



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# Master Seminar

Andreas Dubs

Aufgabenorientierter (implizierter) Aktionsplaner

# Inhaltsverzeichnis

<b>1</b>	<b>Rückblick</b>	<b>3</b>
1.1	Motivation . . . . .	3
1.2	verwandte Arbeiten . . . . .	4
1.2.1	BDI-Konzept . . . . .	4
1.2.2	KAMRO . . . . .	4
1.2.3	TUM Rosie . . . . .	4
<b>2</b>	<b>Konzept</b>	<b>5</b>
2.1	Projekt I . . . . .	6
2.2	Projekt II . . . . .	7
<b>3</b>	<b>Roboterarm Aktionsplaner</b>	<b>8</b>
3.1	expliziter Planer . . . . .	9
3.2	Simulation . . . . .	9
3.3	impliziter Planer . . . . .	9
3.4	Erfolgsprüfung . . . . .	10
3.5	Arbeitsbereiche . . . . .	10
3.6	Fazit . . . . .	11
<b>4</b>	<b>Abschlusswort</b>	<b>12</b>
4.1	Risiken . . . . .	12
4.2	Ausblick . . . . .	12
	<b>Literaturverzeichnis</b>	<b>14</b>

# 1 Rückblick

## 1.1 Motivation

Die mobile Roboterplattform »SCITOS« G5 mit einem darauf montierten 5-DOF Roboterarm und Greifer der Firma Schunk (Abbildung 1.1) soll effizient die gestellte Aufgaben lösen. Es sind mehrere Sensoren, wie Laserscanner, Kamera, Ultraschall und Bumper mit Drucksensoren, vorhanden. Antrieb der Plattform und Roboterarm sollen mit einem in der Plattform sich befindenden Recheneinheit synchronisiert werden. Auch ein autonomer aufgabenorientierter Verhalten erfordert effiziente Algorithmen, um erfolgreiche Pläne zu erstellen und die rechtzeitige Reaktionen bei der Umgebungsveränderungen zu ermöglichen. Eine Batterie dient als Energiequelle und begrenzt die Laufzeit.

Weiterer Aspekt ist ein sicherer Betrieb in einer Umgebung mit Menschen. Die statische Objekte lassen sich in einer 3D-Weltmodell abbilden, die dynamische Objekte dagegen können ihre Positionen ändern und müssen bei der Bewegungen ständig betrachtet und mögliche Kollisionen vermeiden werden. Dies erfordert eine Architektur der Robotersteuerung, die nicht nur in der Lage ist Pläne zu erstellen und umzusetzen, sondern auch mit veränderlichen Umständen umgehen zu können.



Abbildung 1.1: Scitos G5

## 1.2 verwandte Arbeiten

Für einen Konzept ist das Wissen über Architekturen erforderlich. So stehen zum Auswahl zwei Gegensatzpaare der Architektur mit jeweils ihren Vor- und Nachteilen. Funktionsorientierte (deliberative) Architektur ermöglicht Planung der Aktionen und erfordert genaues Weltmodell, dagegen verhaltensorientierte (reaktive) Architektur bringt kurze Reaktionszeiten auf äußere und innere Ereignisse, weil auf Planung von optimalen Routen und Langzeitstrategien verzichtet wird. Zweites Gegensatzpaar beschreibt innere Aufbau. So kann hierarchische Architektur mit ihrer Geschlossenheit des Systems oder verteilte ausgewählt werden. Eine verteilte Architektur zeichnet sich durch unabhängige Teilsysteme aus, es wird aber ein zusätzlicher Kommunikationsaufwand notwendig. Durch Aufteilung auf Ebenen mit verschiedenen Ansätzen entstehen hybride Architekturen, die Effizienz der Robotersteuerung sehr stark verbessern können.

### 1.2.1 BDI-Konzept

Der BDI-Konzept soll menschliche Entscheidungsfindung nachbilden. Beliefs sind das Wissen (z.B. Weltmodell) und können unsichere Informationen beinhalten. Desires sind feste Ziele, die der Roboter verfolgen soll. Intentions sind aus Zielen abgeleitete widerspruchsfreie Aufgaben.

### 1.2.2 KAMRO

KAMRO ist ein Versuch die Vorteile von reaktiven (für Kollisionsvermeidung) und deliberativen (für Planung) Verhalten in einem System zu vereinen. Die hierarchische Architektur besteht aus deliberativen Planungssystem und reaktiven sensorunterstützten Robotersteuerungskomponente.

### 1.2.3 TUM Rosie

TUM Rosie ist ein aktueller Projekt der TU München mit einem verteiltem Ansatz. Die verwendete CRAM (*Cognitive Robot Abstract Machine*)-Architektur besitzt Kernel, der an die BDI-Konzept angelehnt. So ist es möglich aus Aufgaben abstrakte Pläne mit Teilzielen zu erstellen. Drei Steuerungskomponenten (Perzeption, Navigation, Manipulation) realisieren vorgegebene Teilziele und weisen damit einen sensorunterstützten reaktiven Verhalten auf.

## 2 Konzept

Die gewonnenen Erkenntnisse aus anderen Projekten dienen als eine gute Grundlage für die Entwicklung der Scotos G5 Robotersteuerung. Die verteilte Aufbau und reaktiver sensorunterstützter Verhalten werden verfolgt und bei der Entwicklung eingesetzt. ROS (*Robot Operating System*)-Framework bietet ein Kommunikationsmechanismus für eine verteilte Aufbau. ROS-Bibliotheken beinhalten viele Methoden und Tools, die sehr hilfreich bei der Entwicklung der Robotersteuerung sind. Angelehnt an CRAM-Architektur wurde ein vorläufiger Konzept der Roboterteuerung erstellt (Abbildung 2.1).

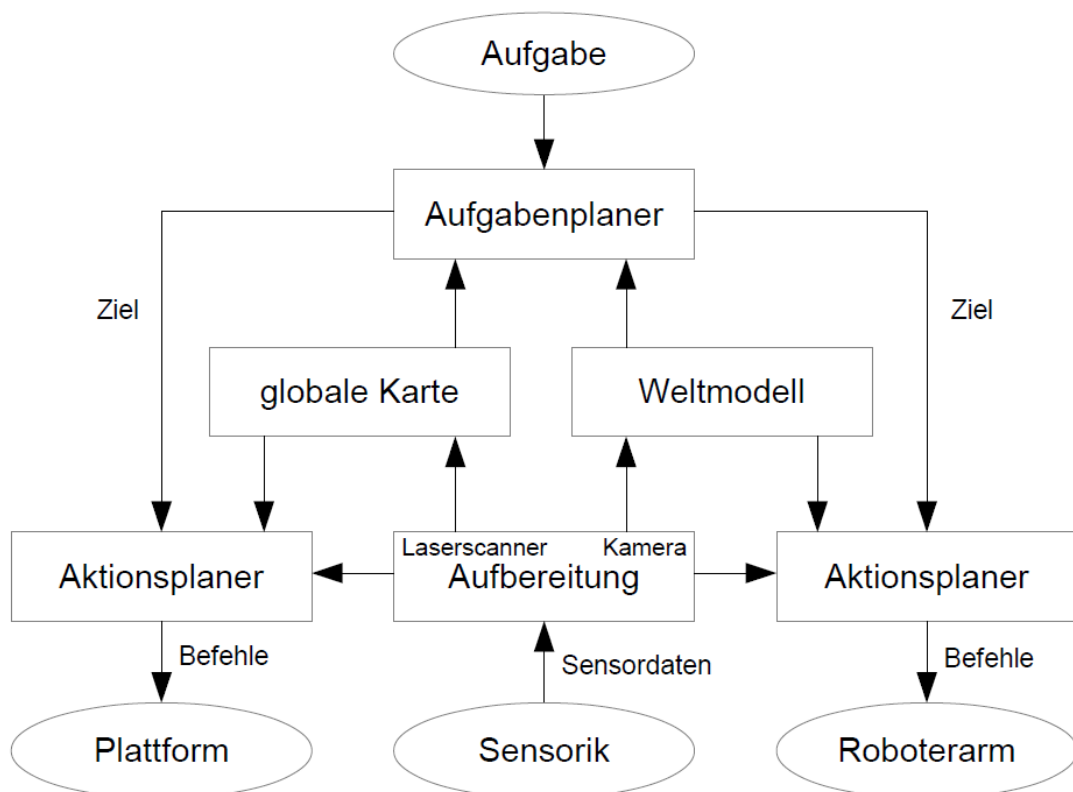


Abbildung 2.1: Konzept

Dieser Konzept ermöglicht parallele Plattform- und Roboterarmsteuerung mit Einbeziehung aktueller Sensordaten für einen reaktiven Verhalten. Der Aufgabenplaner soll die gestellten Aufgaben in Teilziele zerlegen und an die entsprechende Aktionsplaner weiter reichen. Die Aktionsplaner versuchen diese Teilziele durch Erstellung von geeigneten Aktionspläne auszuführen und suchen bei möglichen Kollisionen nach Alternativen.

## 2.1 Projekt I

Im Projekt I wurde ein vorhandener Aktionsplaner für Ansteuerung der Plattform angeschaut und optimiert. Dieser Aktionsplaner sorgt für die Roboternavigation und ist auf zwei Ebenen aufgeteilt (Abbildung 2.2).

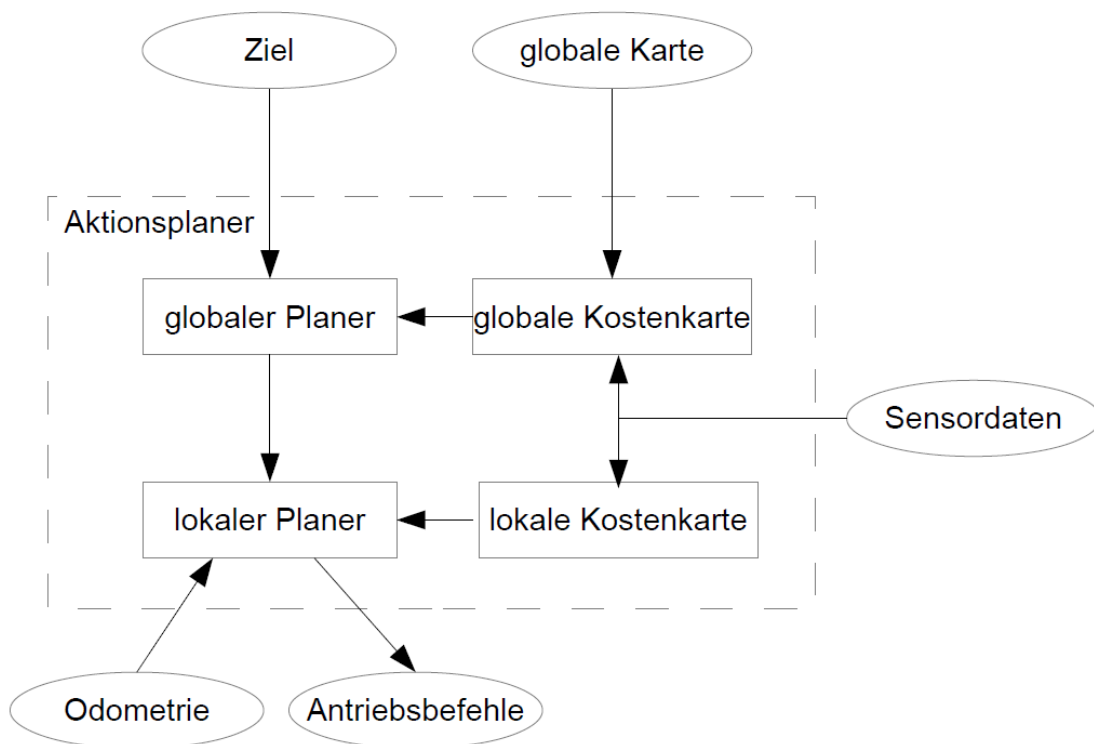


Abbildung 2.2: Plattform Aktionsplaner

Auf der ersten Ebene wird aus der mit GMapping von OpenSLAM erstellten globalen 2D-Rasterkarte eine globale Kostenkarte durch Zielvorgabe auf ein Teil begrenzt und die Kollisionsbereiche markiert. Der globaler Planer sucht in dieser Karte eine Pfad, mit dem das

vorgegebene Ziel erreicht werden kann. Die lokale Kostenkarte zeichnet sich durch einen begrenzten Bereich um den Roboter herum und verändert sich mit der Fahrt der Plattform, so dass der Roboter sich immer in der Mitte der Kostenkarte befindet (Abbildung 2.3). Der lokale Planer versucht dem berechneten Pfad zu folgen und erstellt mit Hilfe der lokalen Kostenkarte die Antriebsbefehle für die Plattform. Die Sensordaten könnten zur Aktualisierung der Karte und damit zur Neuberechnung des Pfades führen.

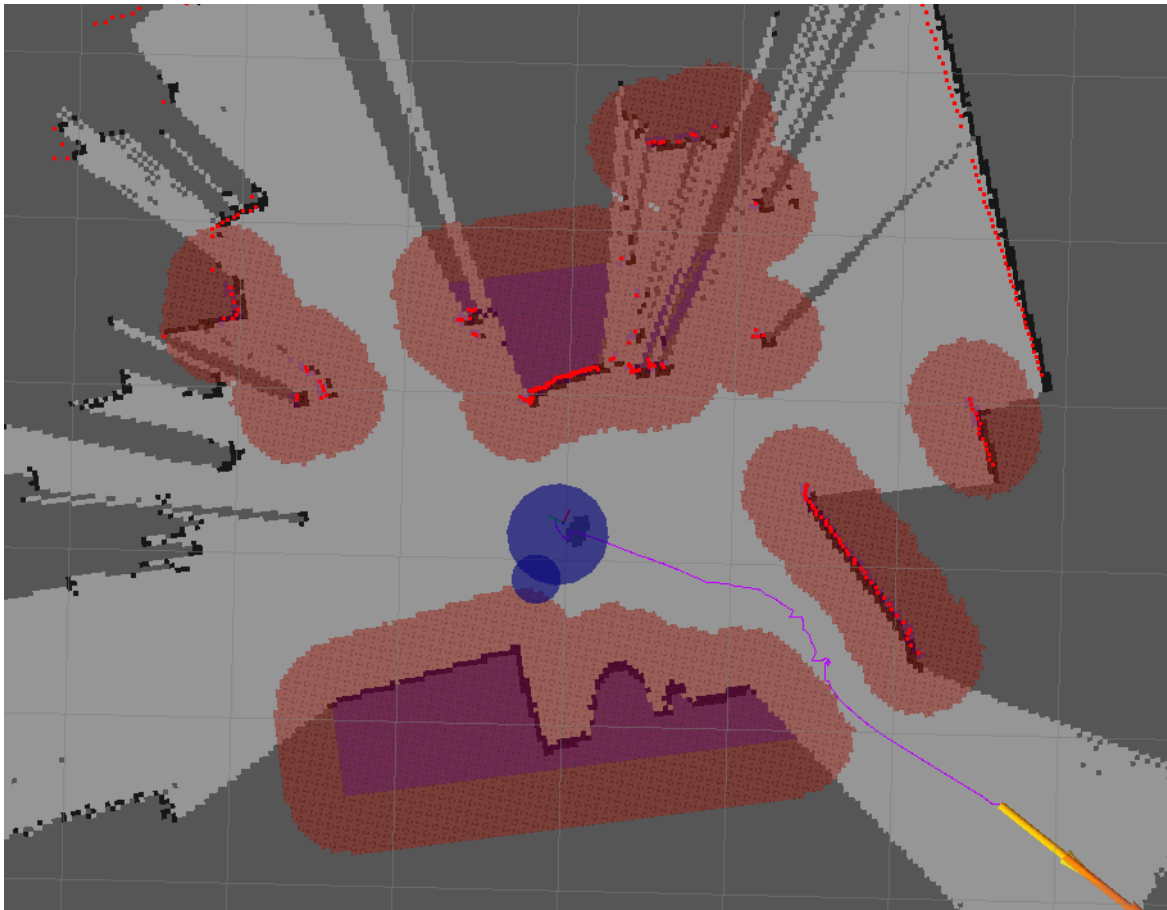


Abbildung 2.3: lokale Kostenkarte

## 2.2 Projekt II

Im Projekt II erfolgen erste Schritte bei der Aufbau des Roboterarm-Aktionsplaners. Die Fragestellung beschäftigt sich mit Struktur des Planers und welche Aufgaben dieser übernehmen sollte. Es werden einzelne Komponente und die Schnittstellen des Aktionsplaners definiert. Im nächsten Kapitel wird dies genauer betrachtet.

### 3 Roboterarm Aktionsplaner

Angelehnt an 2-Schichtenstruktur des Plattform-Aktionsplaners wurde eine vorläufige Aufbau des Roboterarm-Aktionsplaners erstellt (Abbildung 3.1). Die vorgegebene Ziele sind: ein Objekt aufnehmen, abstellen oder sich zu eine bestimmten Position bewegen (mit oder ohne einen Objekt). Aufgabe des impliziten Planers ist die Erstellung von Trajektorien, die zur erfolgreichen Erfüllung des Ziels führen sollen. Der expliziter Planer verarbeitet die Trajektorien zur ausführbaren Roboterarmbefehlen.

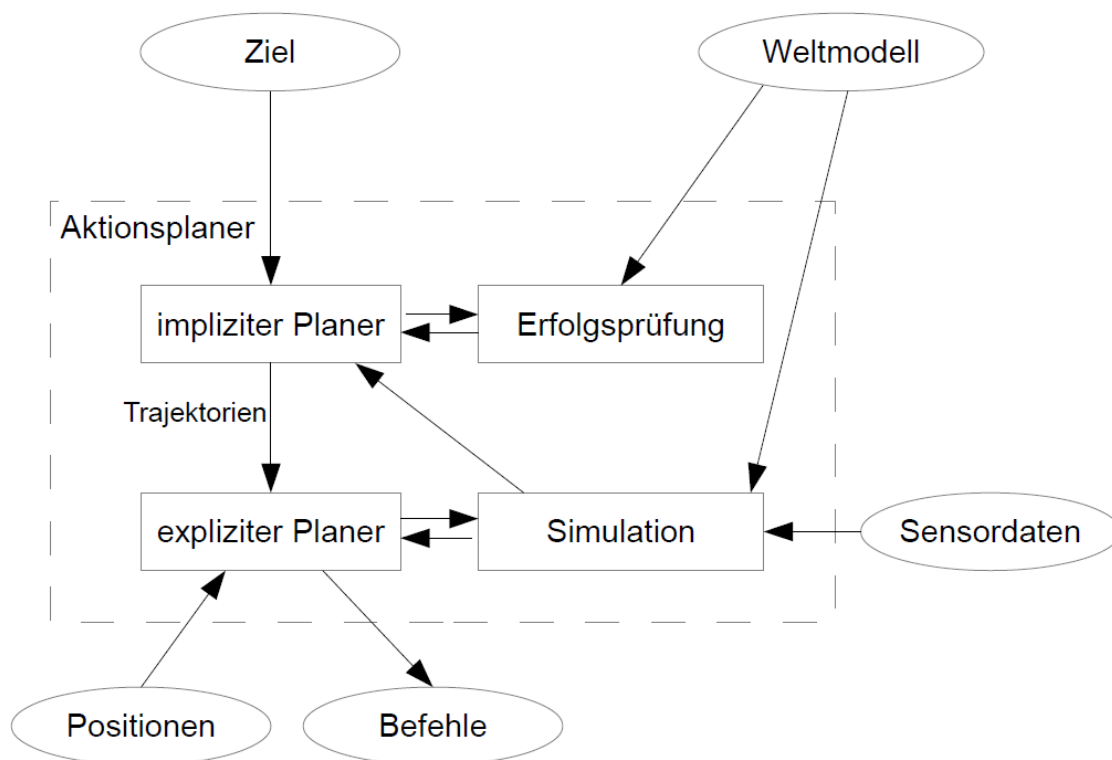


Abbildung 3.1: Roboterarm Aktionsplaner



### 3.1 expliziter Planer

In der vorherigen Arbeiten wurden Methoden für Bahnbewegungen und Kinematik entwickelt, die weiterhin eingesetzt werden können. ROS-Framework beinhaltet ein Interface für Kinematik-Rechner mit Funktionen für Vorwärts- und inverse Kinematik, die mit vorhandenen Methoden gefüllt werden müssen. Vorwärtskinematik erlaubt aus Motorpositionen und Gelenklängen den TCP (*Tool Center Point*) zu berechnen. Die inverse Kinematik berechnet mit numerischen Verfahren aus angestrebten TCP-Position, aktuellen Positionen der Motoren und TCP-Position, die neue Motorpositionen.

Der expliziter Planer interpoliert mit Hilfe der vorgegebenen Bahngeschwindigkeit die Trejektorie und berechnet die notwendige Zwischenpositionen. Mit inversen Kinematik werden zu jeder Position entsprechende Motorpositionen und daraus Motorbefehle errechnet.

### 3.2 Simulation

Bestimmte Motorpositionen können aus mechanischen Gründen nicht erreicht werden, um Eigenkollisionen der Gelenke zu vermeiden. Die Simulation prüft Ausführbarkeit der Motorbefehle. Mit Vorwärtskinematik und Umgebungsinformationen (Weltmodell, aktuelle Sensordaten mit dynamischen Veränderungen) findet eine Kollisionsprüfung statt. Bei möglichen Kollisionen wird eine Neuplanung der Trajektorien notwendig.

Die Simulation sorgt nicht nur für Kollisionsprüfung, sondern auch für Überwachung. Bei Greifoperationen werden die Sensordaten für Justierung der Armbewegungen benutzt. Dies bedeutet, dass die Greifoperationen nicht implizit planbar sein können, lediglich die Vorposition und Ausrichtung des TCP zu dem greifbaren Objekt muss mit implizitem Planer ermittelt und anschließend angefahren werden. Nach erfolgreichem Greifen oder Abstellen eines Objekts befindet sich der TCP in einer Nach-Position und folgende Bewegungsbahnen können weiter geplant werden.

### 3.3 impliziter Planer

Der impliziter Planer beschäftigt sich mit Suche einer Trajektorienfolge, die zu Erreichung einer bestimmten Position führen sollte. ROS-Framework beinhaltet bereits mehrere Bewegungsplaner- Implementierungen. Ein Bewegungsplaner kann Trajektorien, ohne das Wissen über tatsächliche Roboterarm-Konstruktion, ermitteln. Der expliziter Planer und Simulation sollen die Ausführbarkeit und Kollisionsfreiheit solcher Trajektorien sicherstellen.

Bei ungeeigneten Trajektorien muss nach alternativen Bewegungsbahnen gesucht werden. Solche Aufbau erlaubt Einsatz voneinander unabhängiger Komponente.

### 3.4 Erfolgsprüfung

Eine bessere Möglichkeit ist, schon bei Trajektorienplanung die Roboterarmspezifikation einzusetzen. Dies ermöglicht frühzeitigen Ausschluss nicht ausführbarer Trajektorien, die zu Eigenkollisionen führen können. Auch Gelenkpositionen zu TCP in einer Trajektorie können zur Überprüfung dienen. Dabei müssen eigentliche Motorbefehle noch nicht ausgerechnet werden.

### 3.5 Arbeitsbereiche

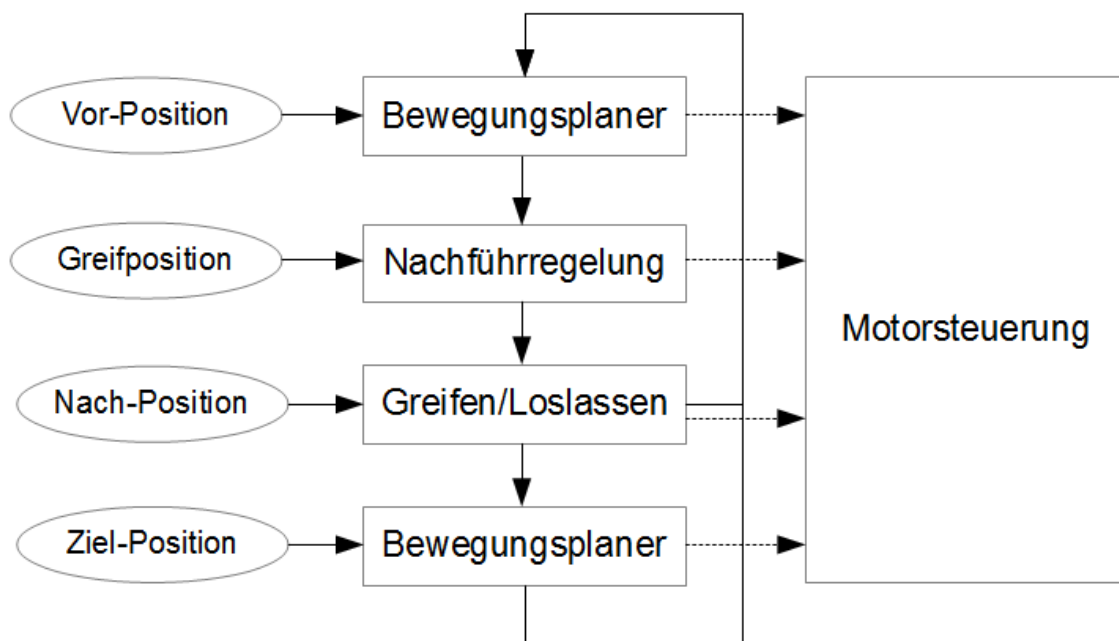


Abbildung 3.2: Ablauf des Aktionsplaners

Derzeit wird an zwei angrenzenden Bereichen gearbeitet. Erster Bereich stellt die Objekt - Erkennung und -Lokalisation mit KINECT-Kamera dar. Daraus sollen eine geeignete Stelle für Greifoperation und sichere Vor-Position des TCP ermittelt werden. Die Vor-Position ist für Bahnplanung notwendig und wird damit zu eigentlichem Ziel des impliziten Planers. Nach

dem die sichere Vor-Position erreicht wurde, soll der Greifer aus Vor-Position in die Greifposition mit einer linearen Bewegung überführt werden. Dieser Prozess wird mit Hilfe der Kamera nachgeführt, um den Objekt dabei nicht umstoßen.

Zweiter Bereich ist mit dem taktilen rutsch-freien Greifen der Objekte verbunden. Die Drucksensoren sorgen für nicht zu festes Greifen der Objekte und erkennen mögliches Abrutschen beim Aufheben.

Der gesamte Ablauf des Aktionsplaners ist in Abbildung 3.2 dargestellt. Die Nach-Position ist die Position des TCP nach dem der Objekt gegriffen und angehoben wurde, aber auch losgelassen und auf sichere Abstand gebracht wurde. Nach einer Greifoperation ist die Bewegung zur nächsten Vor-Position für weitere Greifoperation oder zu einer bestimmten Ziel-Position möglich. Nach Erreichung der Ziel-Position sind weitere Greifoperationen mit ihren Vor-Positionen möglich. Ziel-Position ist somit nichts anderes, als eine Zwischenposition, zum Beispiel um die Plattform zu bewegen, ohne dabei die Roboterarm zu schwenken.

## 3.6 Fazit

Die Arbeit bei der Fertigstellung des Aktionsplaners setzt sich aus folgenden Aufgaben:

- Erstellung von Anwendungsfällen und Abläufen
- Abhängigkeiten von anderen Komponenten ermitteln
- Kompetenzen- Aufteilung der einzelnen Module, Definition von Schnittstellen
- Suche nach geeigneten Methoden, Anpassung vorhandener Methoden
- bei Bedarf Entwicklung eigener Methoden
- Testen und Evaluieren
- Validieren

Viele der oben genannten Aufgaben waren bisher nur teilweise angeschnitten und wurden nicht im vollem Umfang bearbeitet. Wie man sehen kann, es liegt viel Arbeit vor. Trotzdem bin ich zuversichtlich, weil RobotVisonLab Teammitglieder bei Schwierigkeiten mit Ratschlägen und Konsultationen unter die Arme greifen und keinen allein stehen lassen.

# 4 Abschlusswort

## 4.1 Risiken

Weil der Konzept nicht im Endstadium ist, kann er sich ändern. Die Änderungen bedeuten zusätzlichen Zeitbedarf, um die vorhandene Module an den neuen Konzept anzupassen. Zeitbedarf ist allgemein schwer einschätzbar. Dies hängt mit Einsetzbarkeit der vorhandene Modulen und Methoden. Projekt I hat dieses Problem deutlich gezeigt, die eingesetzte Navigation-Komponente entsprach nicht den gestellten Voraussetzungen. Es ging viel Zeit drauf, die bestehende Algorithmen zu analysieren und mögliche Lösungswege zu prüfen.

Die verwendete Lösungsansätze sollten möglichst optimal sein. Es ist sehr schwer eine optimale Lösung eines Problems zu ermitteln. Oft werden schon funktionierende Lösungen eingesetzt, obwohl die entweder nicht optimal sind oder der veränderten Anforderungen nicht mehr entsprechen.

Auch Aufbau eines stabilen Systems steht im Vordergrund. Ein sehr raffiniertes System ist unbrauchbar, wenn es nicht robust ist. Das System sollte mit auftretenden Fehlern umgehen können. Die Module sollten nicht bei jeder Kleinigkeit abstürzen, aber auch wenn doch ein Absturz vorliegt, so sollte dieser erkannt, aktueller Zustand gesichert und entsprechender Modul neu gestartet werden.

## 4.2 Ausblick

Derzeit werden Aktionsplaner entwickelt und verfeinert. Auch Arbeiten an dem Weltmodell und der Objekterkennung finden statt. Der Nächster schritt Konzept wäre der Aufgabenplaner, der die gestellten Aufgaben in Teilziele zerlegen. Erst soll die Aufgabe analysiert werden, um einen abstrakten Plan ohne konkreten Parameter zu erstellen, also was muss getan werden, um die Aufgabe zu bewältigen (Abbildung 4.1).

Jeder Punkt im Plan hat Vor-, Nach- und Kontextbedingungen. Die Vorbedingungen könnten unter anderem für die Ausführung des Teilziels benötigte Parameterbeschreibungen enthalten. Die Kontextbedingungen repräsentieren, anders als Vor- und Nachtbedingungen, wäh-

rend der ganzen Ausführungszeit geltende Bedingungen. Weiter soll der abstrakter Plan ausgeführt werden. Hier werden die Teilziele mit Parametern versehen. Die Parameter können aus dem Weltmodell oder als Resultate aus erfolgreichen Abarbeitungen vorheriger Teilziele stammen.

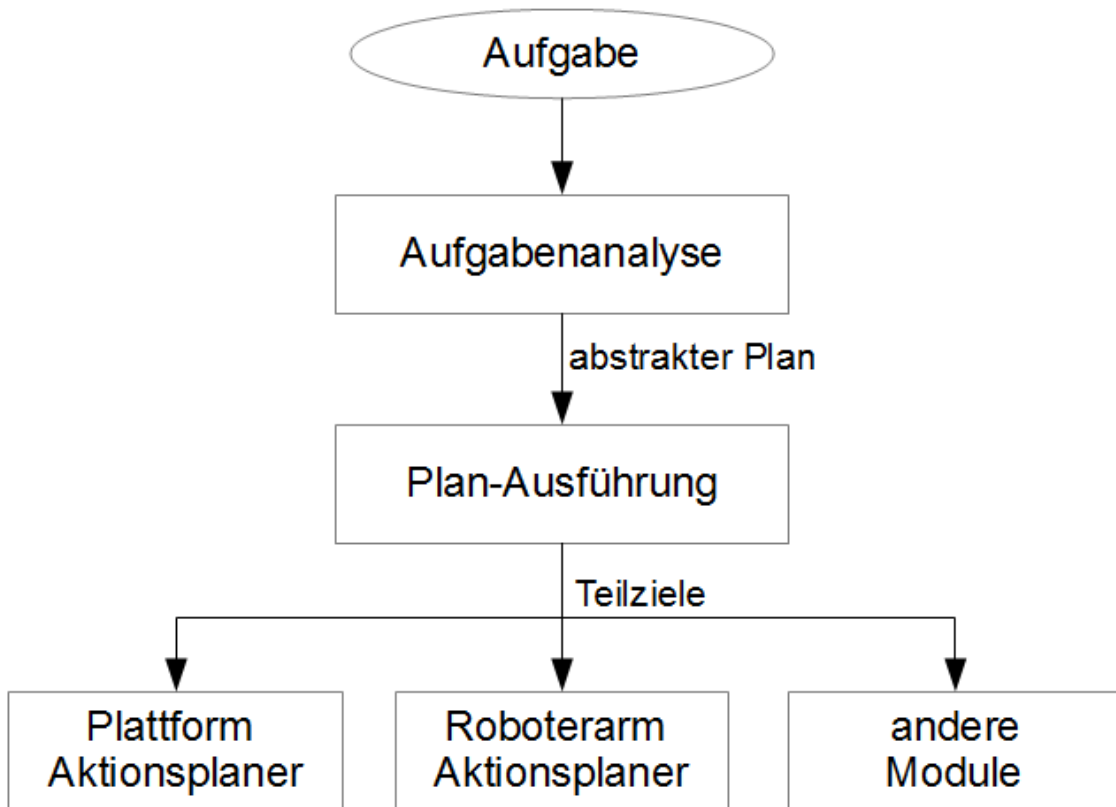


Abbildung 4.1: Aufgabenplaner

Die letzte geplante Erweiterung wäre die konkurrierende Aufgaben. Zwar kann eine Liste von Aufgaben erstellt werden und die Aufgaben nacheinander abgearbeitet werden, auch eventuell mit Prioritäten, aber die Möglichkeit mehrere Aufgaben gleichzeitig abarbeiten zu können entspricht dem angestrebten BDI-Konzept. Die Reihenfolge der Teilziele verschiedener Aufgaben könnten kombiniert werden, um ein optimales Ergebnis zu bekommen, zum Beispiel im Hinblick auf verbrauchte Zeit.

# Literaturverzeichnis

[Dubs 2012a] DUBS, Andreas: *Entwicklung eines impliziten Aktionsplaners*. 2012. – AW2 Ausarbeitung

[Dubs 2012b] DUBS, Andreas: *Entwicklung eines impliziten Planers mit Echtzeit-Kollisionsprüfung für einen mobilen Assistenzroboter Scitos G5*. 2012. – AW1 Ausarbeitung

[Kolbe 2011] KOLBE, Felix: *Einsatz des ROS-Frameworks zur Umgebungskartierung und Navigation des SCITOS G5-Service-Roboters*. 2011. – Ausarbeitung PJ2

[ROS] ROS: *ROS Wiki*. – URL [http://www.ros.org/wiki/move\\_base](http://www.ros.org/wiki/move_base)