

Planung und Entwicklung einer mobilen, autonomen Roboterplattform für pflanzenbauliche Anwendungen

Entwicklung eines Softwarekonzepts für einen mobilen Roboter unter Verwendung des freien Pakets ROS – navigation stack

Georg Supper¹, Christian Aschauer¹, Andreas Gronauer¹ und Norbert Barta¹

Abstract: In den Agrarwissenschaften wird ausgehend von Industrie 4.0 an der Digitalisierung der Landwirtschaft gearbeitet. Mehrere Forschungsarbeiten beschäftigen sich daher mit der Entwicklung von Robotern für eine Vielzahl von landwirtschaftlichen Arbeitsprozessen. Ein großes Problem in der Robotik ist die Sicherheit in der Mensch-Maschinen-Interaktion. Hier bietet das Projekt Urban Cyber Farming-Laboratory (UrbFarmLab) eine Plattform, um kollaborative Anwendungen mit einem mobilen Roboter in einer sicheren Umgebung zu testen. Dafür wird ein mobiles Roboterfahrzeug mit einem Roboterarm entwickelt und mit Laserscanner, GPS, Encodern und Sicherheitstechnik ausgestattet. Als Software wird die freie Software des Roboter Operating Systems (ROS) verwendet. Das Paper beschreibt die Entwicklung eines Softwarekonzepts unter Verwendung der ROS-Software „navigation stack“.

Keywords: Robotik, ROS, Automatisierung, Landwirtschaft, Digitalisierung

1 Einleitung

Die Automatisierung hat die Produktivität der landwirtschaftlichen Maschinen durch die Steigerung der Effizienz, Zuverlässigkeit und Präzision erheblich erhöht und die Notwendigkeit menschlicher Eingriffe verringert [BV16]. Der Technologietransfer in der Robotik von der Industrie hin zur Anwendung in der Landwirtschaft und die Entwicklung von autonomen, intelligenten Systemen werden als wesentlich für die Verbesserung der Arbeitseffizienz angesehen und umfassen das Potenzial zur Verbesserung der Qualität von Produkten, Senkung der Produktionskosten und Verringerung von Handarbeit. [Re16] In der Forschung wird daher intensiv an der Entwicklung landwirtschaftlich genutzter Roboter für eine Vielzahl von Anwendungen, wie zum Beispiel für die Saat, den Pflanzenschutz, die Unkrautkontrolle und die Ernte, gearbeitet. [BV17]

Die Robotik befindet sich derzeit in einem fundamentalen Paradigmenwechsel, sowohl in der Forschung als auch in praktischen Anwendungen. Nachdem sie in den letzten Jahrzehnten von positionsgesteuerten, starren Robotern für typische

¹ Universität für Bodenkultur Wien, Institut für Landtechnik, Peter-Jordan-Straße 82, 1190 Wien,
georg.supper@boku.ac.at

Automatisierungsaufgaben wie Positionierung und Wegverfolgung in verschiedenen Anwendungen dominiert wurde, hat sich eine neue Generation von Robotern in der Landwirtschaft etabliert [Sp15]. Die Autonomie wurde in verschiedenen Bereichen auf unterschiedliche Weise konzeptualisiert. In der Mensch-Maschine-Interaktion wurde zum Beispiel die Autonomie weitgehend als eine Zuweisung von Funktionen zwischen einem Menschen und einem Roboter erklärt. Daraus abgeleitet gibt es zwei Ansätze, die Autonomie konzipieren. Einerseits erfordert höhere Roboterautonomie weniger häufige Interaktion und andererseits höhere Ebenen oder komplexere Formen der Interaktion [Th04].

Ein großes Sicherheitsproblem bei der Mensch-Maschine-Interaktion ist im Zusammenhang mit jeder Art von autonomer Navigation verbunden. RTK-GPS-basierte Navigationssysteme bestimmen die Position eines Fahrzeuges auf wenige cm genau und für viele Anwendungen in der Landwirtschaft sind grobe Karten verfügbar. Diese Systeme liefern keine Informationen über die „Dynamik“ der Umgebung. Karten können Fehler enthalten, welche durch die Veränderungen von Menschen oder der Natur verursacht wurden. Außerdem enthält die Karte keine Informationen über sich bewegende Objekte (z. B. Menschen, Tiere und Fahrzeuge), welche in einem Gebiet vorhanden sind [Re16].

Der Bedarf an Forschung zur Mensch-Maschine-Interaktion in der Robotik ist groß. Der Einfluss von menschlichen Faktoren kann durch ein besseres Verständnis in den Computerwissenschaften (künstliche Intelligenz) und durch neue Sensortechnik verstanden werden [Sh16].

Das Projekt Urban Cyber Farming-Laboratory (UrbFarmLab) beschäftigt sich mit der Produktion von landwirtschaftlichen Erzeugnissen, welche durch autonome Roboter unterstützt wird. Das UrbFarmLab besteht aus einer mobilen Robotereinheit, welche mit einem Greifer diverse Arbeitsaufgaben an mehreren Pflanzmodulen erledigt und die dafür notwendigen Werkzeuge aus einer Basisstation bezieht. Dieses Setting stellt einen Übergangsbereich zwischen kontrollierter Laborumgebung und dem Freiland dar. Es bietet die Möglichkeit, kollaborative Anwendung im Urban Farming oder im Gartenbau im kleinen Maßstab zu testen.

Der Schwerpunkt dieses Papers liegt in der Beschreibung eines Softwarekonzepts zur autonomen Navigation des mobilen Roboters, welches mit dem Open-Source-Programm ROS – navigation stack arbeitet. Das Ziel ist es, ein intelligentes und vollautonomes Fahrzeug zu steuern, welches in semistrukturierten oder unstrukturierten dynamischen Umgebungen, in denen Menschen, Tiere und andere Maschinen vorkommen, sicher navigiert und verschiedene Arbeits- und Servicepositionen anfährt. Die Herausforderung besteht in einem flexiblen Einsatz in unterschiedlichen Urban-Gardening-Anlagen. Für den Transport und einen flexiblen Einsatz resultiert eine einfache, kompakte und wendige Konstruktion des Roboters.

2 Material und Methode

2.1 Roboterarchitektur

Der Einsatzraum der mobilen Roboterplattform ist das Umfeld des Urban-Gardening. Um einen Transport zwischen verschiedenen Anlagen zu vereinfachen, hat der entwickelte Roboter eine Breite von 850 mm, eine Länge von 1.100 mm und ein Gewicht von 250 kg. Das Chassis besteht aus einer Stahl-Schweiß-Konstruktion. Ein elektrisches Antriebskonzept, wie in der Abb. 1 dargestellt, minimiert die Umweltbelastungen. Ein Motorcontroller (Roboteq HDC 2460) steuert die zwei Antriebsräder vorne, mit welchen über eine Drehzahlregelung die Lenkung realisiert wird. Am Heck befinden sich zwei frei drehbare Räder. Die Roboterplattform soll durch ihre technische Ausstattung in einer beliebigen Umgebung eingesetzt werden und sich in dieser zurechtfinden. Um eine autonome Navigation zu ermöglichen, sind an den Antriebswellen jeweils ein Quadraturencoder (ifm electronic RVP510), ein DGPS-Empfänger (Emlid Reach M+) und ein Laserscanner (Velodyne VLP 16) vorgesehen. Um Arbeitsaufgaben zu erledigen, bietet der Roboter eine Schnittstelle für einen Roboterarm (Universal Robots UR10). Dieser Roboterarm kann mit unterschiedlichen Werkzeugen vielfältige Aufgaben erledigen. Als Rechner steht ein In-Vehicle-PC mit ROS zur Verfügung. Im Gegensatz zu einem industriellen Setting soll der Roboter in einem begrenzten Gebiet frei navigieren. Um die notwendige Sicherheit für Menschen im Betrieb zu gewährleisten, verfügt der Roboter über 2 Not-Aus-Taster, 2 Sicherheitslaserscanner (Sick S30B) und Sicherheitskontaktpuffer am Chassis, sowie über einen Funk-Not-Aus, um die Motoren des Roboters in seiner Bewegung zu stoppen.

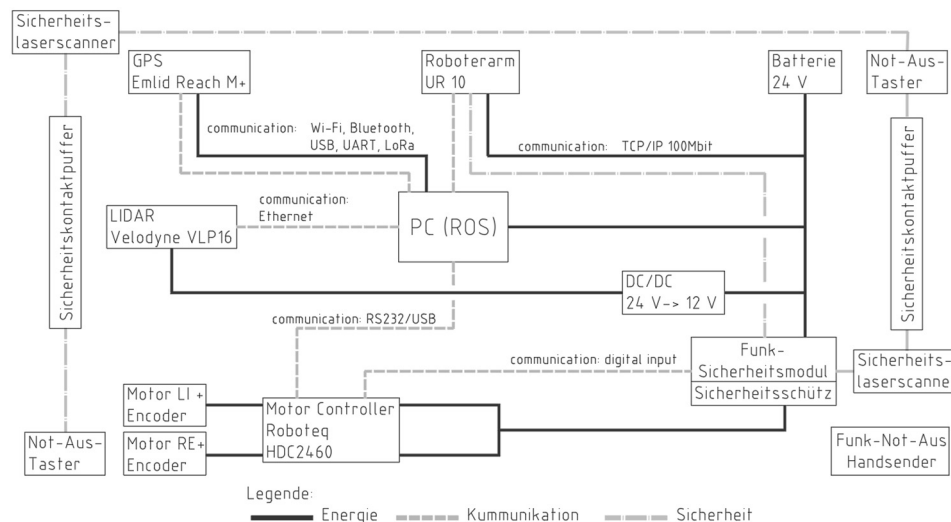


Abb. 1: Hardwarekonzept der Roboterplattform

2.2 Roboter Operating System (ROS)

ROS ist ein Framework zum Schreiben von Software für Roboter. Es handelt sich um eine Sammlung von Tools, Bibliotheken und Konventionen, die darauf abzielen, die Aufgabe zu vereinfachen, ein komplexes und robustes Roboterverhalten über eine Vielzahl von Roboterplattformen hinweg zu erstellen. ROS wurde von Grund auf entwickelt, um die kollaborative Robotik-Softwareentwicklung zu fördern [Ma16].

Die wesentlichen Merkmale sind, dass ROS eine peer-to-peer, werkzeuggestützte, mehrsprachige, leichte und frei Open-Source-Software ist. Das grundsätzliche Konzept von ROS sind Nodes, messages, topics und services [Qu09].

3 Ergebnis

3.1 Softwarekonzept

Das Softwarekonzept, wie in Abb. 2 dargestellt, basiert auf der freien Software ROS navigation stack, welches um Nodes weiterer freier Software und eigene Entwicklungen erweitert wird. Das „move base“-Paket beinhaltet die wichtigsten Schnittstellen für die autonome Navigation von Robotern und ermöglicht einfache Erweiterungen [Ma16].

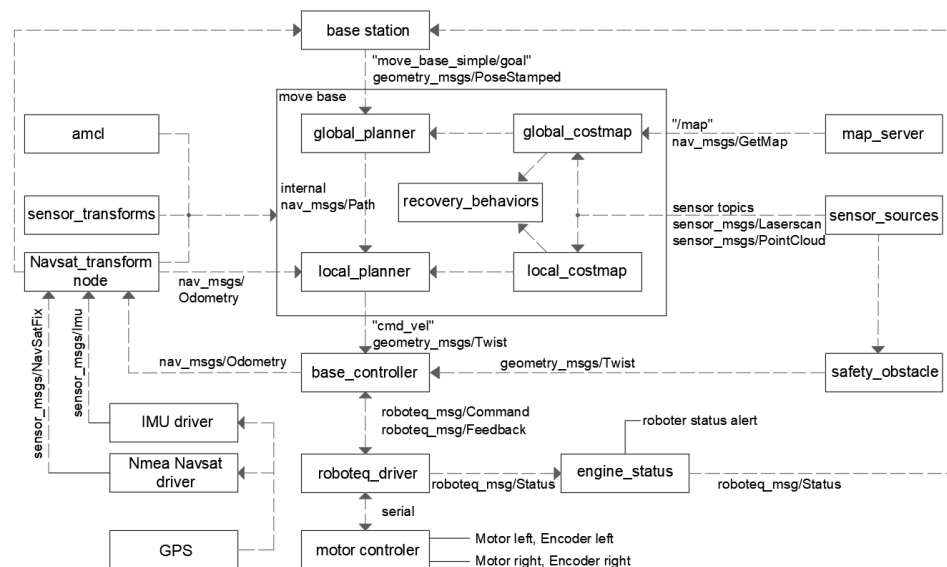


Abb. 2: Softwarekonzept des mobilen Roboters mit dem ROS – navigation stack als Basisprogramm (modifiziert nach [Ma16])

Das Navigationsziel sind die verschiedenen Arbeits- und Servicepositionen, sie werden von einer übergeordneten Basisstation an den Roboter übermittelt. Eine Karte des aktuellen Einsatzgebietes unterstützt die Software bei der Navigation durch die Berücksichtigung von bekannten Hindernissen [Ro18]. Um eine autonome Navigation zu ermöglichen, ist in der mobilen Robotik die Kenntnis der aktuellen Position eine wichtige Fragestellung [Th01].

In der verwendeten kartenbasierten Positionsbestimmung spielt der Laserscanner eine wichtige Rolle. Mit Hilfe der Daten eines Laserscanners, eines „range finders“-Modells und einer Karte der Umgebung wird eine Positionsschätzung ermittelt [Th05]. Als zusätzliche Information der Positionsschätzung werden Odometrie-Daten der Antriebsräder, welche durch das GPS-System ergänzt werden, um eine Navigation im Freifeld zu ermöglichen, integriert. Diese Sensorfusion wird durch das „Navstat transform“-Node errechnet [Mo18]. Mit Hilfe der „Adaptive Monte Carlo Localisation“ wird aus den Daten eine Positionsschätzung ermittelt [Th01].

Aus dieser Positionsschätzung und dem Navigationsziel wird im „move_base“-Paket durch einen „global-planner“ ein Pfad errechnet. Ein „local-planner“ übernimmt das „path-following“ und errechnet die Vorfahrtgeschwindigkeit sowie eine Winkelgeschwindigkeit der z-Achse des Roboters. Die Geschwindigkeits-Information wird als ROS-Topic „geometry_msg/Twist“ vom „move_base“-Paket gesendet [Ma16] [Ro18]. Das „base_controller“-Paket errechnet aus diesen Geschwindigkeiten und dem kinematischen Modell des Antriebkonzeptes die Drehzahl der Antriebsräder, übergibt die Antriebsinformation mittels Motortreiber-Node an den Motorcontroller, regelt die Drehzahl mittels PID-Reglers und errechnet aus dem Feedback der Motoren eine Odometrie-Information.

Um das Sicherheitsrisiko beim Betrieb zu minimieren, sind Informationen über die Umgebung durch die Laserscanner notwendig. Werden nicht kartierte Hindernisse erfasst, passt der „local_planner“ den Pfad entsprechend an. Zusätzlich errechnet das „safety_obstacle“-Node mit Hilfe der Laserscanner einen softwareüberwachten Raum. Befindet sich ein Objekt im geschützten Raum, wird darauf durch eine Geschwindigkeitsminderung reagiert.

4 Ausblick

Im nächsten Schritt werden die Navigationsgenauigkeit und die Hinderniserkennung mit dem beschriebenen Roboter-Setting in einer dynamischen Umgebung evaluiert. In weiterer Folge dient der Roboter mit dem Urban Cyber Farming-Laboratory als Forschungsplattform zum Testen von Sensoren und Sicherheitstechnik und für die Entwicklung von kollaborativen Robotikanwendungen im Gartenbau und darauf aufbauend für weitere landwirtschaftliche Anwendungen.

Literaturverzeichnis

- [BV16] Bechar, A. & Vigneault, C. 2016. Agricultural robots for field operations: Concepts and components. *Biosystems Engineering*, 149, 94-111.
- [BV17] Bechar, A. & Vigneault, C. 2017. Agricultural robots for field operations. Part 2: Operations and systems. *Biosystems Engineering*, 153, 110-128.
- [De14] Deutsches Institut für Normung, 2014. Roboter und Robotikgeräte - Sicherheitsanforderungen für persönliche Assistenzroboter. DIN EN ISO 13482:2014-11.
- [Ma16] Mahtani, A., Sanchez, L., Fernandez, E. & Martinez, A. 2016. *Effective Robotics Programming with ROS*, Packt Publishing.
- [Mo18] Moore, T. 2018. robot_localization [Online]. http://wiki.ros.org/robot_localization. [Accessed 2018-10-30].
- [Qu09] Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R. & NG, A. Y. 2009. ROS: an open-source Robot Operating System. ICRA Workshop on Open Source Software.
- [Re16] Reina, G., Milella, A., Rouveure, R., Nielsen, M., Worst, R. & Blas, M. R. 2016. Ambient awareness for agricultural robotic vehicles. *Biosystems Engineering*, 146, 114-132.
- [Ro18] ROS. 2018. navigation tutorial [Online]. <http://wiki.ros.org/navigation/Tutorials>. [Accessed 2018-10-30].
- [Sh16] Sheridan, T. B. 2016. Human-Robot Interaction: Status and Challenges. *Human Factors*, 58, 525-532.
- [Sp15] SPARC. 2015. Robotics 2020 Multi-Annual Roadmap – For Robotics in Europe [Online]. <http://sparc-robotics.eu/wp-content/uploads/2014/05/H2020-Robotics-Multi-Annual-Roadmap-ICT-2016.pdf>. [Accessed].
- [Th04] Thrun, S. 2004. Toward a framework for human-robot interaction. *Hum.-Comput. Interact.*, 19, 9-24.
- [Th05] Thrun, S., Burgard, W. & Fox, D. 2005. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, The MIT Press.
- [Th01] Thrun, S., Fox, D., Burgard, W. & Dellaert, F. 2001. Robust Monte Carlo localization for mobile robots. *Artif. Intell.*, 128, 99-141.