

乗り心地を重視した車両制御システムの設計とシミュレーション

服部有里子, 青木慎吾, 小林 颯

筑波技術大学 産業技術学部 産業情報学科

要旨: 自動車の機能は、走る・止まる・曲がるに加えて乗り心地を考慮する必要があり、本研究では車の車両速度制御について自動運転制御により、乗り心地を重視した車両制御システムを開発した。「乗り心地を重視した車両制御システム」では、走行中に車両速度を追従制御し、ドライバがアクセルペダルを操作しなくても自動的に車両速度を一定に維持できるクルーズコントロールシステムを構築する。本稿では、PID 制御のパラメータ調整法を適用し、制御対象である自動車の乗り心地を重視することにより、車両制御の安定化を図った制御システムの設計とシミュレーションについて報告する。

キーワード: クルーズコントロールシステム, PID 制御, フィードバック制御, 自動走行運転

1. はじめに

交通事故死者削減・渋滞低減のため、自動車の自動走行・高度運転支援技術の開発が進められている。車の機能は、走る・止まる・曲がるに加えて乗り心地を考慮する必要があり、本稿では自動車の車両速度制御について自動運転制御により、乗り心地を重視した車両制御システムを開発した。ここでは、車両速度が急変化することなく、安定した車速での走行を制御することで乗り心地がよくなると考える。

「乗り心地を重視した車両制御システム」では、走行中に車両速度を追従制御し、ドライバがアクセルペダルを操作しなくても自動的に車両速度を一定に維持できるクルーズコントロールシステムを構築する。クルーズコントロールでは、車両速度を動的に計算し、最適なペダル開度の設定による、安定した乗り心地のよい運転制御システムの開発が重要である。この場合、要求されるシステムの課題は以下の2点である。

- 車両の速度変化を動的に計算し、車両速度を一定に維持することで、車速が急変化することなく、走行安全性を優先すること
- 外乱がある場合もクルーズコントロールすることにより、振動を最小限に抑え、即時に車両速度を追従制御すること

これまで車両制御システムとして、フィードバック制御系の PID 制御 (Proportional, Integral and Differential 制御: 制御偏差に基づいて制御入力信号を生成する制御) のパラメータ調整方法が提案されている [1, 2]。しかし、実際の車両モデルを考慮した場合と比較すると、たとえ正確にモデリングできたとしてもモデル誤差が少なからず存在する。本

稿では、モデル誤差について PID 制御のパラメータ調整法を適用し、制御対象である自動車の乗り心地を重視することにより、車両制御の安定化を図った制御システムの設計とシミュレーションについて報告する。具体的には、車の自動速度制御を行う、フィードバック制御系として PID 制御を設計した。シミュレーションにより検証・評価を行い、自動走行運転の速度制御による乗り心地向上、及び快適性向上について、サービスの有効性を評価した。

2. 関連研究

車両制御システムとしては、フィードバック制御系の PID 制御のパラメータ調整方法が提案されている。しかし、実際の車両モデルを考慮した場合と比較すると、モデル誤差が少なからず存在する。また、車両重量、外乱による環境の変化による各パラメータの変動等を考慮すると、モデルの安定性は重要な要素であると考えられる。機械系において PID 制御は広く利用されているが、対象が時変系であるために PID パラメータの調整がうまくできない場合が往々にして存在している。本稿では、そのような問題に対してアドバンス制御であるセルフチューニング法やロバスト制御の観点から PID パラメータを調整する方法について研究した。

3. 問題設定と要求条件

3.1 PID 制御

本研究では、フィードバック制御系として機械系で広く利用されている PID 制御を採用する。PID 制御は入力値の制御を出力値と目標値との偏差、その積分及び微分の3つの要素によって行う方法である [3]。

3.2 パラメータ調整

PID 制御の各パラメータの値 (Kp, Ki, Kd) は設計値であるため、制御対象と独立に自由に与えることができ、要求を満たす最適値を決定する必要がある。しかし、一般的な制御対象に対して、最適なゲイン値を与える代表的な方法は存在せず、実際の調整は数値シミュレーションを行ったり、実際の対象物に調整器を繋いで試行錯誤的に最適値を得ている。

パラメータ調整の経験則として、ジョン・ジエグラ (John G. Ziegler) とナサニエル・ニコルス (Nathaniel B. Nichols) らにより提案された2つの調整法、ジエグラ・ニコルスの限界感度法とジエグラ・ニコルスの過渡応答法が知られている [4]。

3.3 要求条件

本稿で提案する乗り心地を重視した車両制御システムでは、これらの課題と特性に対応した安定性の高いシステムが要求される。ここで、システムが満たすべき要求条件を以下にまとめる。

- ① アクセルペダルを踏まなくても一定の速度を保って走行するシステムであること
- ② 横風などの外乱を受けた場合も外乱による操舵角や位置のずれを算出し、走行方向を維持するシステムであること
- ③ ブレーキを踏み込むと、クルーズコントロールが解除されること

4. 乗り心地を重視した車両制御システムの提案

前章で整理した要求条件を満足するシステムとして、乗り心地を重視した車両制御システムを提案する。

車両制御システムは、エンジン・トルクコンバータ (T/C) ・

車両側・変速制御の4つのサブシステムから構成される。

エンジンにて、ペダル開度と現在のエンジン回転数を入力としてエンジントルクを計算する。エンジントルクは T/C にて出力トルクとして伝達され、余った余剰トルクはエンジン回転数の回転上昇に使用される。T/C にて計算した出力トルクから回転角加速度を計算し、回転角速度から速度に変換し、それを変速制御に送り込む。変速制御は、適切なタイミングでどのギヤ段を使うかを判断する装置である。車両制御システムのシステム構成を図 1 に示す。

5. フィードバック制御系の PID 制御の設計

5.1 エンジン

エンジントルク [Nm] は、シミュレーションスタート後はデータマップと無関係に 0.2 秒間、100 [Nm] のトルクを発生する。0.2 秒経過した後はデータから計算した値に切り替える。

ペダル開度 [%] と現在のエンジン回転数 N_e [rpm] を入力として、2 次元関数を近似するブロックでエンジントルク [Nm] を出力する。

5.2 トルクコンバータ

5.2.1 釣り合いトルク・出力トルクの計算

エンジン回転数 N_e [rpm], 出力回転数 N_{out} [rpm] として、以下の数式を計算する。

$$\text{駆動側速度比: } e = N_{out} / N_e \quad [0 \sim 1]$$

$$\text{被駆動側速度比: } e' = N_e / N_{out} \quad [0 \sim 1]$$

ただし、 $N_{out} = 0$ のとき、 e' は 0 とする。

$$\text{容量係数: } c = f1(e) \quad [\text{Nm}/\text{rpm}^2]$$

$$\text{トルク増幅比: } T = f2(e) \quad [\text{Nm}]$$

$$\text{釣り合いトルク駆動側: } T_{out1} = c \times N_e^2 \quad [\text{Nm}]$$

$$\text{釣り合いトルク被駆動側: } T_{out1}' = -c \times N_{out}^2 \quad [\text{Nm}]$$

$$\text{出力トルク: } T_{out2} = T_{out1} \times T \quad [\text{Nm}]$$

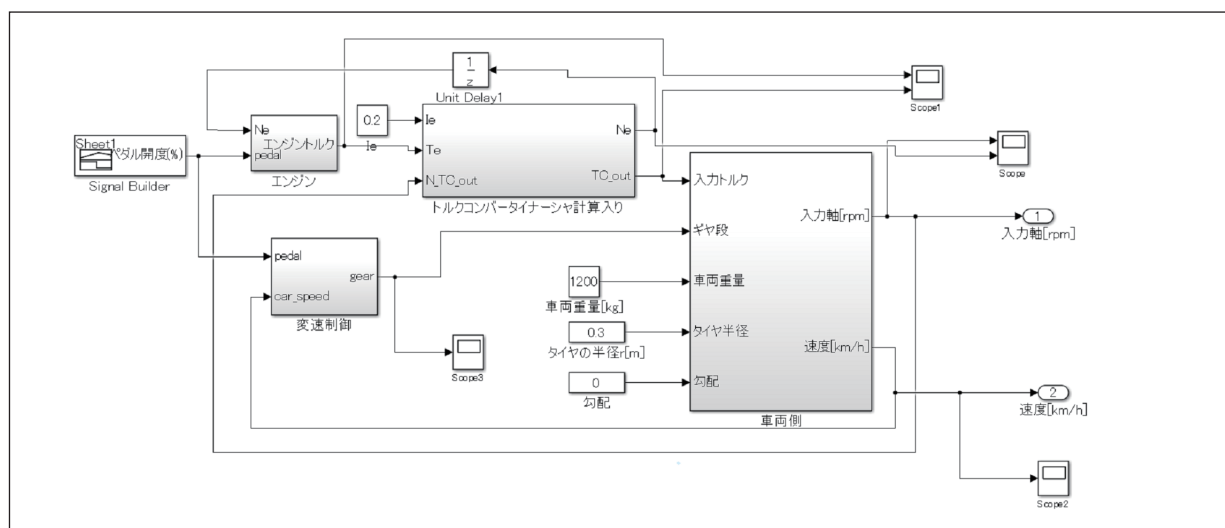


図1 乗り心地を重視した車両制御システムの構成

5.2.2 トルクコンバータのモデル

エンジン回転数 Ne [rpm] は, T/C に入力されるトルク Te [Nm] と釣り合いトルク $Tout1$ [Nm] との差を使用して計算する。入力トルク Te は出力トルクとして伝達され, 余った余剰トルクはエンジン回転数 Ne の回転上昇に使用される。 Ne は次の式で計算する。

$$Ne = \int \frac{Te - Tout1}{Ine} dt \quad [\text{rpm}]$$

Ine : 入力イナーシャ [kgm^2] (Ine を $0.2[\text{kgm}^2]$ とする)

5.3 変速制御

変速制御は, 速度などに応じてシフトアップ・シフトダウンするかを判断する制御である。

前回のギヤ段やペダル開度, 車両速度を基に新たなギヤ段へシフトする (図 2)。ペダルを踏み込み速度が上がる時, shift1-2UP 線を通過するとギヤ段が 1 から 2 へと変わる (①)。同様に shift2-3UP 線を通過するとギヤ段が 2 から 3 へと変わる (②)。shift3-4UP 線を通過するとギヤ段が 3 から 4 へと変わる (③)。ペダルを緩めると速度が下がってくるため, shift4-3DOWN 線を通過するとギヤ段が 4 から 3 へと変わる (④) という構造である。

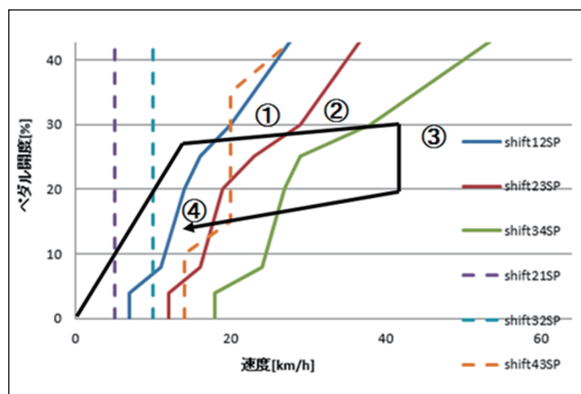


図 2 変速制御のギヤ段シフト

5.4 車両側

車両側サブシステムは, 入力トルク [Nm], ギヤ段, 車両重量 [kg], タイヤ半径 [m], 勾配 [rad] の 5 つの入力から入力軸回転数 [rpm] と速度 [km/h] を算出する。

5.4.1 走行抵抗の計算

走行抵抗 R [N] には, 空気抵抗 $R1$ [N], 勾配抵抗 $R2$ [N], 道路抵抗 $R3$ [N] の 3 つの抵抗がある。

$$R = R1 + R2 + R3$$

$$R1 = \text{抵抗} \times \text{車両速度}^2$$

$$= k \times S \times \rho \times \left(v \times \frac{1000}{3600} \right)^2 \times \frac{1}{2}$$

k : 空気抵抗係数 S : 前面投影面積 [m^2] ρ : 空気密度 [kg/m^3]

$$R2 = m \times g \times \sin \theta$$

$$R3 = m \times g \times \cos \theta \times C_{rr}$$

m : 車両重量 [kg] g : 重力加速度 [m/s^2] θ : 勾配 [rad]

C_{rr} : 転がり抵抗係数 (アスファルトの場合は 0.01)

5.4.2 出力軸回転数の計算

車両重量 m [kg], タイヤ半径 r [m], 走行抵抗 R [N], 入力トルク T [Nm] から出力軸回転数 n [rpm] を計算する。車両重量, タイヤ半径から回転慣性モーメントであるイナーシャ I [kgm^2] を求め, イナーシャとトルクから出力回転数を割り出すことができる。

イナーシャ I [kgm^2] の式は次のとおりである。

$$I = m \times r^2 \quad m: \text{車両重量 [kg]} \quad r: \text{タイヤ半径 [m]}$$

走行抵抗によって, 外からタイヤにトルクがかかり, 損失トルクが発生する。よって, 車両からタイヤに入るトルクから損失トルクを考慮して, タイヤにかかる正味のトルク (損失込みトルク) を計算する必要がある。

損失込みトルク $Tout$ [Nm] の式は,

$$Tout = T - Tr$$

T : タイヤに入るトルク [Nm] Tr : 損失トルク [Nm]

次にイナーシャ I と損失込みトルク $Tout$ から回転角加速度 a [rad/s^2] を求める。式は次のとおりである。

$$a = \frac{Tout}{I}$$

回転角加速度 a [rad/s^2] を積分して角速度 [rad/sec] を求め, 出力軸回転数 n [rpm] を計算する。式は次のとおりである。

$$n = \int a dt \times \frac{1}{2\pi} \times 60 \quad [\text{rpm}]$$

5.4.3 速度計算

出力軸回転数 n [rpm] とタイヤ半径 r [m] から速度 V [km/h] を計算する。式は次のとおりである。

$$V = n \times (2 \times \pi \times r) \times \frac{1}{1000} \times 60 \quad [\text{km/h}]$$

5.4.4 車両側モデル

これまで計算したモデルを組み合わせることで, 図 3 に示す車両側モデルが構築できる。

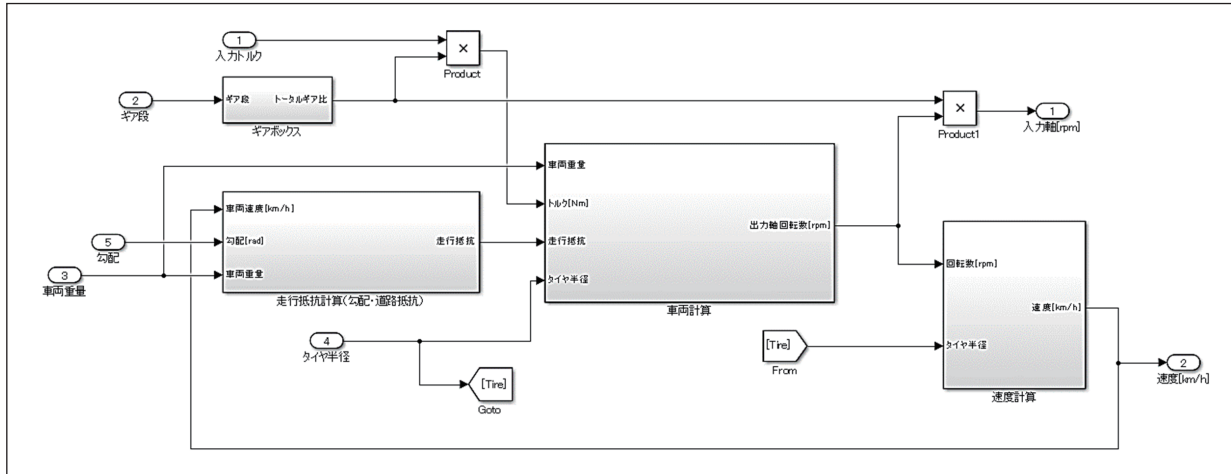


図3 車両側モデル

6. シミュレーションによる検証・評価

6.1 フィードバック制御系の検証・評価

6.1.1 P・PI・PID 制御の評価

フィードバック制御系を用いてペダル開度 [%] を制御し、次の要求仕様を満たすパラメータを設定する。

要求仕様：20 秒後に 60[km/h] の速度に到達すること。

P 制御・PI 制御・PID 制御モデルを作成する。パラメータ値は、 $K_p=4$ 、 $K_i=1$ 、 $K_d=0.1$ とする。また、外乱の影響はないものとして外乱の計算は省く。作成した P・PI・PID 制御モデルをそれぞれ車両モデルに取り付けてシミュレーションを実行した [5]。目標値の設定として、ステップ時間を 20[s]、初期値を 60[km/h]、最終値を 60[km/h] とした。

3つの制御系を使用してペダル開度 [%] と速度 [km/h] を調べた結果、PI 制御と PID 制御による結果はほとんど同じになっていることがわかった。P 制御については初期の加速する部分は緩やかになっているが、要求仕様である「20 秒後に 60[km/h] の速度に到達すること」に達していないことがわかる。P 制御の速度グラフを見てみると、58[km/h] あたりにとどまっていることから、P 制御では目標速度に収束することが難しいと考える。一方、PI 制御と PID 制御では 20 秒後丁度に目標設定である 60[km/h] に収束していることがわかる。ただし、初期の加速する部分は急になっており、5 秒以内に 60[km/h] に達しているため、乗り心地が良くないだろうと考える。

そこで、加速する部分のグラフが緩やかになるように、表 1 に示すパラメータ値を入れ替えてシミュレーションを実行した。今回は PI 制御と PID 制御の結果がほとんど同じであるため、パラメータが 1 つ少ない PI 制御を使用した。

6.1.2 PI 制御の評価

PI 制御において、表 1 のパラメータ値を用いてシミュレ-

表 1 PI 制御のパラメータ値

No.	K_p	K_i
1	6	1
2	2	1
3	4	10
4	4	0.1

ーションを実行した。まず、 $K_p=6$ 、 $K_i=1$ と $K_p=2$ 、 $K_i=1$ のように元のパラメータ値 $K_p=4$ 、 $K_i=1$ において K_p だけ数値を変えて実行した結果、 K_p の数値が大きいほど急速に目標値に近づくことがわかった。逆に、小さいほど緩やかになっている。ただし、この値が小さいほど目標値を大きく超えてしまい、さらに元に戻そうとしても目標値を行き過ぎてしまい、結果的に振動が起きてしまうことがわかった。また、 $K_p=4$ 、 $K_i=10$ と $K_p=4$ 、 $K_i=0.1$ のように K_i の数値だけ変えて実行した結果、 K_i の数値が大きいほどグラフの波形の振幅が大きくなることがわかる。つまり、 K_i の数値が小さいほど収束が早くなるということがいえる。

以上の4ケースのパラメータ値から、加速が緩やかかつ収束が早いグラフにするためには、 K_p と K_i の数値を両方も小さくする必要があると考える。ただし、どちらかが一方的に小さくなりすぎるとグラフが乱れてしまうため、2つのパラメータ値を用いてうまく調整する必要がある。4ケースのパラメータ値を基にいろいろな数値を入れて試行錯誤を重ねた結果、 $K_p=2$ 、 $K_i=0.1$ が最も適した数値であると推測した。その結果を図 4 に示す。

安定走行するまでに 20[sec] かかるが、オーバーシュートや振動がないことが確認できる。実際のシステムで過度な速度変化は車両の損傷や安全性や乗り心地に関わる点であるため、その観点からこの結果は良いといえる。

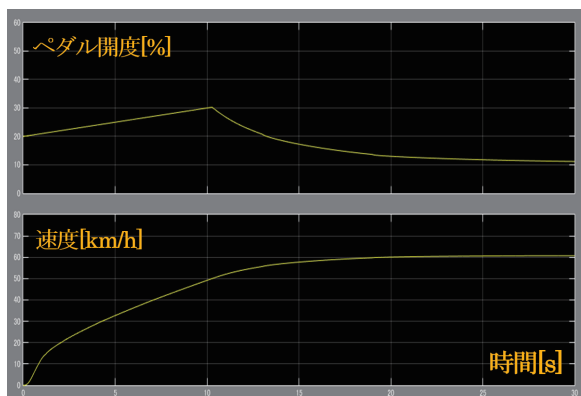


図4 PI制御の結果 ($K_p=2$, $K_i=0.1$ のとき)

6.1.3 パラメータ値を変えずに加速を調整する方法

次に、パラメータ値を変えずに加速度を調整する別の方法を考える。Saturation ブロックを使用して入力信号を上限と下限の飽和値に制約する方法である。これまでの実験では Saturation ブロックにおける上限は 10、下限は -10 である。今回の実験では、上限を 5、下限を -5 に変えて PID 制御のフィードバック制御系を使用する。その結果を図 5 に示す。上限 10、下限 -10 の場合よりも加速度が緩やかになっていることがわかる。その理由は、上限 10、下限 -10 の場合と比べて上限 5、下限 -5 の方がフィードバックの回数が多くなり、目標値に近づくのに時間がかかるため、加速度が緩やかになっていると考える。

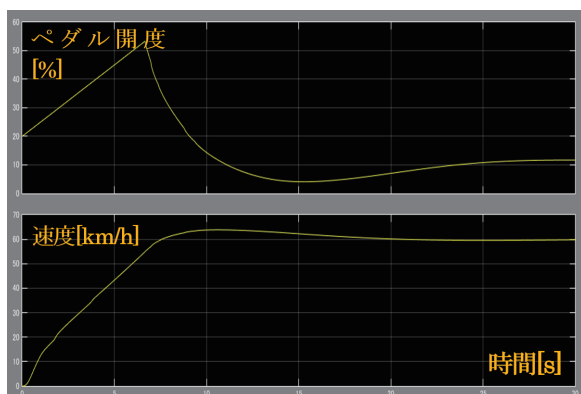


図5 PID制御の結果 (上限5, 下限-5のとき)

6.2 PI 制御と PID 制御の比較

これまでの実験で PI 制御と PID 制御におけるシミュレーション結果がほとんど同じだった。PI 制御における結果を見ると、速度 [km] の波長の振動があまり見られず、収束も早くなっていることから、PID 制御に変えても些細な変化に気づいていない可能性があるということである。そこで、意図的に PI 制御のパラメータ K_p , K_i の数値を変えて波長の振幅を大きくさせ、その状態で PID 制御に変えて波長の変化を調べる実験を行った。 K_i の数値が大きくなるほど振幅

が大きくなるため、 $K_p=1$, $K_i=5$, $K_d=0.1$ とした。シミュレーション実行時間は 300[s] とした。図 6 に PI 制御の結果、図 7 に PID 制御の結果を示す。

図 6 と図 7 を比較すると、150[s] あたりのところで振幅の大きさがわずかながら図 7 の PID 制御の方が小さいことがわかった。波長の振幅の大きさが小さいほど収束は早くなるため、PID 制御の微分動作 (D 動作) は波長の振動を抑制する性質を持っているといえる。

次に、 K_d の数値を大きくしてシミュレーションを実行した。 $K_d=3$ とした結果を図 8 に示す。 K_d の数値が大きいくほど、波長の振幅の大きさが小さくなってきており、収束も早くなっていた。

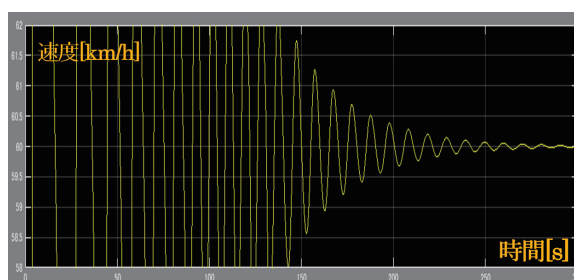


図6 PI制御の結果 ($K_p=1$, $K_i=5$ のとき)

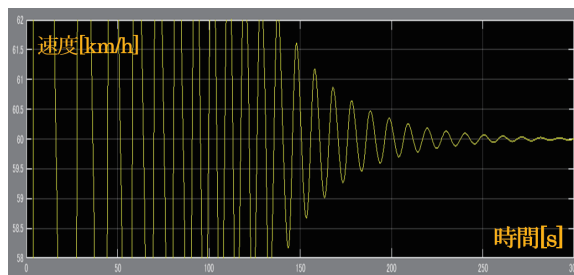


図7 PID制御の結果 ($K_p=1$, $K_i=5$, $K_d=0.1$ のとき)

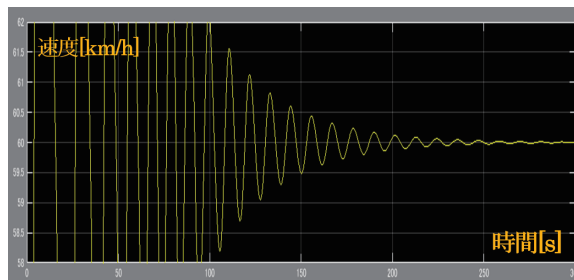


図8 PID制御の結果 ($K_p=1$, $K_i=5$, $K_d=3$ のとき)

PID 制御は、目標値と現状の制御量との差 (偏差) に比例して操作量を変化させる比例動作 (P 動作) と、偏差をなくそうとする性質を持つ積分動作 (I 動作) と、波長の振動を抑える性質を持つ微分動作 (D 動作) を組み合わせることで、偏差がより少なく収束がより早く安定した応答が実現できると考える。

6.3 外乱がある場合の検証・評価

外乱がある場合、P・PI・PID 制御においてどのフィードバック制御系が最も適しているのか検証を行った。外乱として、スタート時点から坂道を走行するものとする。ペダル開度 [%] と勾配 [°] を設定し、勾配に入っても速度が下がらないようにフィードバック制御を行った。

$K_p=4$, $K_i=1$, $K_d=0.1$ とした。3つのシミュレーション結果から、P 制御において最初は目標値 (60[km/h]) に向かって速度を上げようとしているものの、勾配という外乱が原因なのか 30[km/h] にさえも達しておらず、外乱がない状態よりもかなり偏差が残っていることがわかる。このことから、P 制御では外乱がある場合は速度をコントロールすることが難しいと考える。

PI 制御と PID 制御において、ほとんど同じ結果となった。いずれも 50[s] までに目標値である 60[km/h] に収束しているように見えた。本当に収束しているかまたどのような波長になっているのか確認するために速度の表示範囲を 59[km/h] ~ 61[km/h] に設定した。PI 制御と PID 制御において違いはほとんど見られなかった。PI 制御において波長の振動もほとんど見られず、振動を抑制する役目を果たす D 動作を加えても変化が起きないことが原因だと考える。いずれも目標値である 60[km/h] に向かって収束しつつあるが 300[s] になってもまだ偏差 (-0.2 ~ 0) が残っていることがわかった。偏差があるため、偏差をコントロールできる I 動作の数値を $K_i=5$ に変更した。 K_i の数値を大きくした結果、より振動的になったが 0.03 程の偏差は残っているものの、目標値である 60[km/h] により近づいた。波長の振動を抑えるために、D 動作を $K_d=1$ に変えてみた結果、より波長の振幅の大きさが小さくなった。図 9 に PI 制御の結果、図 10 に PID 制御の結果を示す。波形からは、外乱に対してはかなりの強さを持っていると考えられる。

ただし、各所に見られる振動はギヤの変化による影響と見られる。外乱の変化が激しいほどギヤ段の変化も激しくなり、振動を与えることなく安定した一定の速度を保つようにするためには、坂道を走るときエンジブレーキをかけるというように坂道対象の制御システムを設計する必要があると考える。

7. おわりに

本稿では乗り心地を考慮した車両制御システムを提供するため、フィードバック制御系の P 制御・PI 制御・PID 制御において、各パラメータ K_p , K_i , K_d の数値及び車両速度の上下限の数値の調整方法について研究し、シミュレーションにより検証・評価した。PI 制御と PID 制御については、D 動作を加えることによって少しでも波長の振動を抑えることができるため、どのような制御対象においても PID 制御を



図9 PI制御の結果(外乱あり, $K_i=5$ のとき)



図10 PID制御の結果(外乱あり, $K_i=5$, $K_d=1$ のとき)

使用することが望ましいことを確認した。ただし、PID 制御は PI 制御に比べて位相の遅れが生じることがデメリットである。外乱がある場合は、PID 制御において外乱がない場合よりも偏差が生じやすくなるため I 動作を大きくする必要がある。

今後は、各パラメータ K_p , K_i , K_d の数値の調整において、最適な数値を算出する方法としてジエラ・ニコルスの2公式と呼ばれる限界感度法と過渡応答法の2つの方法についてさらに研究し、本稿のシミュレーション評価との差異について検討する所存である。

参考文献

- [1] 佐藤和也, 平元和彦, 平田研二. はじめての制御工学. 講談社, 2010.
- [2] 佐伯正美. 制御工学 -古典制御からロバスト制御へ-. 朝倉書店, 2013.
- [3] 森泰親. 制御工学. コロナ社, 2001.
- [4] 山本重彦, 加藤尚武. PID制御の基礎と応用. 朝倉書店, 2005.
- [5] 久保孝行. 自動車業界 MBD エンジニアのための Simulink 入門. TechShare, 2012.

Design and Simulation of Vehicle Control System Considering Ride Quality

HATTORI Yuriko, AOKI Shingo, KOBAYASHI Hayate

Department of Industrial Information, Faculty of Industrial Technology,
Tsukuba University of Technology

Abstract: The functions of a vehicle are to run, stop, and turn; further, it is necessary to consider ride quality. In this study, a vehicle control system was developed focusing on ride quality, with regard to vehicle speed control by automatic driving control. "The vehicle control system considering ride quality" controls vehicle speed, which is overtaken during driving, and then the cruise control system is developed, which maintains vehicle speed fixedly automatically, even if the driver does not operate the accelerator pedal. In this study, a parameter adjustment system of PID (proportional, integral and differential) control was applied. We report on the design and simulation of the vehicle control system, which was aimed at the stabilization of vehicle control by considering vehicle ride quality as the control target.

Keywords: Cruise control system, PID (Proportional, Integral and Differential) control, Feedback control, Automatic run driving