卒業論文要旨

ロボット運動の３DCG による可視化

Visualaization fo robot motion by 3DCG

システム工学群

機械・航空システム制御研究室　　1180095　田代　和人

1.　緒言

近年，周辺の環境を認知し，自ら思考して行動する知能ロボットの研究が盛んである．しかしこれらの知能ロボットは様々な方向性の機能を持ち，それらを並行して使用することからシステムが複雑になり Robot Operating System [1] [2] [3]（以下 ROS と記載）のような様々な機能が使用可能なソフトウェアフレームワークの活用が求められる．そこで本研究では，研究室に存在する実際のロボットを ROS によってその運動の再現を行うことを通じてそれらの知能ロボットの開発に向けての理解を深めると共に，現実のロボットの動作と比較することによって，正しくロボット運動を再現することを目指した．

2.　ROS

2.1　概要

ROS とは，ロボット用ソフトウェアフレームワークのことを指し，中心部として複数のプログラムを結合させる通信ライブラリと，これによって結合されているロボットの開発を支援するツール・ライブラリ群で構成されている．これらのツール・ライブラリ群はオープンソースで提供されている．

2.2　通信

ROS における実行プログラムは Node という単位で扱われる．また，通信の中で予め定められた特定の型でやり取りされるデータを Message と呼び, Node 間の通信は Topic を通じて行われる．Node 間の Message の流れを図１に示す．これは送信側・受信側共に通信相手が厳密に誰なのかという情報を必要とせずに通信を行うことができるという特徴がある．

Fig. 1　ROS communication outline.

2.2　Gazebo

今回ロボットの挙動を再現するということで ROS が対応している物理演算機能を搭載した３D ロボットシミュレータである Gazebo を使用している．今回は URDF というロボットを記述するための言語を使用して Gazebo 上にロボットを再現した．

2.3　ros_control

ros_control は ROS のパッケージの一つである．このパッケージでは URDF で設定された関節の種類に合わせてコントローラが用意でき，それぞれの PID 制御が可能になる．目標値の入力を行う場合はコントローラが立ち上げた Topic に目標値を書き込むことで行うことができる．

2.4　ros_bag

ros_bag は ROS パッケージの一つである．このパッケージは，Topic に書き込まれた Message を記録する事ができる．記録された内容は.bag という拡張子のファイルとして生成される．

3.　シミュレータ上での機体と動作の再現

3.1　シミュレータ上での機体の再現

本研究では，試作した二輪車両をシミュレータ上で再現した．試作した車両を図２に示す．車両の３D モデルデータは設計時のものを用いると共に，実機作成の後に加えられた変更を反映したものを STL ファイルで用意し，それを Gazebo に出力するために URDF ファイルを作成した．車両の構造は図３で示す通り，前後に並んだ２つの車輪と左右に設置されたキャスターによって支えられる車両となっている．また，前輪は舵角を調整するためにサーボモータにつなげたリンクで保持され，駆動用モータにつながれた後輪によって駆動するように作成した．また実物の車両はトレーラーを牽引しているが，今回は図４のように牽引車のみをモデル化した．この時の牽引車の諸元を表１に示す．

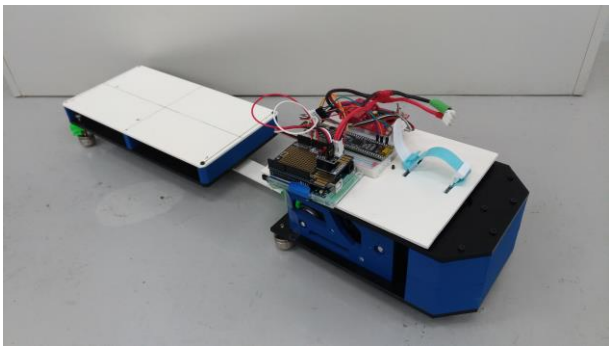

Fig. 2　Actual vehicle.

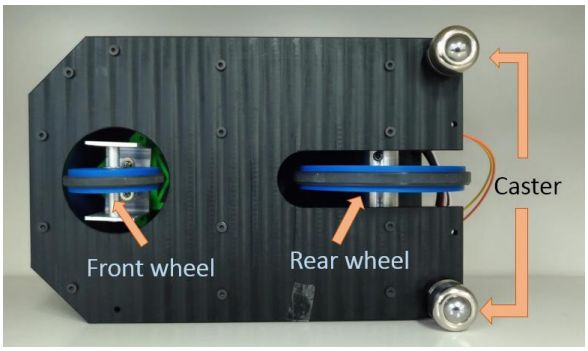

Fig. 3　Vehicle structure.

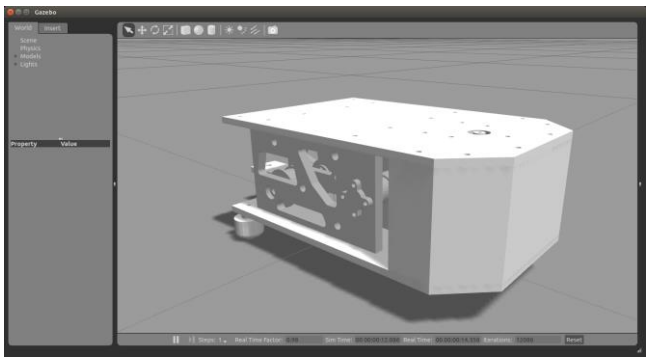

Fig. 4　Reproduction by Gazebo.

Table 1　Various elements.

Overall length [m]	0.289
Overall width [m]	0.15
Overall height [m]	0.108
Weight [kg]	1.631
Wheel coefficient of friction	0.5
Caster coefficient of friction	0.0
Rear wheel diameter [m]	0.1

3.2　シミュレータ上での動作の再現

シミュレータ上で再現する動きは前輪の舵角を一定に保ち，後輪を定速で回転させて 180°車体が回転するまで走行させることとした．これらの動作を行うために，それぞれの関節の操作は ros_control パッケージを使用して行った．また，シミュレータ上でのロボットの動きを記録する方法として ros_bag パッケージを使用して車両を構成する各部品の位置情報と速度情報を記録した．比較として実際の車両の動きのデータを使用する際は，モーションキャプチャを使用して記録を行った．また実際の車両には車両本体に対して非常に軽いものであったがトレーラーが牽引されていた．

3.2.1　舵角・速度変化時の軌道

後輪の回転速度を同一として舵角を変化させた時のそれぞれの軌道と,舵角を同一として後輪の回転速度を変化させた時のそれぞれの軌道が、予想される動作を行うかどうかを検討した．

3.2.2　実際の車両との比較

実際の車両の軌道との比較を行った．前輪の舵角は 10°で車体速度を 0.2[m/s]と 0.1[m/s]の時のそれぞれの軌道を比較した．またこれらの車体速度を再現するために後輪の回転速度は 4[rad/s]及び 2[rad/s]にそれぞれ設定した．

3.2.3　機体の改造

実際に実験を行おうとした場合，機体の改造を必要とする設定におけるシミュレーションを行った．ここでは車体片側のキャスターに摩擦係数を与え，それぞれの摩擦係数における車体の軌道の変化を検討した．また，摩擦係数を設定したキャスターが常時接地するために，摩擦係数を設定した側のキャスターには 0.01[kg]重量を増加させた．この時の車体速度は 0.2[m/s]で舵角を 10°に設定した．

4.　実験結果と考察

4.1　舵角・速度変化時の軌道

車体速度が 0.2[m/s]で舵角を 5°～20°まで変化させた時の軌道を図５に，舵角が 5°で車体速度を 0.02～0.8[m/s]まで変化させた時の軌道を図６に示す．舵角を変化させたときは舵角が大きくなるほど旋回半径は小さくなる事が確認され，ステアリングが機能していることが確認された．また速度を変化せたときは，速度が上昇するほど旋回半径が大きくなり，慣性による旋回半径の増大が確認された．これらの結果は予想される車両の動きと一致しており，それらしい車両の動きを再現することが出来たといえる．

4.2　実際の車両との比較

走行速度が 0.2[m/s]と 0.1[m/s]の時の実際の車両の軌道とシミュレータ上の車両の軌道を図７に示す．実際の機体の軌道と比較した場合，シミュレートした車両の旋回半径の方が実際の車両に比べて大きくなった．各パラメータを操作してシミュレートを行い実際の軌道に近い動きを行うことができるかどうかを検討したところ，重量及び重心位置が軌道に影響を与える事，図８に示す通り舵角が 13.5°の時に実際の軌道に極めて近い軌道を描くことが確認された．実際の軌道との間に誤差が生じた原因として，現実の車両の舵角の計測について精度が十分とはいえなかったためにシミュレータ上の舵角との間に誤差が生じてしまったこと，重量・重心が起動に影響を与えていることからこれらの設定が適切でなかったほか実際には存在したトレーラーを無視したことなど，現実の環境を再現しきれていなかったことに起因するものであると考えられ，実際の車両の正しい状態を確認もしくは想定する方法が必要であると考えられる．

4.4　機体の改造

右側のキャスターの摩擦係数を変化させた時の軌道を図９に，左側のキャスターの摩擦係数を変化させた時の軌道を図１０に示す．　それぞれの図に示された通り摩擦係数を設定したキャスターの側に軌道がずれている事が確認され，摩擦係数が大きくなるほど軌道のずれが大きくなる事が確認された．これらの結果は当初想定していた通りの結果であり，これにより適切な設定を行えば機体の改造を容易に行うことができる事が示された．課題としてどのように適切な条件を決定するかが考えられる．

5.　結言

シミュレータ上に再現した車両を安定して動作させることに成功した．またその動作も物理法則から大きく外れたものにはならなかった．しかし，実際の車両との比較を行った場合，大きな誤差が発生した．今後はこれらの現実との乖離を改善するために入力する車両と周辺の状況に改良を加えるほか，実際の車両の正しい状況を把握するための方法が必要であると言える．

文献

(1)　“ROS　wiki”（2018），http://wiki.ros.org/，(最終検索日：2018 年 2 月 5 日).

(2)　金田浩明，“ROS で始めるロボティクス（1）　ー The Robot Operating System”(2016)，BRILLIANTSERVICE TECHNICALBLOG，http://bril-tech.blogspot.jp/2016/10/ros1-robot-operating-system.html，(最終検索日：2018 年 1 月 19 日).

(3)　RyodoTanaka，“No.6-7:Gazebo を ROS に繋ぐ (ROS Control 編)”(2015)，九州工業大学 CIR-KIT Blog，http://cir-kit.github.io/blog/2015/02/22/gazebo-ros-control/，（最終検索日：2018 年 1 月 19 日）．

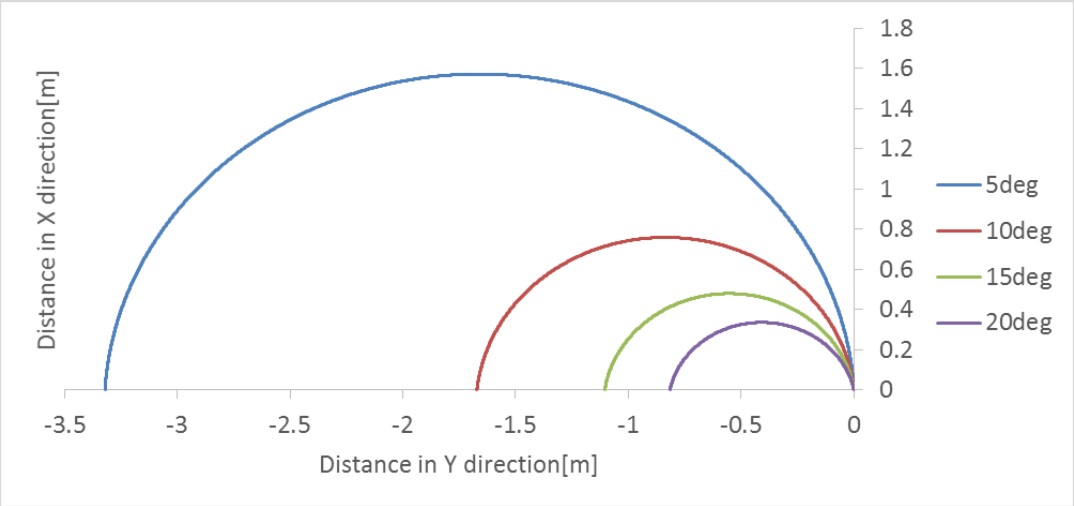

Fig.5 Comparison of orbits 1.

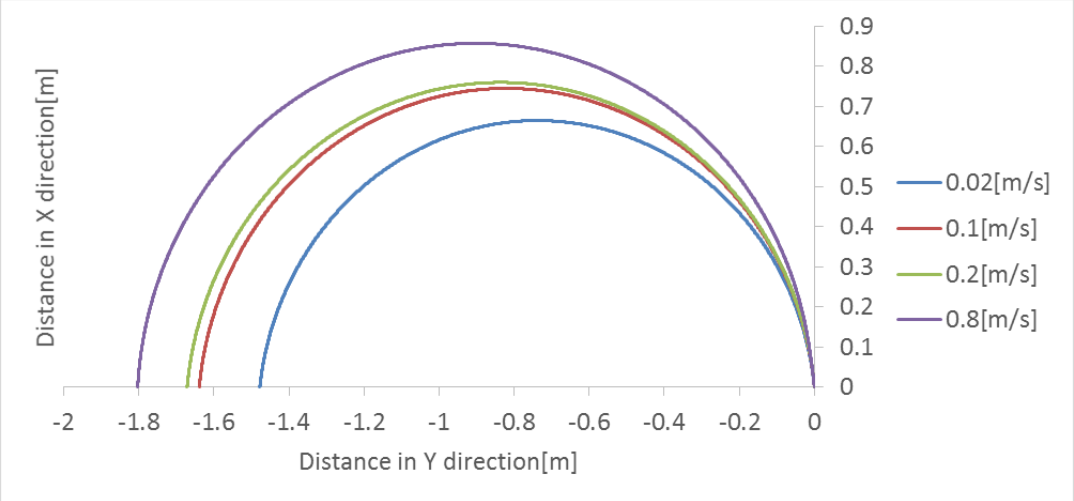

Fig.6 Comparison of orbits 2.

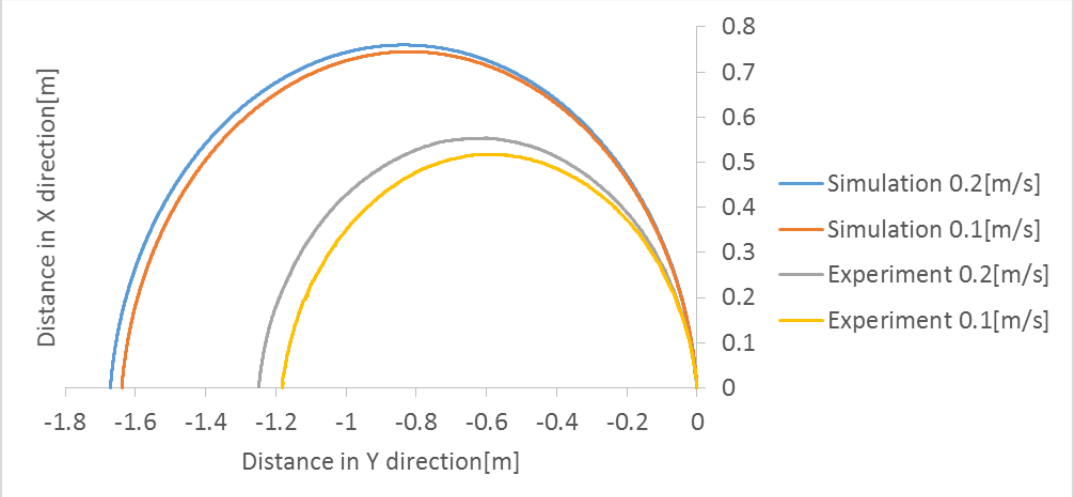

Fig.7 Comparison of orbits 3.

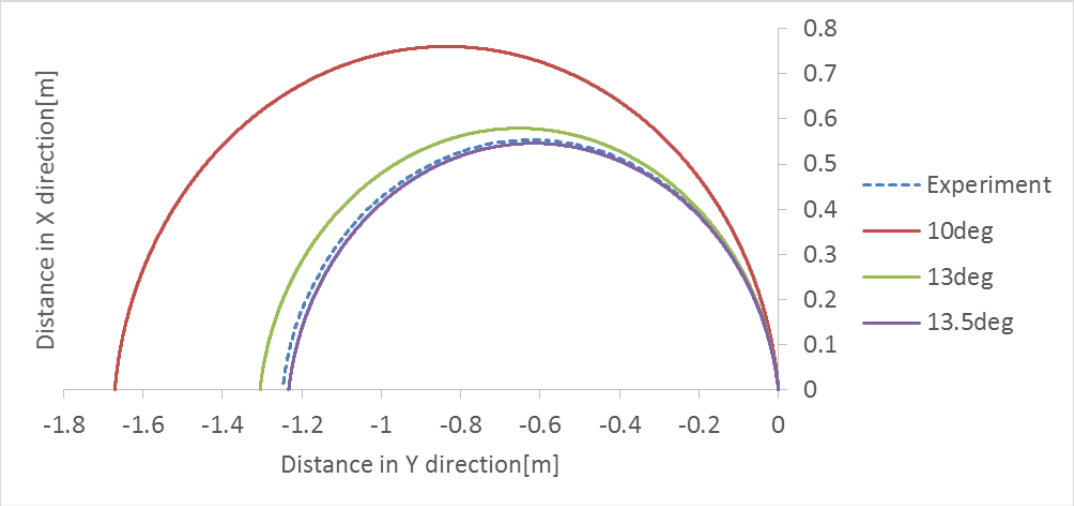

Fig.8 Comparison of orbits 4.

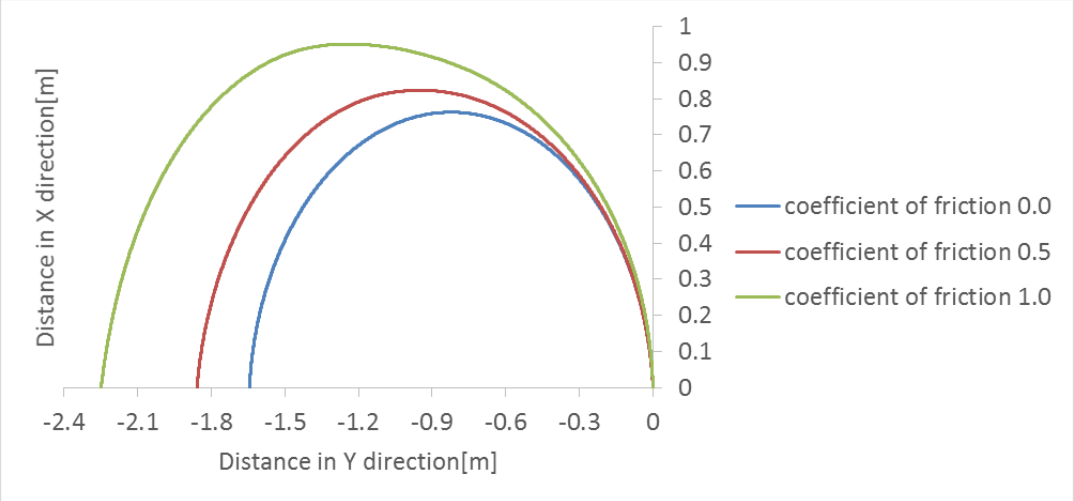

Fig.9 Comparison of orbits 5.

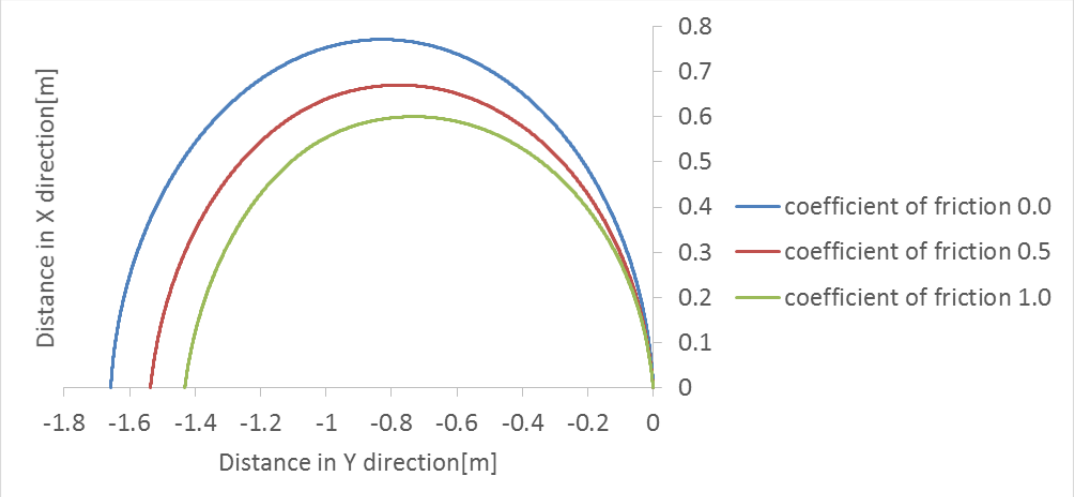

Fig.10 Comparison of orbits 6.