

車両挙動モデルの開発に向けて

5年A組 村田 宏暁

指導教員 米田 隆恒

1. 要約

私は道路交通における交通渋滞の発生原理や性質について調べるため、JAVA 言語を用いてマイクロ道路交通シミュレーターを開発し、研究を行っている。しかしこれまでに開発したシミュレーターは精度が不十分であったため、実際の車両の挙動データを測定し、そのデータを元に開発した車両挙動モデルを用いることで、精度の高いシミュレーターを開発しようと考えた。未だ測定には至っておらず、車両挙動の測定へ向けて準備している段階であるが、現状と今後の方針について述べる。

キーワード 交通渋滞、シミュレーション、モデル、JAVA、OBD2

2. 背景

現在、道路交通は移動や輸送において重要な役割を担っている。しかし、交通集中などによって渋滞が発生すると、所要時間の延伸や事故率の上昇など様々な問題が生じる。これを解決するためには渋滞の発生原理や性質を知ることが不可欠であるが、実際に実験を行うことは困難であるためシミュレーションによって擬似的に実験を行う必要がある。

私はこれまでも独自に開発したシミュレーターを用いて道路交通についての研究を行ってきたが、用いていた車両の挙動モデルはシンプルなもの精度が不十分であった。私は実際の車両挙動のデータをモデル化することで精度の高い車両挙動モデルが開発できるのではないかと

と考え、車両の挙動を測定することにした。

3. 研究内容

車両挙動を測定するためには、実際に測定する情報を決定する必要がある。今回は、運転者は車間距離、速度、前方車の速度の三つの要素を元に一定時間後の加速度を定めるものとして、これらの四つの値を測定することにした。以下でそれぞれの測定方法について述べる。

測定は車外から測定する方法と車内から測定する方法の二つが考えられるが、昨年度の論文で述べたように車外からの測定では精度を確保できないと考えられるため、車内から測定する方法を採用する。

3.1 車間距離の測定

第一にレーザー距離計を用いることが考えられるが、市販のレーザー距離計は測量を目的としているため連続して測定することが困難であり、また、窓や黒い車体はレーザーを反射しにくく精度が悪くなってしまうため、車間距離の測定には適さない。そのため、二つのカメラ(カメラ A とカメラ B)で前方車を撮影し、車間距離を算出する。

通常、観測地点の違いによる画像のずれから距離を求める場合は特定の点についての視差から距離が算出されるが、二つのカメラから得られた画像の中から同一の点を見つけることは困難である。そこで「物体とカメラの位置と向き、及びカメラの画角が既知である場合、カメラに物体がどのように映るかを求めることができる」ことを利用し、前方車の後面をカメラ A の向きに垂直な平面と仮定して、次の①～⑤の処理を行うことで距離を求める方法を考えた。

- ① カメラ A の画像から前方車の後面を切り出す
- ② カメラ A と後面との距離を特定の値 a と仮定し、後面の位置を定める
- ③ カメラ B に写る画像をカメラ A の画像を元に算出する
- ④ ③で求めた画像と実際にカメラ B から得られた画像を比較し、一致度を算出する

⑤ a を変化させて②～④を繰返し、最も一致度の高くなる a を距離とする
しかし上記の方法をそのまま実装すると、計算量が多いため処理に時間がかかってしまう。上の方法ではカメラ B の向きや画角及び位置を一般化しているため計算式が複雑であるが、二つのカメラの向きと画角及び前方車の後面との距離と高さが等しいものとする、カメラ A の画像の後面部分を距離に応じて左右に移動させるだけでカメラ B の計算上の画像の後面部分を求められ、処理時間が短くなるためこの方法を採用する。また、距離を定めて一致度を算出するのではなく、左右の移動量を b ピクセルと定めて一致度を算出し、最後に距離に変換することで処理を簡素化した。

なお、一致度は処理①で切り出した画像のピクセル数を n として以下の式で算出する。

$$\frac{\sum_{k=1}^n \sum_{l=1}^3 (256 - |X_{k,l} - Y_{k,l}|)}{3 \cdot 256 \cdot n}$$

ただし、 X_k は切り出した画像の k 番目のピクセルを、 Y_k は画像 B の X_k と重なるピクセルを表しており、 $X_{k,1}, X_{k,2}, X_{k,3}$ はそれぞれ X_k の色の R,G,B の成分を 0 以上 256 未満の整数で表したものである。 $Y_{k,1}, Y_{k,2}, Y_{k,3}$ も同様である。

しかし、この処理を行うにはカメラから得られた画像のどこに前方車が映っているのかが既知である必要があり、処理

の際に手動で設定するため誤差が生じる可能性がある。また、カメラの位置や向きが既知である必要があるため、エンジンによる振動などによってカメラの位置や向きがほとんど影響を受けないように固定する必要があるが難しく、未だ測定に至っていない。

3.2 速度の測定

GPS を用いる方法も考えられるが、データを得られる間隔が 1 秒程度と長く、連続測定には適さない。

そこで、OBD2 を用いる。OBD2 とは車両の自己故障診断のために ECU と呼ばれる車両内部のコンピュータと通信する際の通信規格であるが、故障情報の他にエンジンの回転数や速度などの情報を得ることもできる。車両に装備されている OBD2 コネクタに図 1 の機器を接続し、Android 端末と当該機器との間で Bluetooth を用いた通信を行うことで、間接的に車両と通信を行い、速度を得る。



図 1 ELM327 搭載の OBD2 アダプタ

ただし、得られる速度は車載の速度計と同じであり、速度計は実際の速度より大きな値を示すよう作られていることが多いため、系統誤差が発生する。また、得られる値は 1 km/h 単位である。

Android アプリを開発して通信を行い、速度が得られることは確認できた。しかし、受信したデータの中に意味の判明しないデータが含まれており、受信データから速度への変換を自動化できていない。

3.3 前方車の速度の測定

測定車の速度と、前方車の車間距離の変化率から前方車の速度を求める。

3.4 加速度の測定

加速度は、スマートフォン(Android)の加速度センサを用いて、独自に開発したアプリで測定、記録する。

しかし、測定に用いた Android 端末(ISW12HT)の加速度センサは重力と鉛直上向き方向の加速による慣性力とを区別できず、常に重力の影響を受ける。道路が水平であれば問題は発生しないが坂道である場合は実際の加速度と異なる値が得られるため、重力の影響を除去する必要がある。

3.5 後方車の挙動測定

車両の挙動は運転者によって変わる。特定の運転者の挙動に強く影響されたモ

デルでは実際の道路交通を再現できないため、運転者集団全体の特徴を反映したモデルが求められる。しかし、それを開発するには様々な運転者の挙動を測定する必要がある。ここまでで述べた測定方法は機器を搭載した測定車自身の挙動を測るものであり、運転者に協力していただく必要があるため、多くの運転者の挙動データを得ることは難しい。そこで、測定車の後方を走行している車両の挙動を測定することで、運転者に特別な協力をいただくことなくその挙動データを得ることを考えた。

後方車から見た「車間距離」は測定車と後方車との間の距離であるからカメラを後方に向けて設置することで、3.1 で述べたのと同様に測定することができ、「前方車の速度」は測定車の速度であるから 3.2 と同様に測定できる。また、後方車の「速度」は車間距離の変化率から相対速度を求めてそれに測定車の速度を加えることで得られる。後方車の「加速度」は 3.4 のように加速度センサを用いて直接測ることはできないが、相対速度の変化率に測定車の加速度を加えることで求められる。しかし、車間距離の測定方法を確立できていないため、未だ測定には至っていない。

4. 今後の課題

まずは車間距離の測定を行い、誤差などを調べる必要がある。また、測定したデータからモデルを開発するためには加速度とその他のデータとの関連を調べる必要があるが、カメラの時計とスマートフォンの時計との間にずれがあると正確な関係を把握することができない。これらの時計を精密に同期させる方法を考える必要がある。

5. 参考文献

- [1]「交通工学」飯田恭敬 監修、北村隆一 編著、オーム社 版(2008年)
- [2]「Java Platform, Standard Edition 6 API仕様」
docs.oracle.com/javase/jp/6/api/
- [3]「Android Developers」
developer.android.com/
- [4]「ELM327 v2.1 data sheet」
www.elmelectronics.com/DSheets/ELM327DS.pdf

6. 謝辞

この研究を行うにあたり、顧問の米田隆恒先生には多大なご指導を賜り、実験走行にも幾度もご協力いただきました。また、立命館大学の野間春生先生、近畿大学の多田昌裕先生にも多くの助言をいただきました。この場を借りて、深く御礼申し上げます。