires learning 68
Available collision safety approaches65
Need for a safety knowledge base . . . 66
Need for active safety management . 66
Parameterization via cognitive level 68
Robot as hazard resource 65
Safety as a matter of interactions . . 67
Operating hazards analysis 80

P

Passive compliant actuation 19
Physical Symbol System Hypothesis 26
Post-collision strategies 19
Pre-collision strategies 22
Preliminary risk estimation 85
Consequence level 85
Likelihood level 85
Risk acceptance 85
Probabilistic Collision States 25
Problem identification 69
Problems
Integration of new knowledge 70
Realized of situation awareness 69
Risk-aware decisions 69
Procedural model 81
Psychodynamic structure model 47

R

Reinforcement learning 131
Reliability 14
Representation 94
Ambiguous - 94
Deficient - 94
Incomplete - 94
Meaningless states 94
Proper - 94
Responsibility for objects 58
Risk
-sensitive perception 107
-sensitive planning 107
Conservative modeling 78
Evidence-based approach 78
Quantification 91
Risk assessment relation 105
Risk assessment server 109
Risk function 98
Example 102
Hicks Scale 103
Robot architectures 8
Deliberative paradigm 8
Hybrid paradigm 9
Reactive paradigm 8
Robot hazard classes 59
Robot science 6
Robotic hazards 57
Rule-based level 11, 53

S

Safety 14

Exploration 133
in reinforcement learning 133
Learning safety from demonstrations 144
Safety Bag concept 29
Safety knowledge
Concept of explicit - 71
Example 72, 73
Generation of - 81
Intrinsic - 75
Problem of incompleteness 132
Refinement of - 79
Safety Principle 88, 93
Applicability 93
Conclusion 91
Definition 93
Degree of detail 76
Dyadic structure 91
Generation procedure 89
Parameters 93
Premise 88, 91
procedural model 89
Required information basis 94
Risk expression 93
Risk function 98
Specification process 88
Safety Principle simulator 118
Safety Protocol 30
Safety standards 17
AS/NZS 4360 85
DIN EN ISO 10218-1 17
DIN EN ISO 12100 . . . . 17, 85
DIN EN ISO 8373 35
ISO/NP 13482 17
Scenario hazard potential 137
Self-safety management 25
Series Elastic Actuation 19
Situation Awareness 31
Situation-Operator-Model 51
Algorithm representation 52
Assumptions 51
explicit 51
implicit 51
Characteristic 51
Control problems 52
Notation 51
Operator 51
Relation 51
Risk relation 105
Situation 51
Situational risks 106
Skill-based level 11, 53
Step-ladder model 11
Structural model human psyche 46
Structural-controlled stiffness 20
System boundary 57
System safety 13
ALARP 16
Design safety feature 15
Hazard identification 14
Hazard risk assessment 14
Hazard risk control 15
Residual risk 15
Risk acceptance 15
Safety vs. reliability 14
System safety process 14

T

Temporal difference learning 135

U

Utility Theory 28

V

Velocity Profiles 24

W

Whole Arm Manipulator 19

Z

Zero-G controller 23

Lebenslauf

Der Lebenslauf ist in der Online-Version aus Gründen des Datenschutzes nicht enthalten.