

市場の実操作を考慮した 車両動的性能評価プロセスの開発

○近藤 俊朗
田島 尚史
宮田 大毅
星原 光太郎

<例>

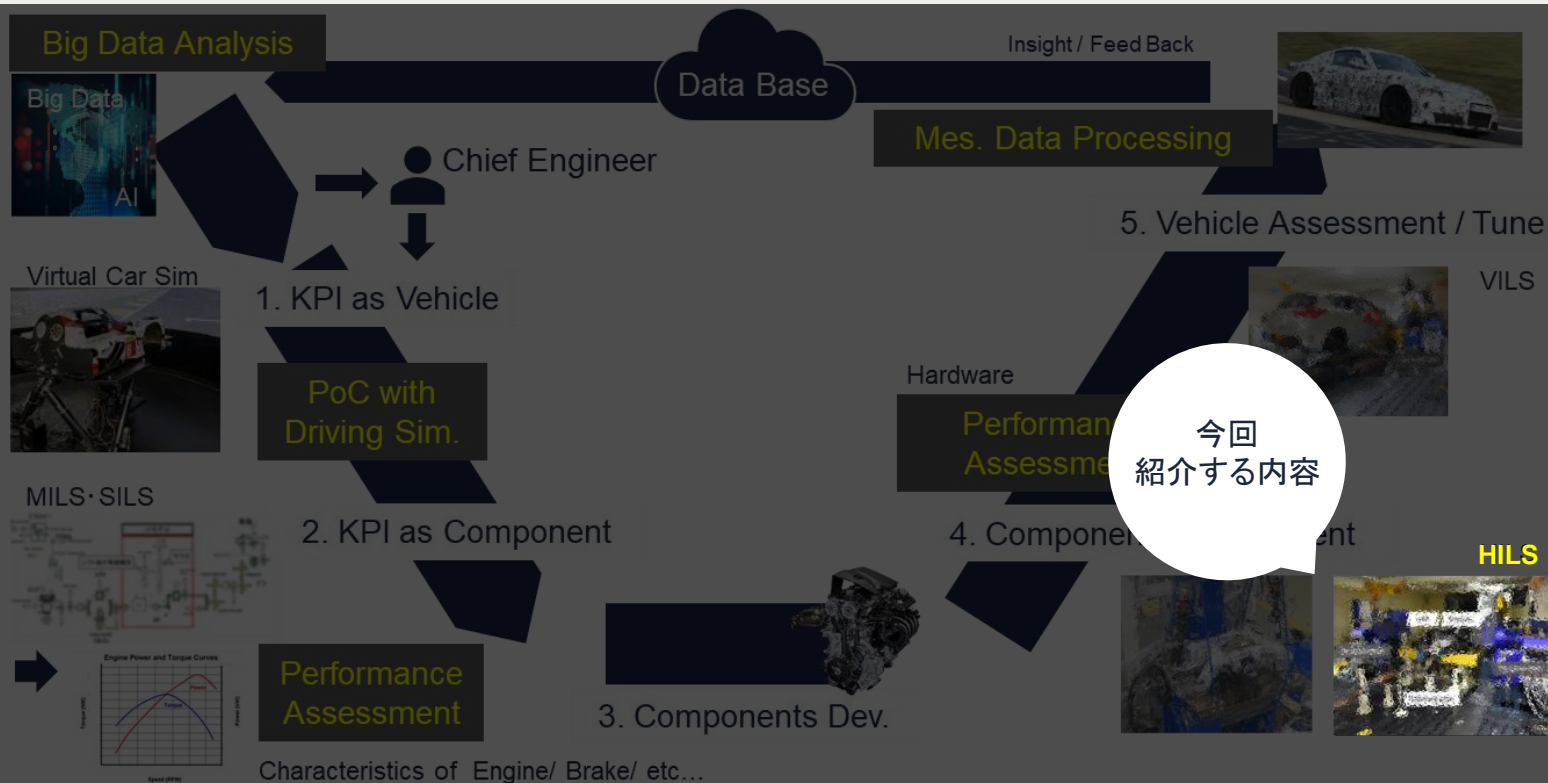
- 制御系の複雑化
→ ADAS、パワートレーンなど運動制御開発のリーン化が求められる。
- ソフトアップデート
→ 短期間で効率的な性能/品質開発が求められる。
- etc



背景:本日話す内容についての開発の立ち位置

開発のリーン化

BigData解析・XILS(X In the Loop Simulation)・AIを用いた開発の効率化が必要



今回、HILS×AIを用いた更なる効率化について紹介

今回話す概要

HILSのLevel up

重要な車両性能のひとつである**ドライバビリティ評価**に対する**課題抽出能力の向上**



Big Dataより市場の走り方をモデル化



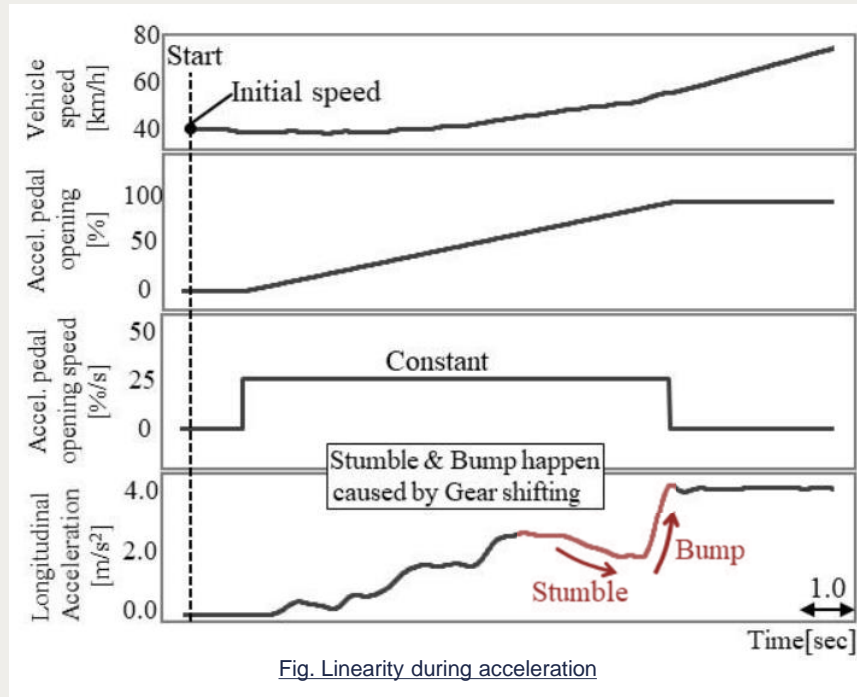
パワートレーン用のHILSに適用し課題抽出能力を向上

市場の走り方をモデル化し、ドライバビリティの課題抽出能力を向上 ⇒ リーンな開発のレベルアップ

ドライバビリティとは

車両の基本性能

運転者が意図した通りに車が動くかどうかを主観的な観点から評価した指標※



※ここでは前後方向に限定して説明しています

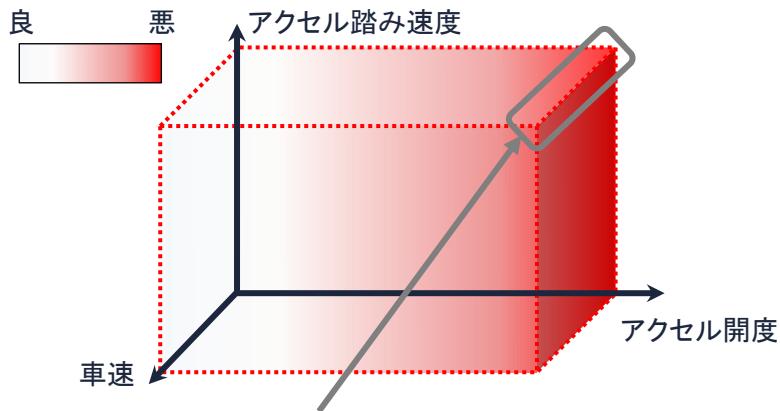
ドライバビリティは様々な場面でお客さんが感じ、車両にとって核となる性能

ドライバビリティ評価の課題

課題となる現象

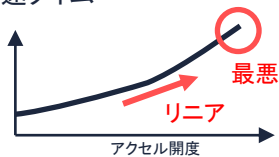
網羅的に課題抽出しようとする様々な条件が絡まりテストケースが膨大になる

最悪条件が明確な現象



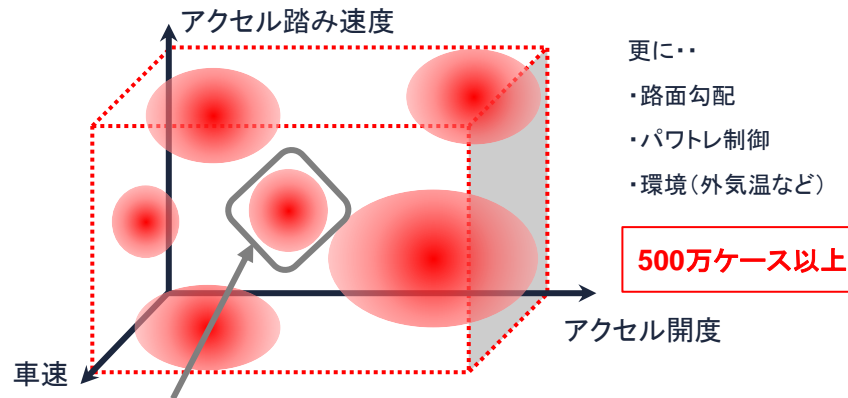
Eg. WOT加速評価

- ・0-100km/h加速タイム
- ・登坂性能
- ・こもり音
- ・最高速

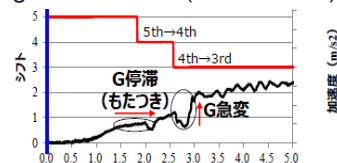


最悪条件のみで
品質確保が可能

局所的に課題が発生する現象



Eg. 変速ショック (ゆっくり踏み)

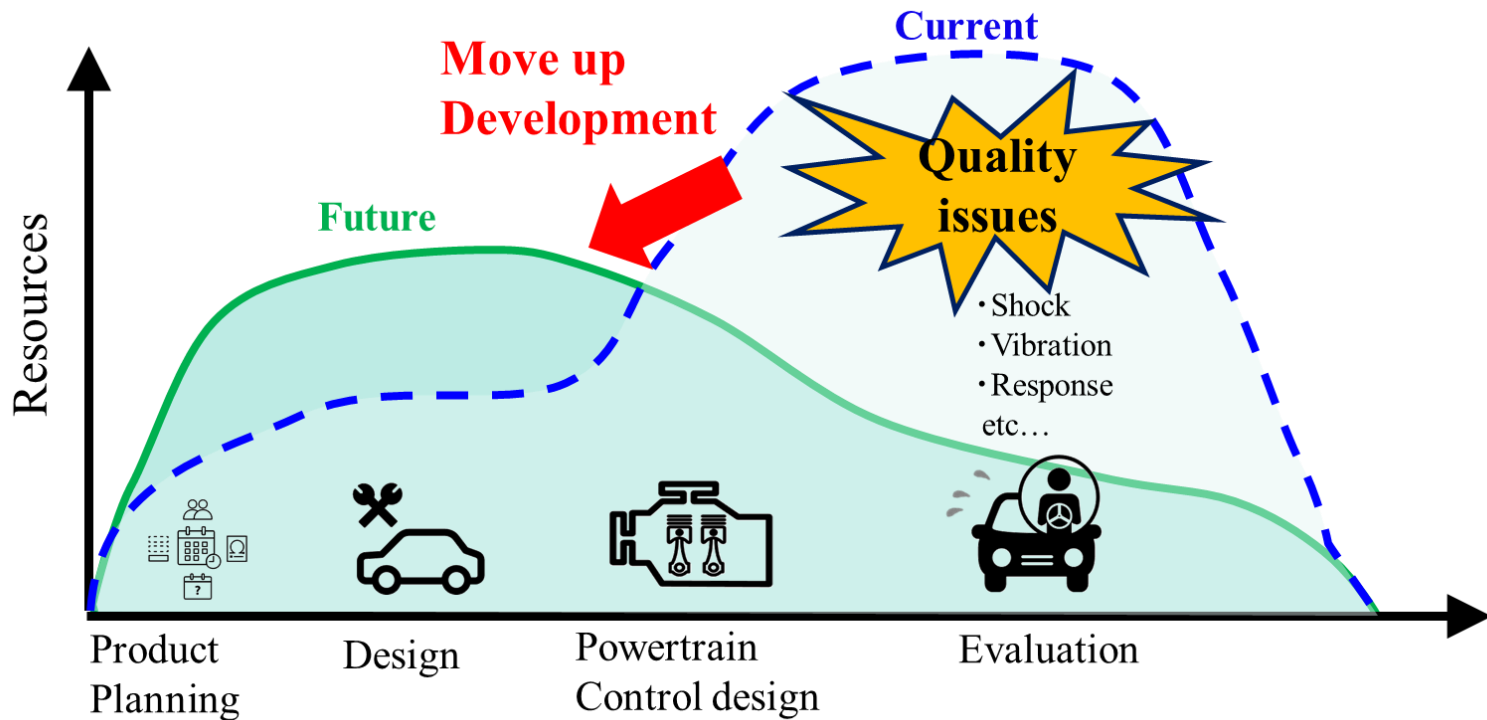


条件により最悪条件が点在
現状過去知見・経験で探索

課題が**点**在しており、網羅的な評価が困難

開発課題

試作車を用いた**熟練ドライバー評価**によって局所的な課題が発覚 ⇒ **開発のやり直しが多く発生**



開発の早い段階ですばやく課題を探索し、成立する制御定数を決めるシステム構築が必要

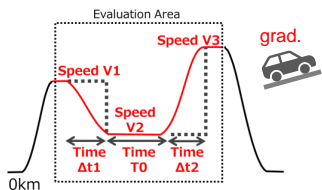
ドライバビリティの効率的な探索

HILS × AI

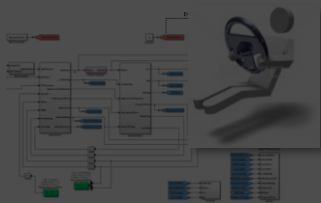
Drivability Automated Screening System (DASS)というシステムを提案 ⇒ 厳しい走行条件を効率的に抽出

 Driving

Vehicle Speed Profile



Driver Model



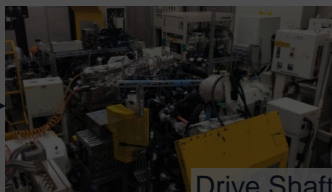
Accel. Pedal

Brake

Vehicle Speed

 Vehicle Behavior

Power Train Virtual & Real Simulator



Drive Shaft
Torque

Wheel Speed

Vehicle Model



Meas.
Data

 Rating

Drivability Self-Rating system



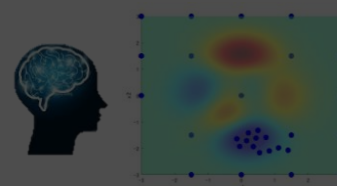
Drive away	(11)	7.5	<div style="width: 75%;"></div>
Acceleration	(297)	5.8	<div style="width: 58%;"></div>
Performance	(8)		
Tip in	(47)	5.8	<div style="width: 58%;"></div>
Tip out	(24)	7.3	<div style="width: 73%;"></div>
Deceleration	(8)	6.9	<div style="width: 69%;"></div>
Gear shift	(204)	6.4	<div style="width: 64%;"></div>

Rating all
Driving situation

Score

 Search/Optimize

Bayesian Active Learning (BAL)



Updated feasible region $\hat{\mathcal{P}}_k$

Feasibility Study

より評価の厳しい部分を評価(V, Δt, grad変更)

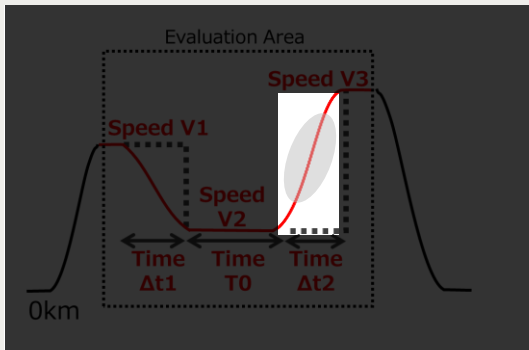
車速プロフィールを改善し、熟練ドライバーに頼らず評価できるシステムのレベルアップを図る

現状の車速プロフィールにおける課題

現状確認

従来の車速プロフィールはSigmoid関数を用いて車速間のつながりを滑らかに表現

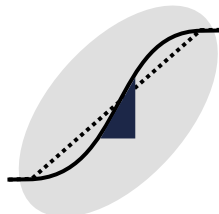
従来の車速プロフィール



Sigmoid

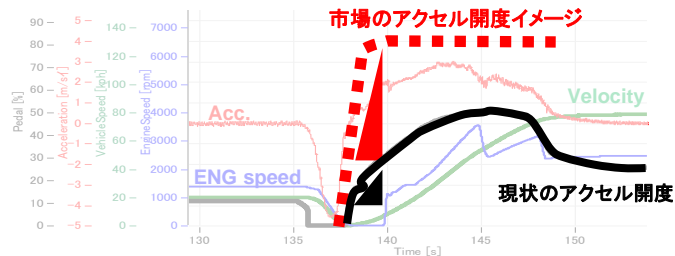
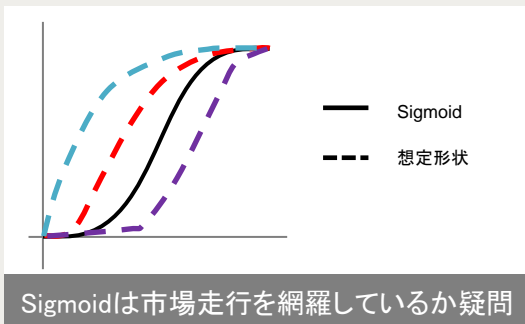
$$f_{sig} = \frac{c}{1 + e^{-b(x-0.5)}}$$

左右対称な形状を示す



加速時の応答をシグモイド関数で表現
⇒変数が少なく車速間のつながりが良い
⇒Sigmoidを採用し、パラメタ調整

発進・中間加速



発進時は踏み込みが強いはず

現状はゆるやかな踏み込み

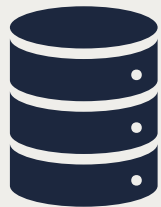
市場と乖離があるのでは？

今回、発進&中間加速においてSigmoidで正しく市場走行を表現できているか調査

調査内容と課題

やったこと

トヨタの保有する映像付き走行データ群から『発進・中間加速シーン』をクエリ抽出



Big Data

シーン

発進加速、中間加減速
(直進、先行車なし)



抽出



	車速	アクセル開度	舵角
UnixTS	CAN:SP1 Vehicle speed (SP1)	CAN:HV_ACCP Accelerator pedal angle	CAN:SSA Steering signal
1.46E+12			7.5
1.46E+12	1.35		7.5
1.46E+12	1.35	4.5	7.5
1.46E+12	1.35	4.5	7.5
1.46E+12	1.35	4.5	7.5
1.46E+12	57.43	0	3
1.46E+12	57.43	0	3
1.46E+12	57.43	0	3
1.46E+12	57.43	0	3
1.46E+12	57.43	0	3

5,000行以上のCSVデータ

× 20万ファイル以上

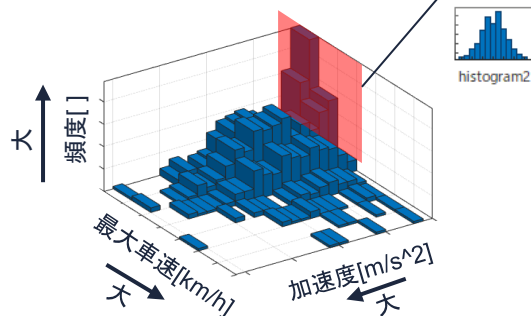
前処理等を行いたいデータを確認するには工数大

やったこと

Datastore等でデータを一括処理し可視化アプリを用いてヒストグラムを表示



Datastoreで一括処理



映像確認

加速度小部分のデータ例



- ・高頻度の最大車速、加速度を示す正規分布
- ・加速度小のデータが存在 (外れ値?)

先行車がいるため加減速を繰り返している
⇒ 加速度が小さい ⇒ 外れ値として除去

やったこと

データクレンジング後、市場の車速プロファイルを正規化しプロット ⇒ シグモイドで表現可能か確認

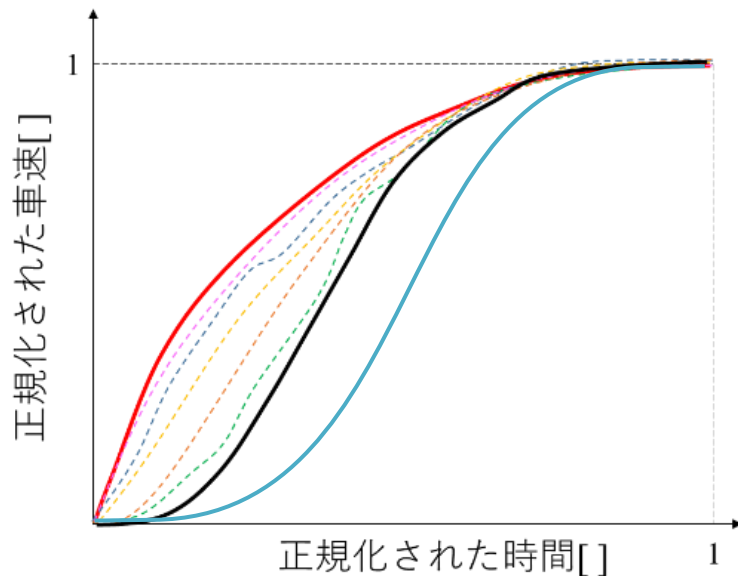


fig. 車速プロファイル一部とSigmoid線

- Sigmoid Curve
- Aggressive line (急激な加速パターン)
- Defensive line (緩やかな加速パターン)
- ⋯ 市場の車速プロファイル(1部抜粋)

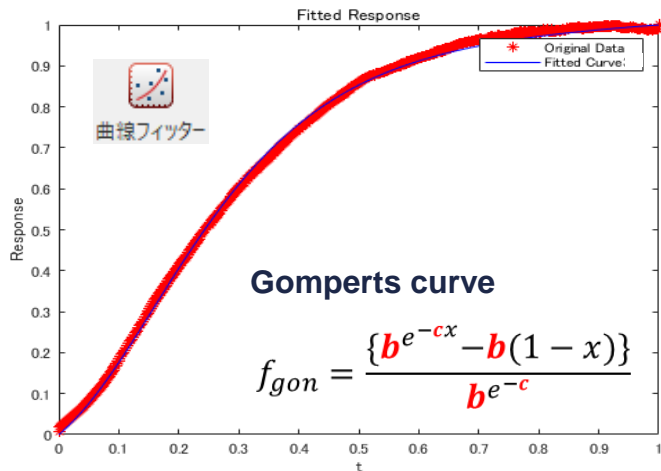
- ・市場の車速形状は、赤線～黒線までのバラツキ幅を持つ
- ・ばらつき3 σ 付近のAggressive/Defensive lineを表現したい
- ・Sigmoidでは**Aggressive lineを表現することが困難**

新たにAggressive/Defensive lineを表現できる関数が必要

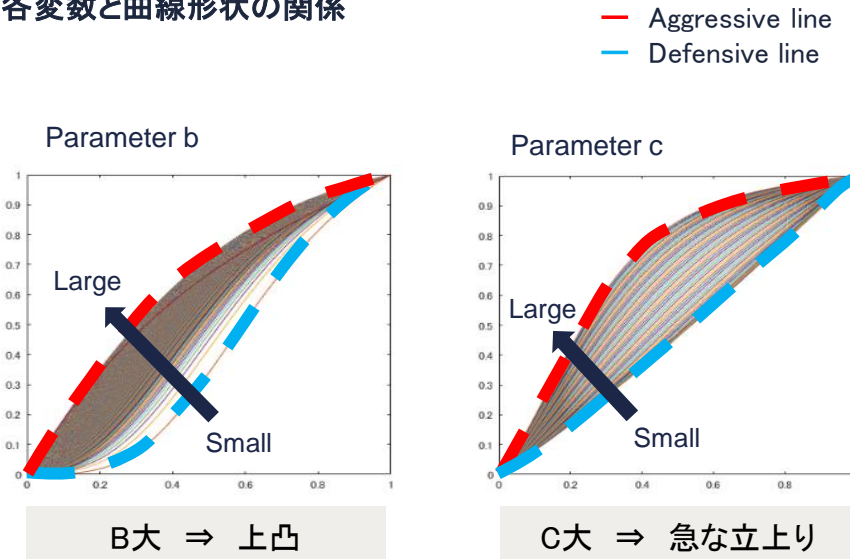
関数の要件

- ① Aggressive/Defensive lineの表現ができる
- ② 少ない変数で表現可能(効率的なAI探索に必要な要件)

Gompertz Curveの導入 ⇒ カーブフィット



各変数と曲線形状の関係



Gompertz Curveを導入しMATLABでカーブフィッティングを行うことができた

変数の範囲と各走行ラインにおける変数分布

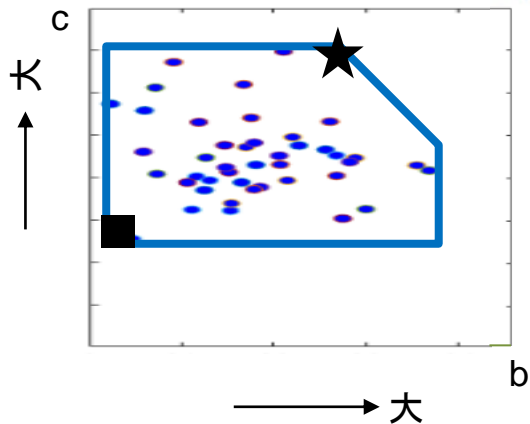
やったこと

発進と中間加減速におけるそれぞれの変数分布に分類し確認

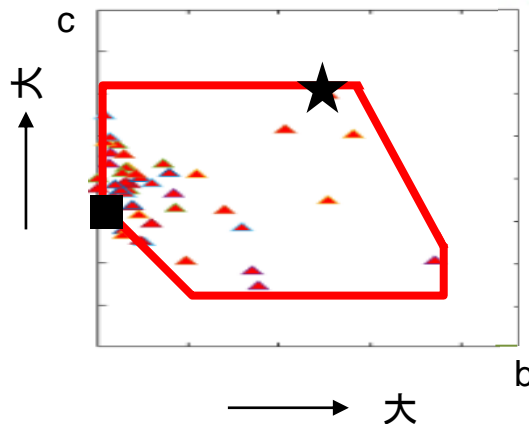
走行パターンごとのbとcの分布と各走行lineの関係

- ★ Aggressive peak (急加減速)
- Defensive peak (緩やかな加減速)

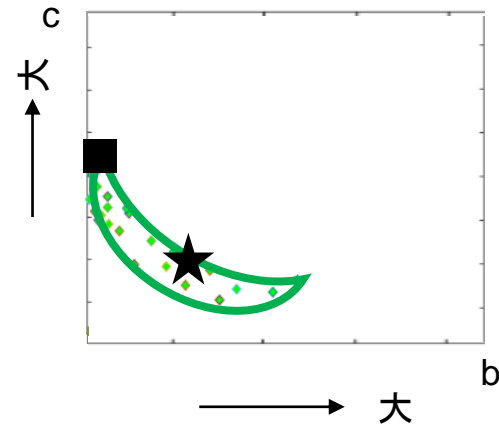
発進加速



中間加速



中間減速



カーブフィッティングからパラメタを抽出 ⇒ 市場及び各line上における b , c の分布を整理できた

やったこと

Defensive / Aggressive PeakでSigmoidとGompertz間に差があるか確認

■ SigmoidとGompertzで比較

Defensive LineではGompertzとSigmoidに差がない事を確認

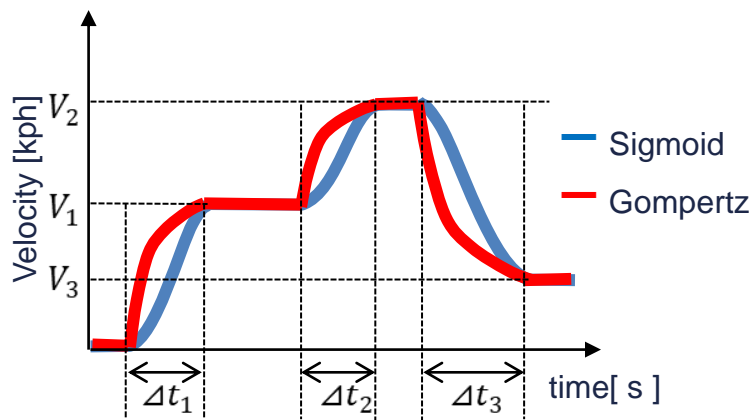


fig. 3つの走行パターンを網羅したテストケース例

Aggressive lineで車速プロファイルに大きな差

■ SigmoidとGompertzで比較

$V_{1,2,3}, \Delta t_{1,2,3}$ が等しいケースで50パターン走行し、比較

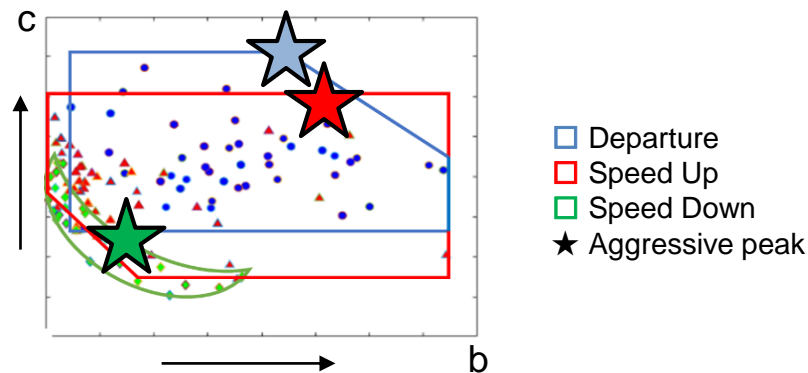


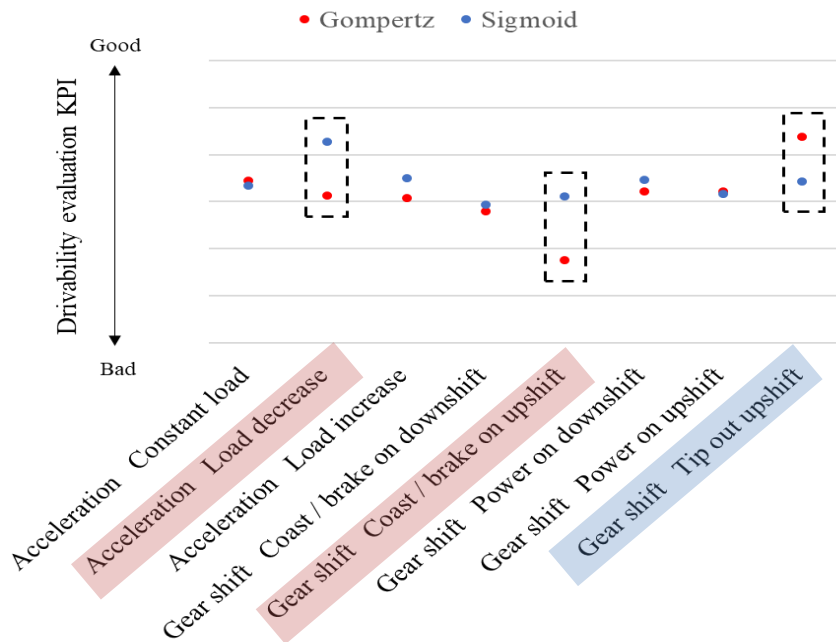
fig. Aggressive peakでの走行比較

Aggressive lineの各シーンで評価し、差を確認する

Sigmoidがうまく表現できないAggressive lineのpeakについて各走行シーンで比較

やったこと

KPIが官能で大きな差を感じるレベルである**1ポイント以上差のあるスコア**をピックアップ



Gompertzの方が評点悪化

- Acceleration Load decrease
- Gear shift coast/break on upshift

Sigmoidの方が評点悪化

- Gear shift Tip Out Upshift

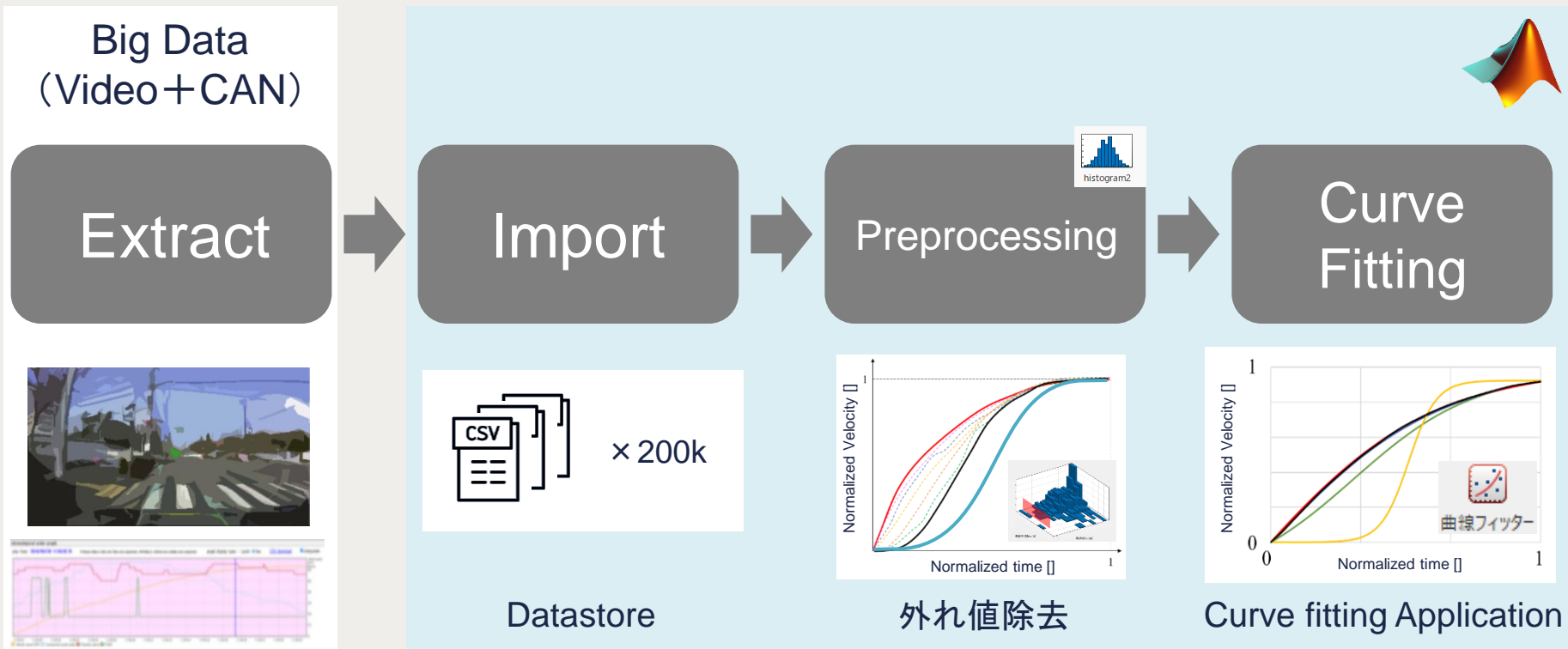
----- Gompertzの変数を調整することで表現可能

発進、中加減速シーンにおいて**従来の車速プロファイルでは抽出困難な課題を抽出**することができた

MATLABを使った開発フローまとめ

実施内容

市場走行データから特定走行の車速プロファイルを改善 ⇒ 従来は困難だった課題の抽出ができるようになった



プログラム初心者でもMATLABをうまく使うことで効率的に問題解決を行うことができた

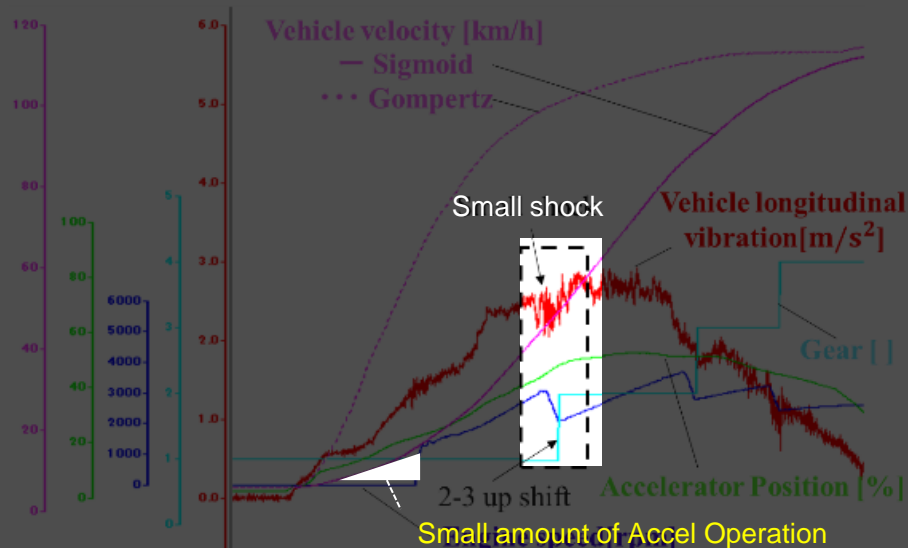
Appendix

時系列データで見る走り方の違い

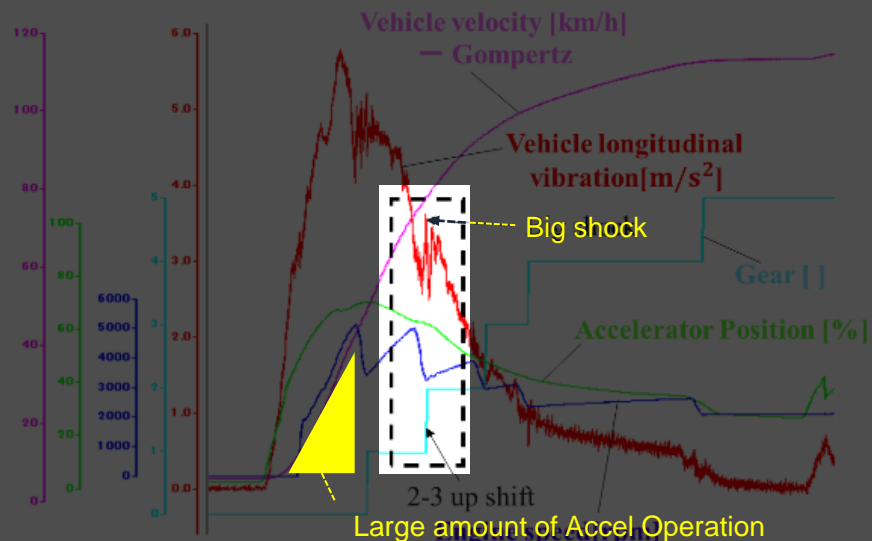
やったこと

Acceleration Load Decreaseについて発進時の違いを確認する

Sigmoid



Gompertz



Gompertzでは初期アクセル操作量大 ⇒ 2-3シフトアップ時のエンジン回転数が高い ⇒ 前後ショック大