

Inhaltsverzeichnis

Vorwort	v
Kurzfassung/Abstract	vii
Abkürzungsverzeichnis	xv
Abbildungsverzeichnis	xix
Tabellenverzeichnis	xxiii
1 Einleitung	1
1.1 Motivation	3
1.2 Problemstellung und Handlungsbedarf	6
1.3 Wissenschaftlicher Beitrag/Ziel der Arbeit	8
1.4 Struktur der Arbeit	11
2 Grundlagen zur Eingrenzung des Themas	13
2.1 Industrielle Steuerungssysteme in der Automatisierung	13
2.1.1 Automatisierbare Funktionen	15
2.1.2 Steuerung von einzelnen Funktionen - Bewegungssteuerung	16
2.1.3 Steuerung von Funktionsabläufen	18
2.1.4 Steuerungsebenen	19
2.1.5 Darstellungsform von Steuerungen	22
2.2 Virtuelle Inbetriebnahme - Test- und Entwicklungsmethode	23
2.2.1 Definition der virtuellen Inbetriebnahme	23

ix

Inhaltsverzeichnis

2.2.2	Vorteile der virtuellen Inbetriebnahme	25
2.2.3	X-in-the-Loop Simulation als Grundlage der VIBN .	27
2.3	Einführung Reinforcement Learning	30
3	Stand der Forschung	35
3.1	Zieldimensionen für die Untersuchung und Bewertung .	36
3.1.1	Zieldefinition 1 - Eingliederung in den bestehenden Prozess und Organisation	37
3.1.2	Zieldefinition 2 - Integration in etablierte Werkzeuge und IT-Systeme	38
3.1.3	Zieldefinition 3 - Kopplung an bestehende virtuelle und physische Artefakte	39
3.1.4	Zusammenfassung Zieldimension und Bewertungskriterien	40
3.2	Lösungsansätze in der Forschung	41
3.2.1	Zinn et al. - Deep Reinforcement Learning	43
3.2.2	Gankin et al./Mayer et al. - Modulare Produktionssteuerung mit Deep Reinforcement Learning . . .	46
3.2.3	Schmidl et al. - Energieoptimierung eines Förder-Handhabungssystems	47
3.2.4	Plappert et al. - Mujoco Trainingsumgebung	49
3.2.5	Waschnek (et al.) - Optimierung der Halbleiterherstellung	50
3.2.6	Dobrescu et al. - HiL Architektur	51
3.2.7	Zamora und Lopez et al. - gym-gazebo (1 und 2) .	52
3.2.8	Chebotar et al. - Sim-to-Real	54
3.2.9	Xie et al.	56
3.2.10	Fellbrich et al. - Additive Fertigung im Bauwesen .	57
3.3	Zusammenfassung und Handlungsbedarf	59

4 Lösungsansatz X-in-the-Loop Trainingsumgebung	63
4.1 Analyse Simulationsbasierte Lösungsprozesse	63
4.1.1 Allgemeiner Systemansatz (Ziel-, Handlungs-, Objekt- system) nach Albers	64
4.1.2 Domänenspezifische Entwicklungsmethodiken . .	66
4.1.3 Entwicklung als iterativer Problemlöseprozess nach Pahl/Beitz	68
4.1.4 Heuristischer Steuerungsentwurf nach Abel	70
4.2 Lösungsansatz X-in-the-Loop Trainingsumgebung	73
4.3 Anforderungen an den Lösungsansatz	75
4.3.1 Architektur	75
4.3.2 Methoden	76
4.3.3 Lösungen	77
4.3.4 Zusammenfassung	78
5 Architektur X-in-the-Loop Gym	79
5.1 Auswahl Reinforcement Learning Toolkit und Simulations- umgebung	80
5.1.1 Reinforcement Learning Algorithmen Bibliotheken	82
5.1.2 Trainingsumgebungs-Plattformen	84
5.1.3 Auswahl	86
5.2 Konzept der XiL-Gym Architektur	87
5.2.1 Anforderungen OpenAI Gym	88
5.2.2 Anforderungen VIBN Simulationswerkzeug	89
5.2.3 XiL-Gym Wrapper	90
5.3 Realisierung XiL-Gym	92
5.3.1 Kommunikation zwischen Agent und XiL-Gym . .	92
5.3.2 Validierung - Vergleich XiL-Gym und Original Gym	95
5.3.3 Erweiterung und Validierung - Skalierbarkeit . . .	97
5.3.4 Bereitstellung als Austauschplattform	98

6 Methoden für anwendungsnahe Belohnungsfunktionen	101
6.1 Herausforderungen	101
6.2 Lösungsansätze für anwendungsnahe Belohnungsfunktionen	104
6.3 Methode 1 - Curriculum Multi-Stage Reinforcement Learning	108
6.3.1 Anwendungsszenario - IMS	109
6.3.2 Beschreibung Curriculum Multi-Stage Reinforcement Learning	111
6.3.3 Anwendung der Methode auf die Teilsysteme des IMS	114
6.3.4 Zusammenführung der Ergebnisse	119
6.3.5 Fazit	120
6.4 Methode 2 - Test-Driven Reinforcement Learning	121
6.4.1 Beschreibung Test-Driven Reinforcement Learning	122
6.4.2 Anwendungsszenario - Roboter Handhabungs- / Fräszelle	125
6.4.3 Realisierung des Test-Driven Reinforcement Learning	127
6.4.4 Ergebnisse	128
6.4.5 Fazit	131
7 Lösungen zur Integration der Steuerungskonfigurationen	133
7.1 Agent-in-the-Loop - Übersicht der zu möglichen Ansätzen	133
7.1.1 Agent im Kontext der VIBN Testkonfigurationen . .	134
7.1.2 Agent in Kontext einer lernfähigen Steuerung . . .	139
7.2 Lösungen für verschiedene Konfigurationen	141
7.2.1 Model-in-the-Loop Substitution	141
7.2.2 Software-in-the-Loop Programmierung	145
7.2.3 Hardware-in-the-Loop Substitution	148
8 Ergebnisbewertung	155
8.1 Bezug zu den Zieldefinitionen	155
8.2 Bewertung der Anforderungen	156
8.3 Fazit	158

9 Zusammenfassung und Ausblick	161
Literatur	165