Eine beispielhafte Studie

Dennis Nienhüser

28. Januar 2018

FZI Forschungszentrum Informatik
Interaktive Diagnose- und Servicesysteme (IDS)
Haid-und-Neu-Straße 10–14
76131 Karlsruhe

Inhaltsverzeichnis

1 Einleitung

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

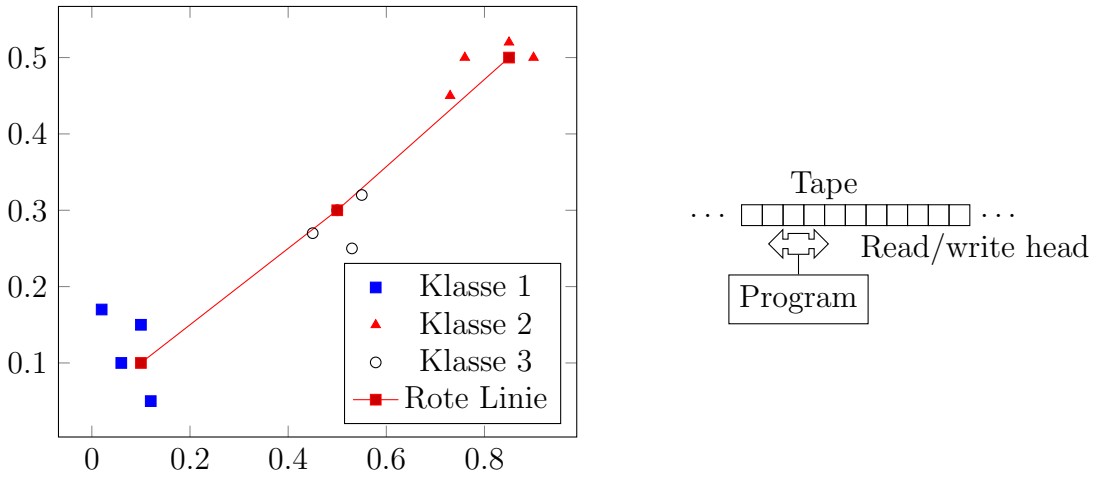

Abbildung 1: Das ist meine Grafik. Es gibt viele solcher Grafiken, aber diese gehört mir. Oder nicht?!

- `http://www.texample.net/tikz/examples`
- `http://pgfplots.sourceforge.net/gallery.html`

1.1 Robot Operating System

Das Robot Operating System bietet Kommunikationsschnittstellen für ein verteiltes System in dem Knoten Nachrichten in sogenannten Topics veröffentlichen und Dienste anbieten können

2 Lokalizierung der Roboter

Die Lokalisierung unserer Roboter bietet entweder Gmapping ROS-Packet oder die Cartographer von Google.Wenn die Laufroboter die Selbstlokalisierung ausführt,wird erst die

Ros-Launch file eingegeben dann die entsprechende Ros-Packet werden gleich aufgerufen.Folgendes wird über den Unterschied zwischen Gmapping und Google-Cartographer diskutiert.

2.1 Gmapping

Kürzlich wurden Rao-Blackwellized Partikelfilter als effektive Mittel eingeführt, um das Problem der Simultanen Lokalisierung und Kartierung (SLAM) zu lösen. Dieser Ansatz verwendet einen Partikelfilter, in dem jedes Partikel eine individuelle Karte der Umgebung trägt. Eine zentrale Frage ist daher, wie die Anzahl der Partikel reduziert werden kann. Die Anzahl von Partikeln wird in einem Rao-Blackwell-Partikelfilter zum Erlernen von Gitterkarten reduziert.
Einen Ansatz wird vorgeschlagen, um eine genaue Vorschlagsverteilung zu berechnen, die nicht nur die Bewegung des Roboters, sondern auch die jüngste Beobachtung berücksichtigt. Dies verringert die Unsicherheit über die Pose des Roboters im Voraussage-Schritt des Filters drastisch. Darüber hinaus wird in Gmapping Algorithmus diese Ansatz verwendet, um selektiv Re-Sampling-Operationen durchzuführen,die das Problem der Partikelverarmung stark reduzieren.

2.2 Cartographer

Cartographer kann als zwei separate, aber verwandte Systeme gesehen werden. Der erste ist lokaler SLAM (manchmal auch Frontend genannt).Seine Aufgabe besteht darin, einen lokal konsistenten Satz von Submaps zu erstellen und sie miteinander zu verknüpfen, aber er wird sich im Laufe der Zeit verschieben. Die meisten Optionen befinden sich in "trajektory-builder-2d.lua"für 2D und "trajektory-builder-3d.lua"für 3D.
Das andere System ist ein globales SLAM (manchmal als Backend bezeichnet). Es läuft in Hintergrundthreads und seine Hauptaufgabe besteht darin, Loop-Close-Constraints zu finden. Dies geschieht durch Scan-Matching-Scans gegen Submaps. Es enthält auch andere Sensordaten, um eine Ansicht auf höherer Ebene zu erhalten und die konsistenteste globale Lösung zu identifizieren. In 3D versucht es auch, die Richtung der Schwerkraft zu finden. Die meisten Optionen finden Sie in pose-graph.lua.
Bei einer höheren Abstraktion besteht die Aufgabe des lokalen SLAM darin, gute Submaps zu erzeugen, und die Aufgabe des globalen SLAM besteht darin, sie am konsequentesten zu verknüpfen.

Quigley et al. (2009).

Literatur

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe.