

# 自動車の運転支援・自動運転システムと センシングと通信

九州大学

大学院 システム情報科学研究所

大学院統合新領域学府

オートモーティブサイエンス専攻

川邊武俊

# 目次

1背景：情報化に向けた制御からの取り組み

2運転支援システムと自動運転

3“最適経路”から省燃費運転へ

- 最適制御問題とモデル予測制御

- 車線変更

- 地形情報の利用

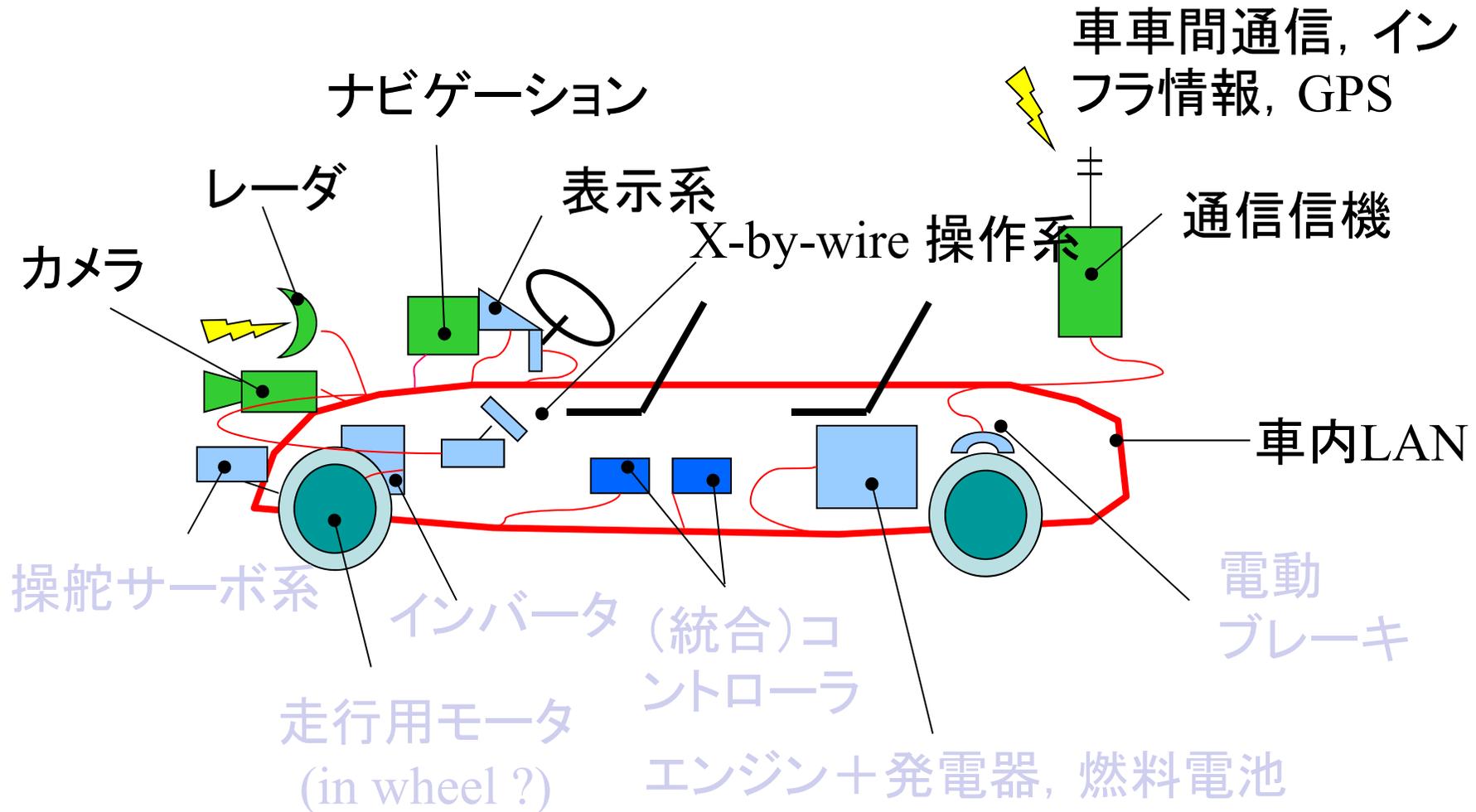
- 隊列走行への応用

- 合流のモデル化と制御

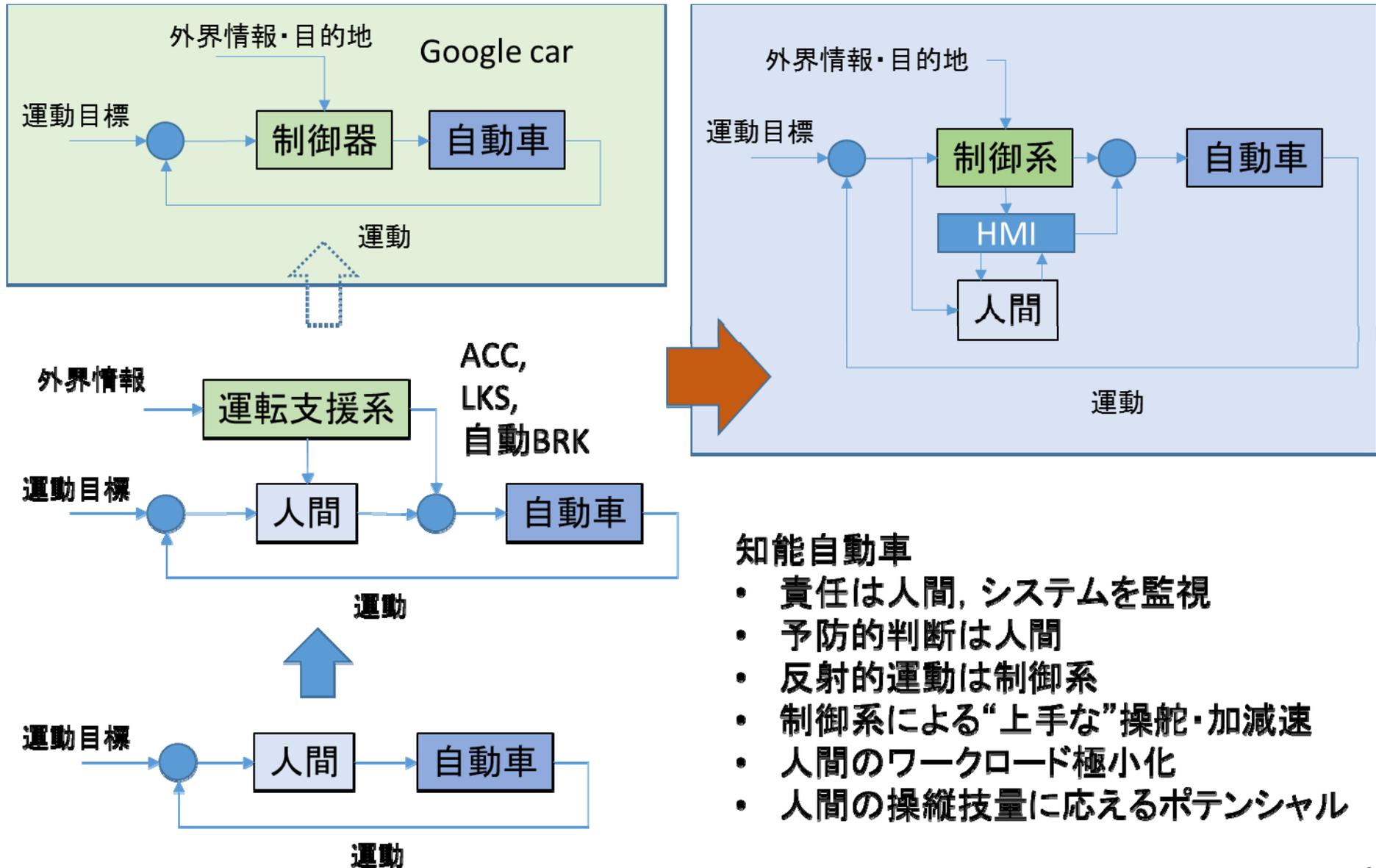
4 まとめ

# 1. 背景: 自動車の電動化・情報化

-次世代制御・運転支援システムのプラットフォーム-  
情報を用い“走り方”の改善による省燃費化を考える



# 運転支援システム・自動運転システム



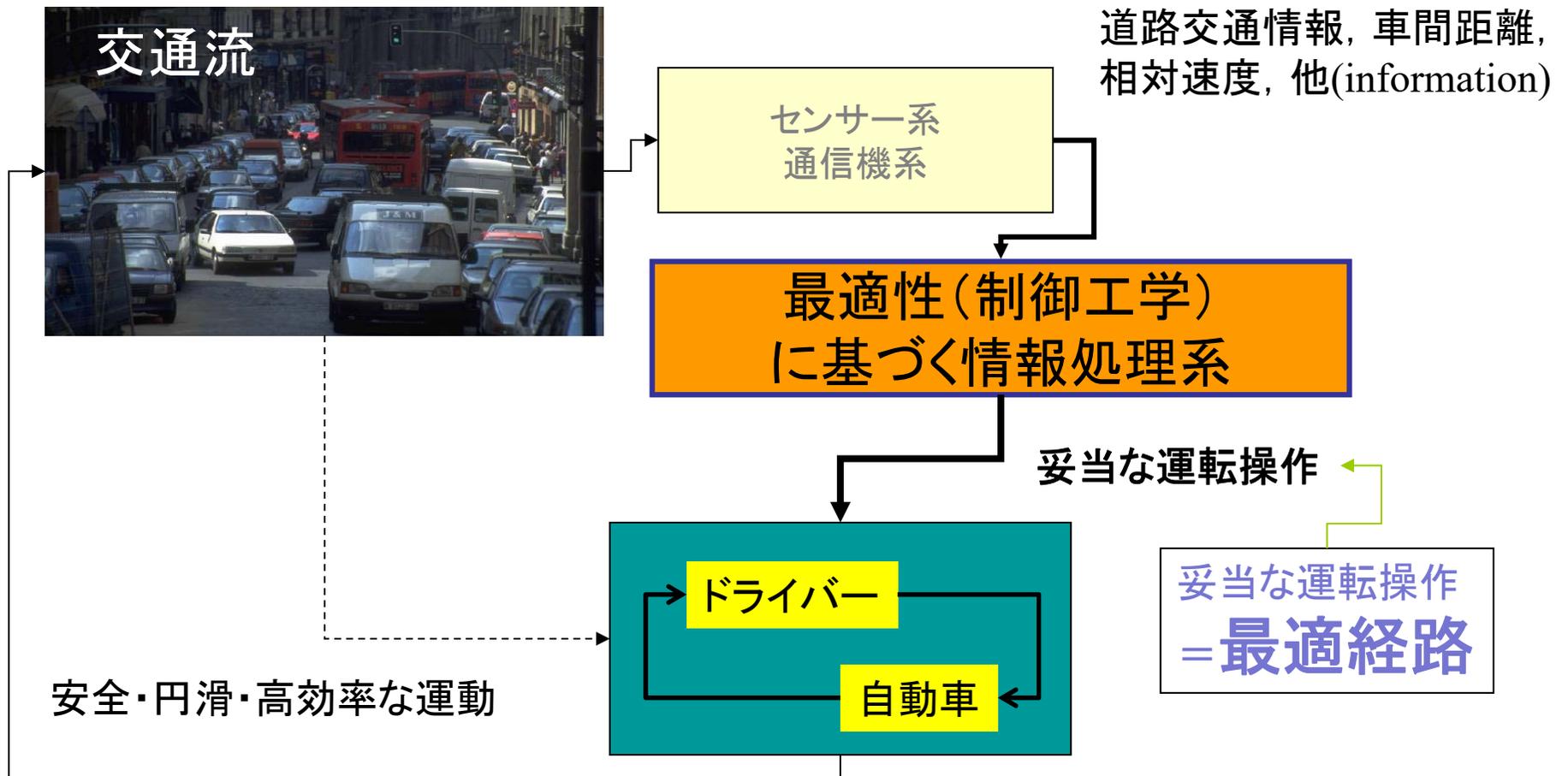
## 知能自動車

- 責任は人間, システムを監視
- 予防的判断は人間
- 反射的運動は制御系
- 制御系による“上手な”操舵・加減速
- 人間のワークロード極小化
- 人間の操縦技量に応えるポテンシャル

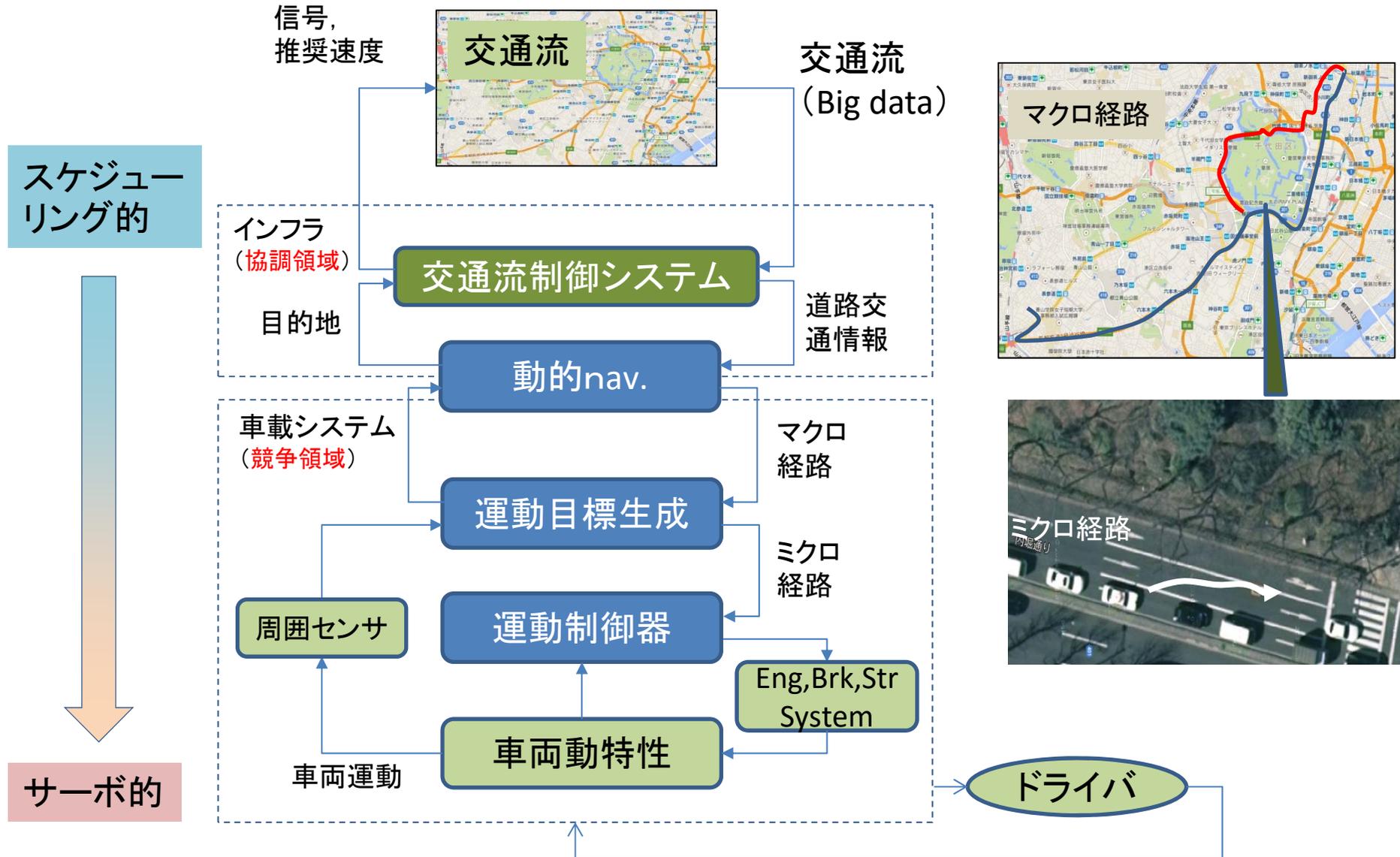
## 2. 運転支援システムと制御

支援の規範となる信号(妥当な運転操作)をリアルタイムで生成する.

- 交通流の考慮
- 最適性に基づく情報の圧縮と加工



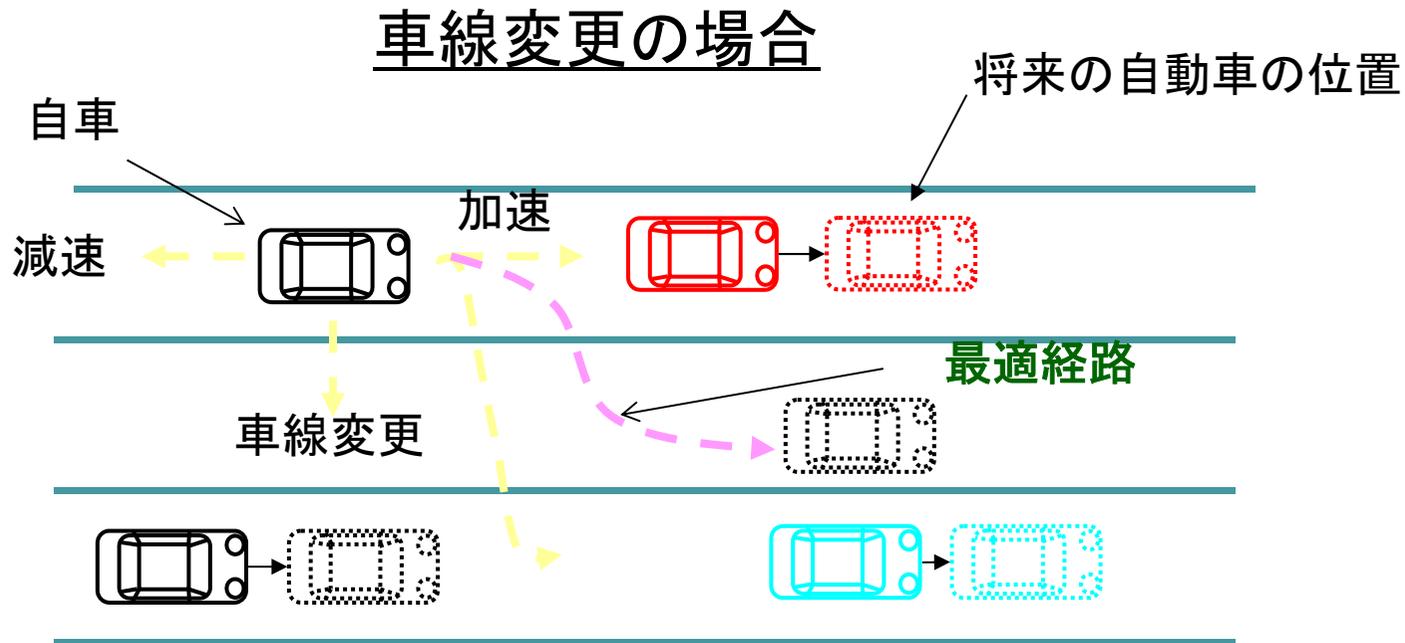
# 運動制御と情報化



自動運転システムがあるとするば

### 3. 最適経路

“最適経路: 最も安全かつ円滑な自転車の運動系路(⇔ハンドル操作, ブレーキ・アクセル操作のパターン)



周囲車両の運動情報から車載計算機で最適経路を実時間で生成する。

# 最適制御問題

評価関数

$$J[u(t)] = \int_{t_0}^{t_f} L(x(t), u(t), q(t), t) dt$$

を最小にするような操作量  $u(t)$ 、 $t_0 \leq t \leq t_f$  を求めよ。

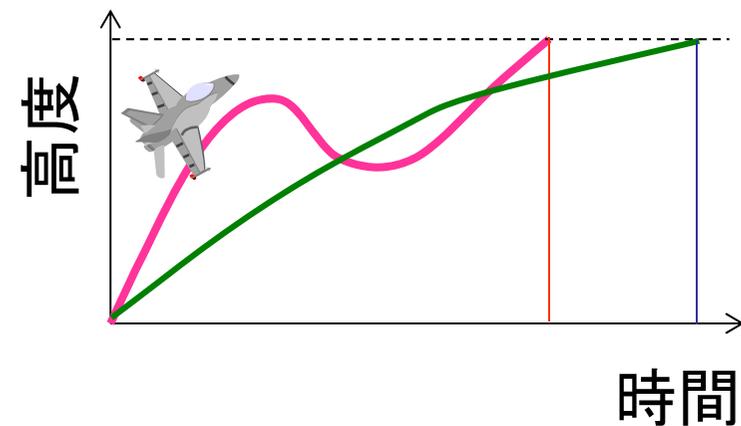
ただし、 $x(t)$ 、 $q(t)$  は以下の式に従うものとする。

状態方程式  $\dot{x}(t) = f(x(t), u(t), t), \quad x(t_0) = x_0$

拘束条件  $C(x(t), u(t), q(t)) = 0 \quad q(t_0) = q_0$

- 制御目標を表す評価関数と
- 制御対象の状態方程式と
- 制御対象の現在の状態量から
- 現時点からある時刻までの最適な操作量 (最適入力: 評価関数を最小にする操作量) を計算する。

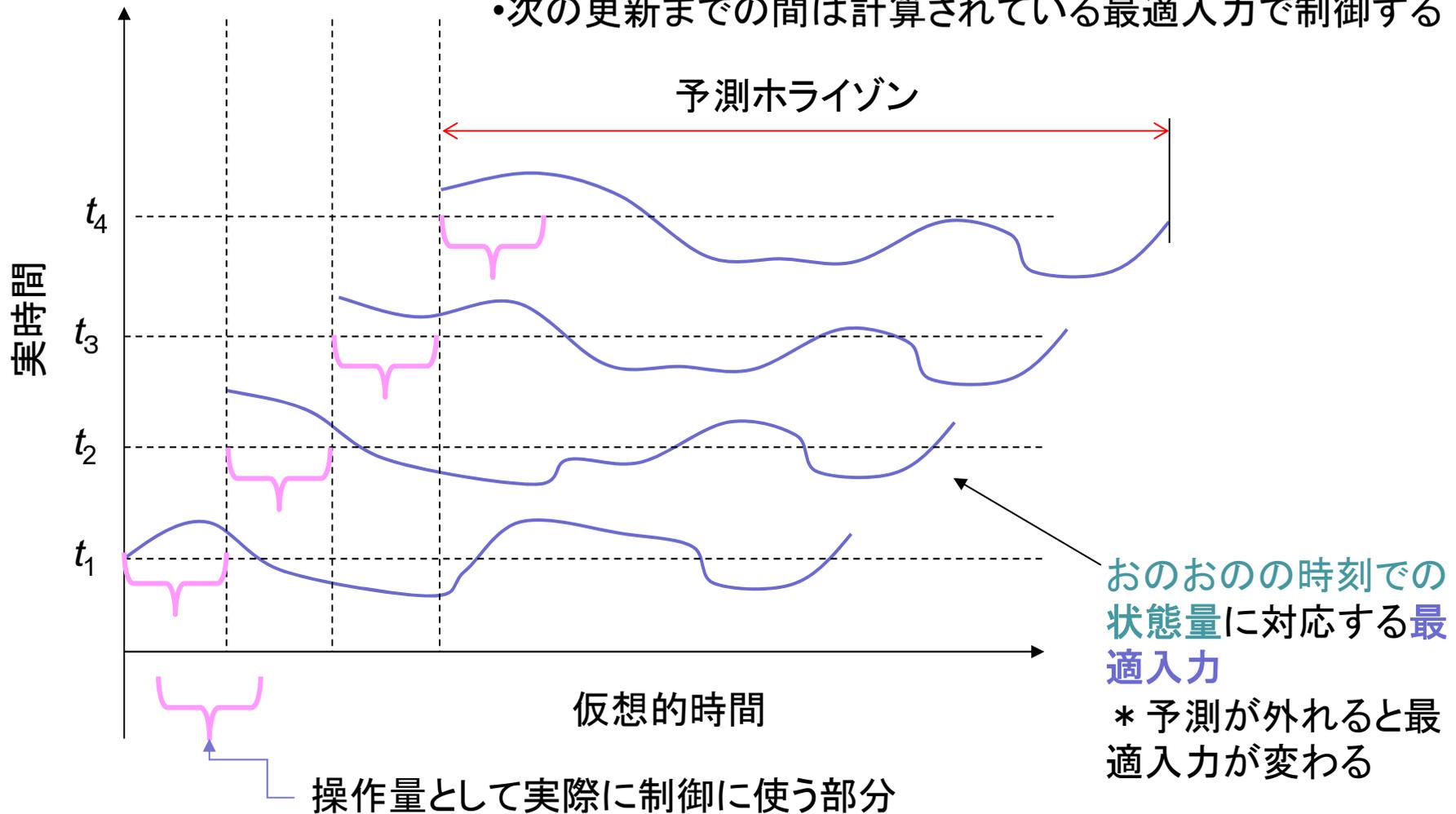
例: 最短時間上昇問題



# モデル予測制御(実時間最適制御)

## Receding Horizon 制御 (RHC)

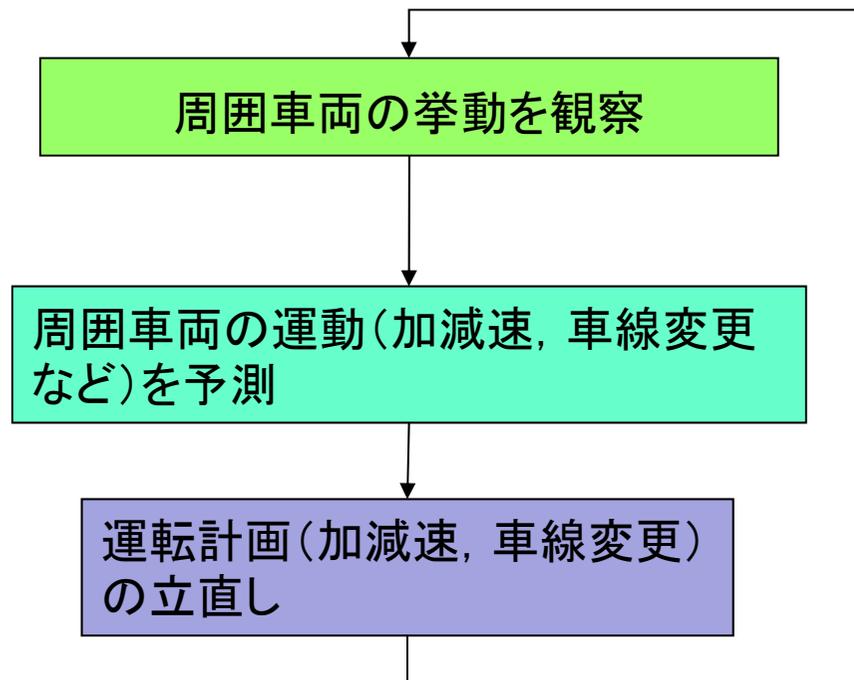
- 状態量 $x(t)$ の変化に応じて最適入力を定期的に更新する
- 次の更新までの間は計算されている最適入力で制御する



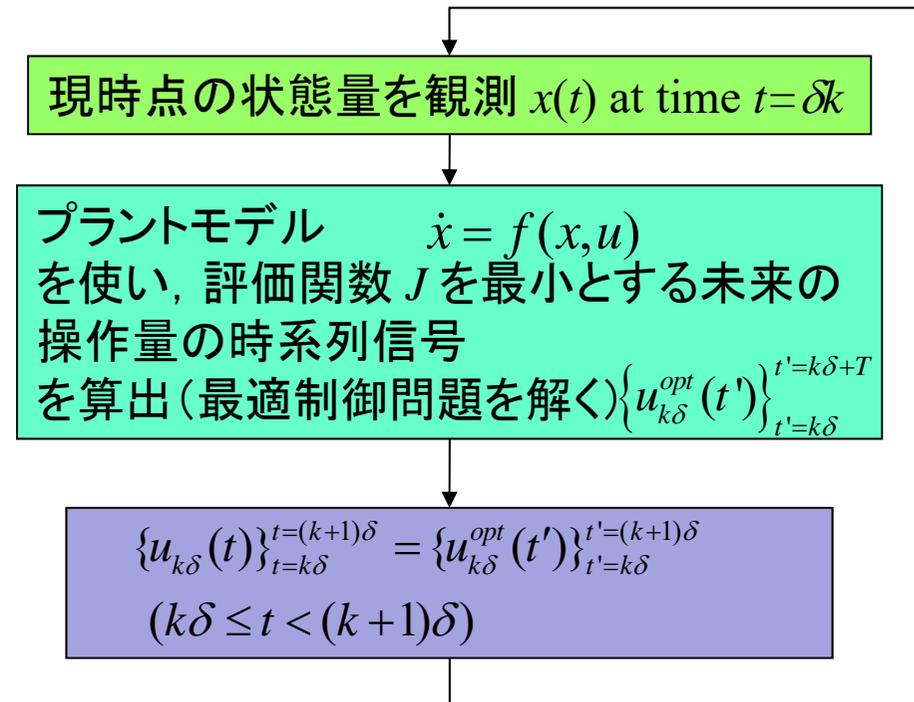
# 熟練ドライバーとモデル予測制御

仮説: ベテランドライバーの運転は妥当である

ベテランドライバーは周囲状況を理解し予測的に自動車を運転している



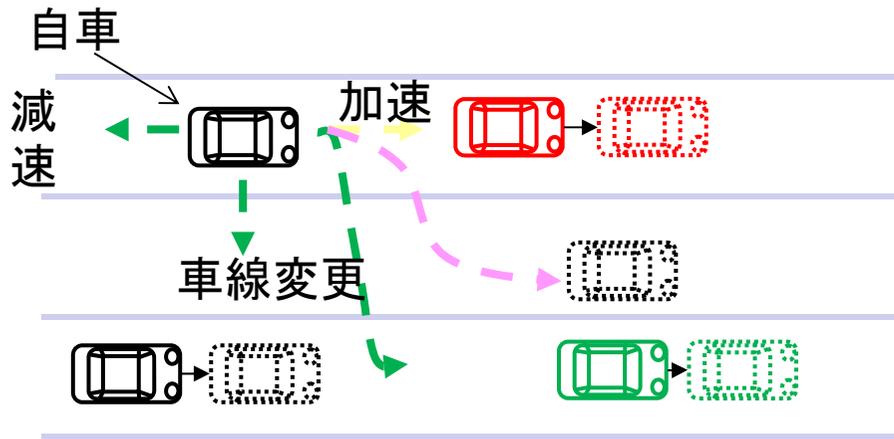
## ベテランドライバーの運転動作



## RHC algorithm

- ベテランドライバーの運転アルゴリズムはRHCアルゴリズムと類似である.
- RHCによりベテランドライバーを模擬するアルゴリズムを考える.

# RHCによる最適経路の生成



## 評価関数の設定:

- 車間距離が短くなりすぎない
- 他の車との相対速度が大きくなりすぎない
- 急ハンドル, 急ブレーキ, 急加速を控える

## 車群の状態方程式をたてる

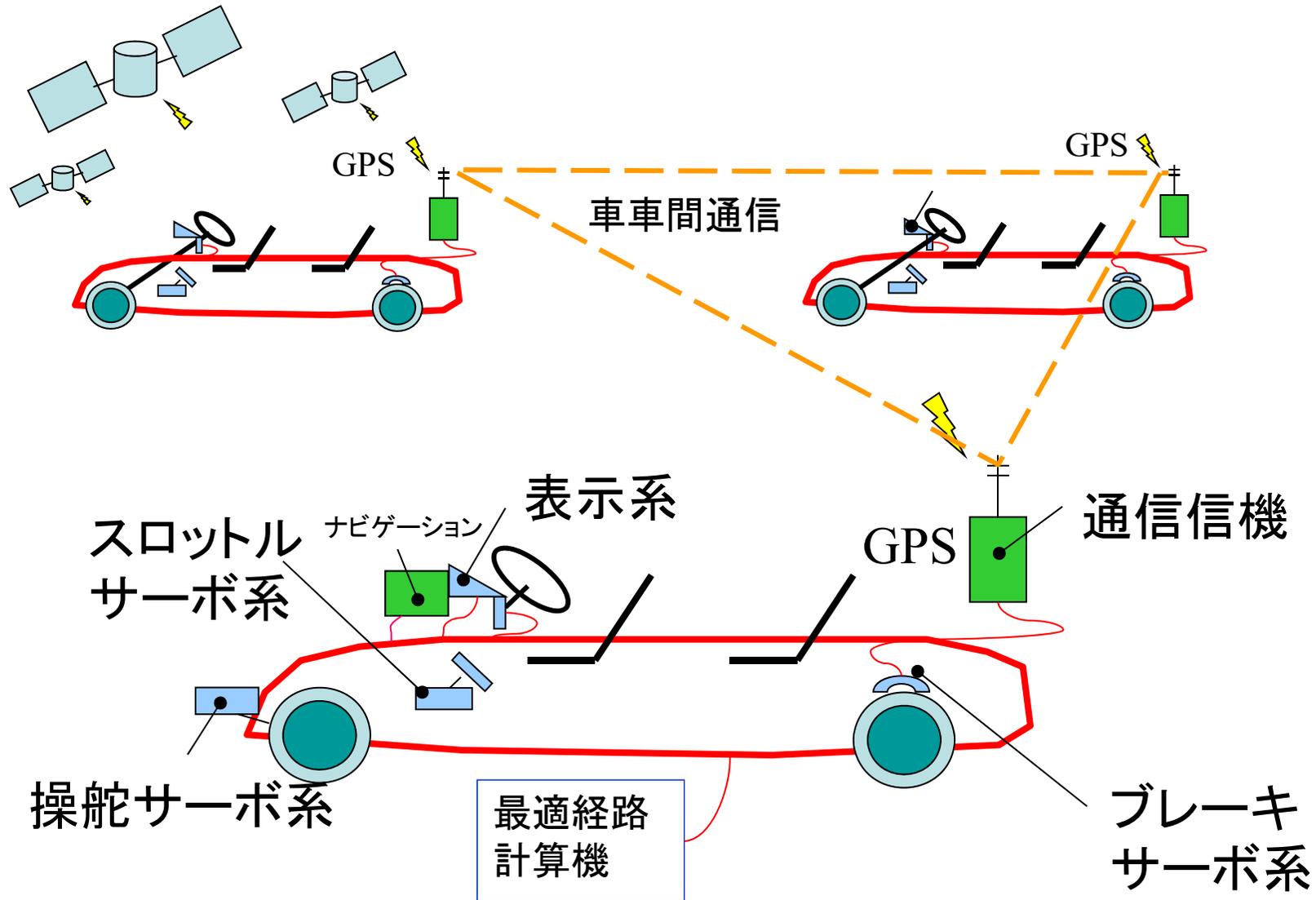
- 車はそれぞれお互いに影響を及ぼしながら運動している---一つのシステムとみなせる.
- そのシステムの入力は自車の運動(運転操作)とみなせる.

$$\dot{x}(t) = f(x(t), u(t), t)$$

$x(t)$ : 他の車の位置, 速度

$u(t)$ : 自分の車の位置, 速度(運転)

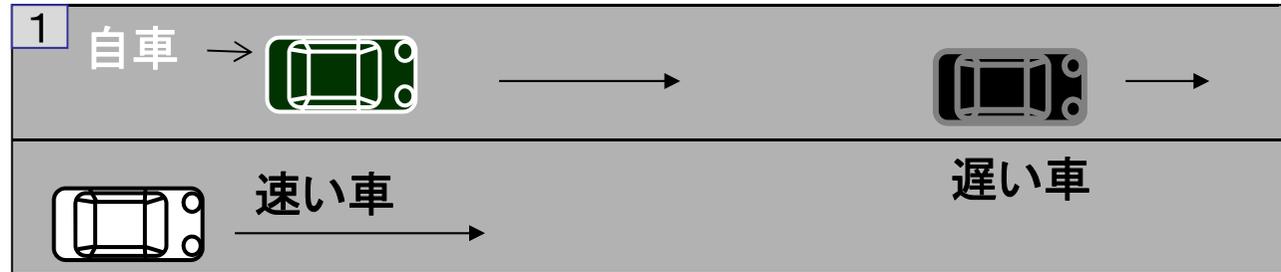
# 車線変更への応用(検証実験)



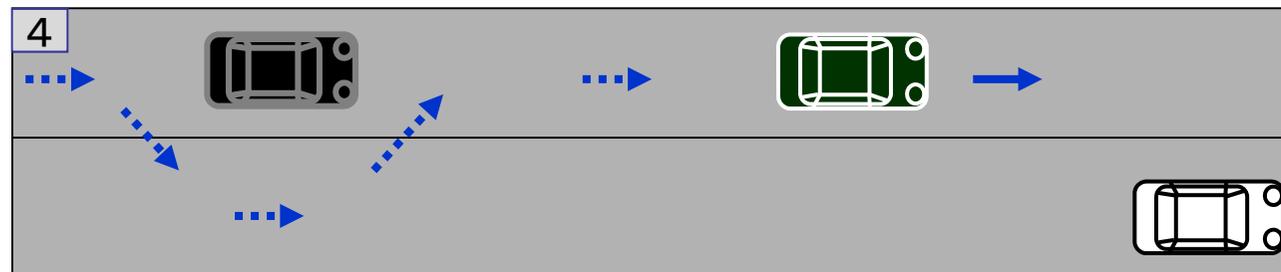
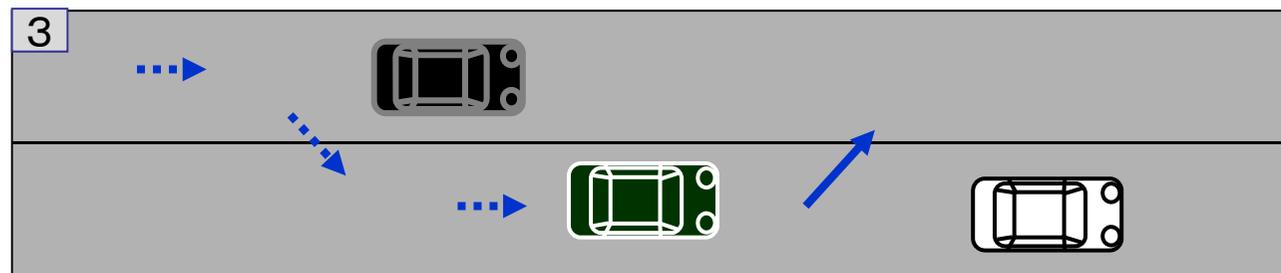
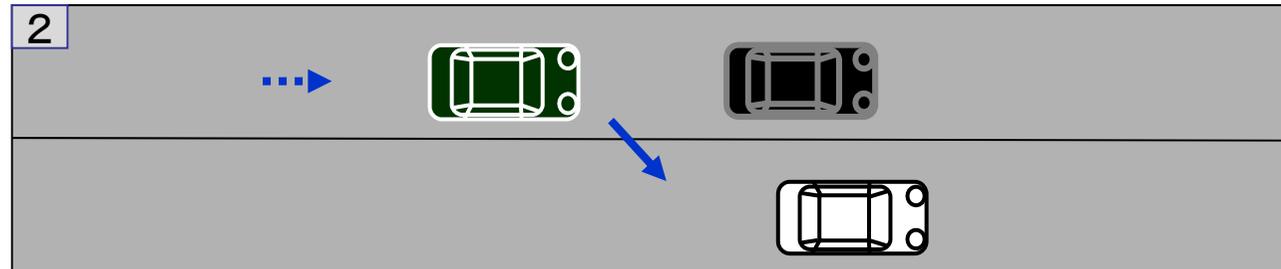
# 追越しシーンで生成された経路

自車は最適経路を生成トレースしながら自動運転する(車車間通信とGPSを利用)

初期状態



最適経路の生成と追従



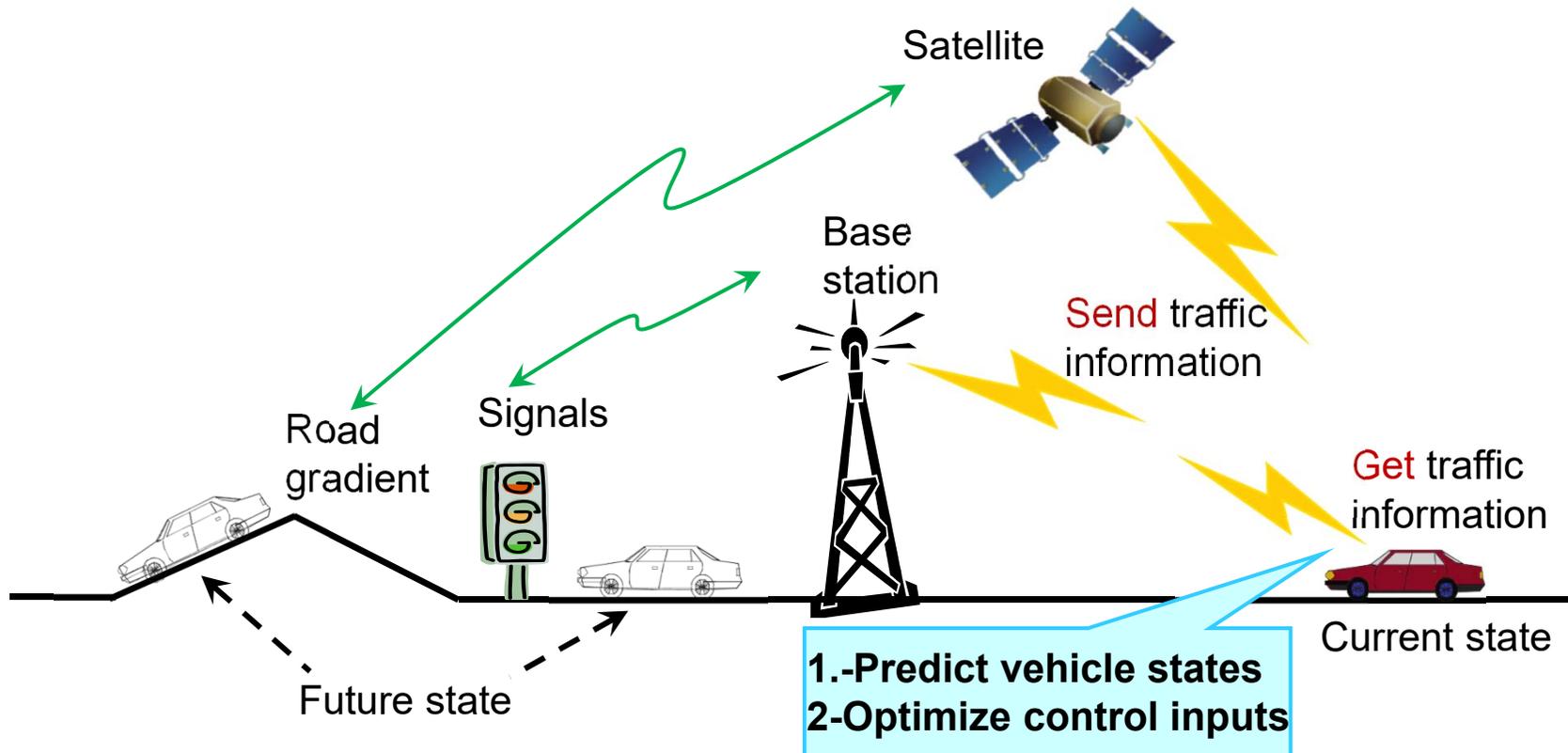
# 実験の様子

Camera2



(日産自動車提供)

## エコドライブのための速度パターン生成

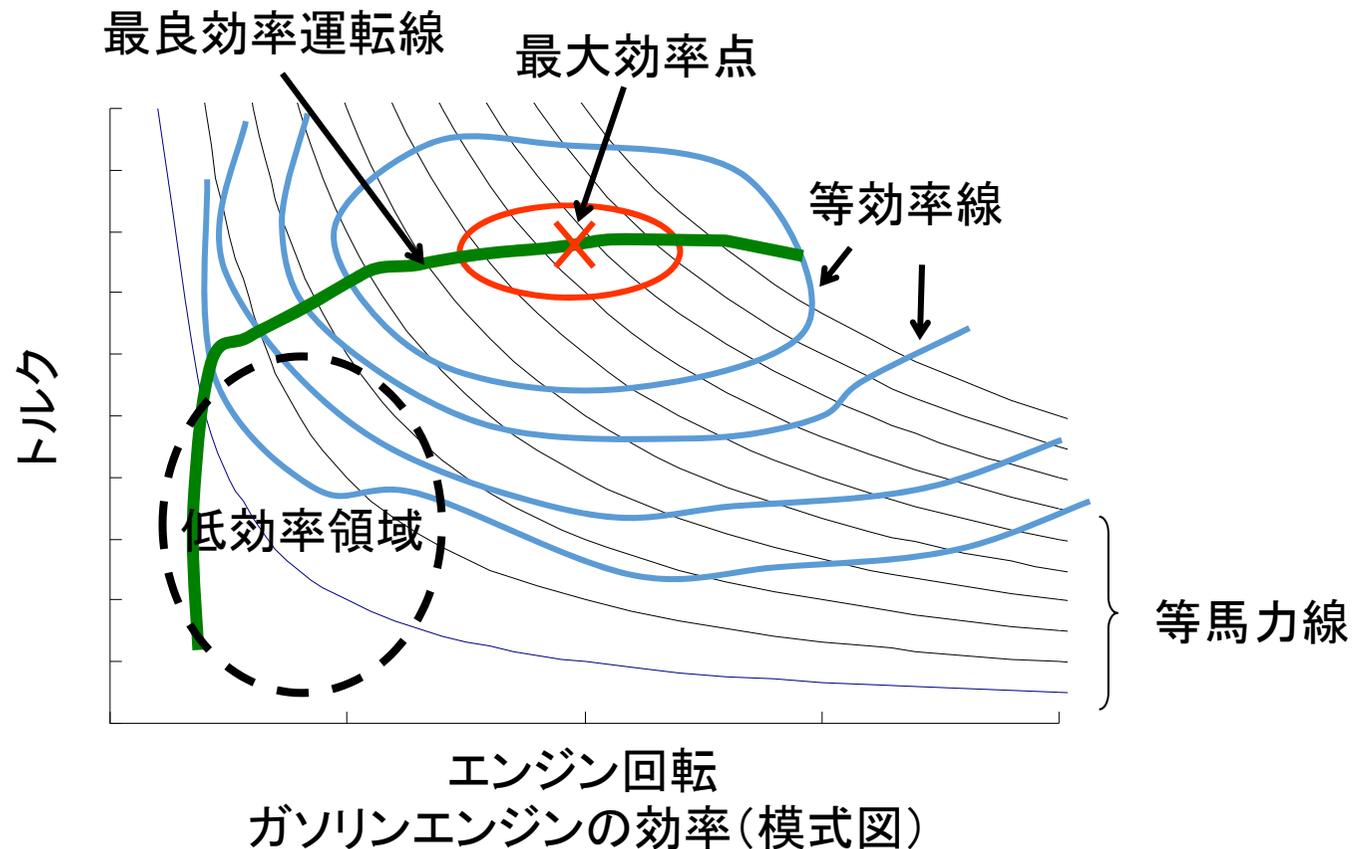


目的地までの燃費最適な速度パターンを実時間生成

- 目的地までの道路形状(勾配)データを利用
- 信号機現示情報を利用
- 先行車の挙動を予測
- 目的地までの混雑情報を利用

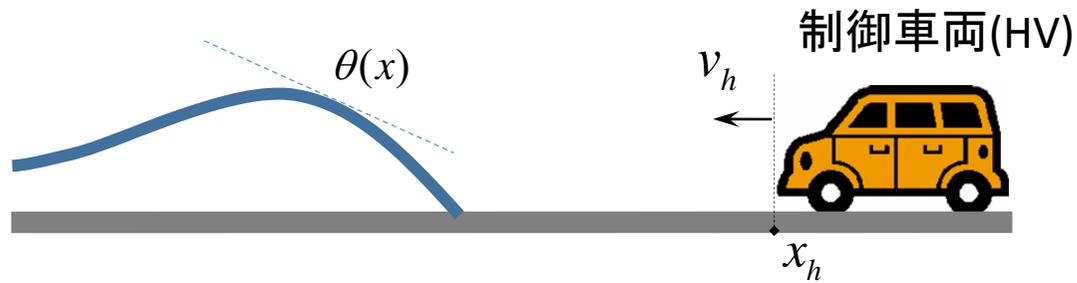
## 実時間最適制御で燃費が良くなるか？

- ・効率の良い動作点でエンジンを運転し、運動エネルギーを(車体質量で)蓄える。

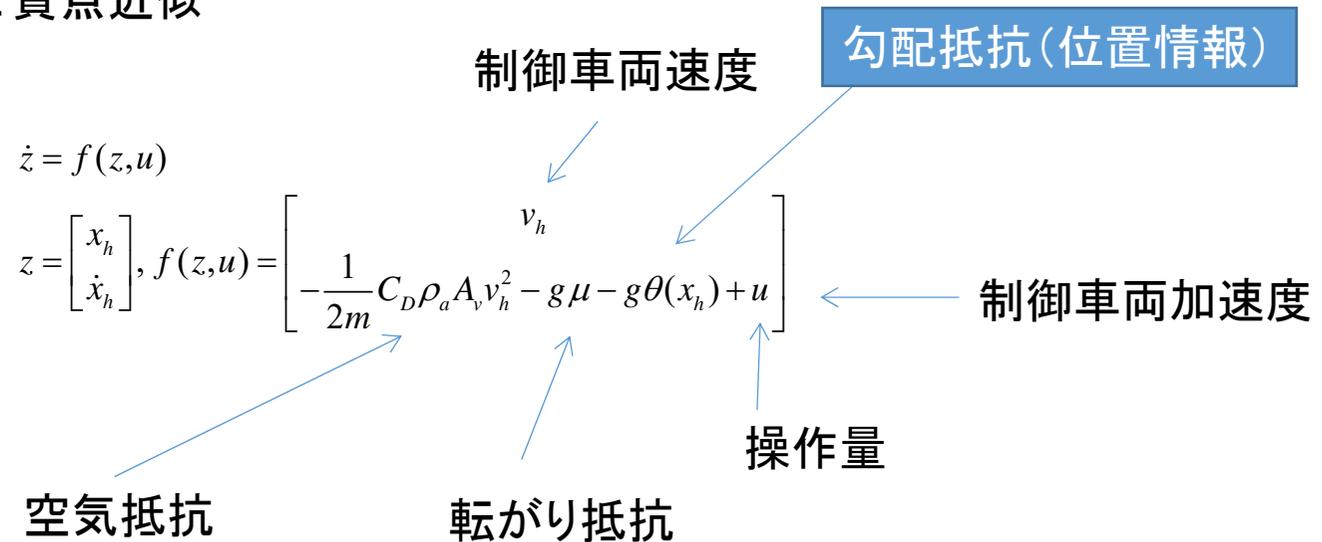


- ・ブレーキで運動エネルギーを散逸しない。←信号器の通過, 加速抵抗などのトレードオフ

# 道路勾配情報を利用した実時間最適制御

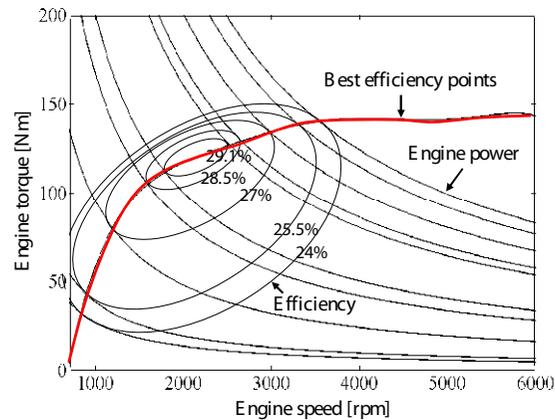


車両の状態方程式: 質点近似

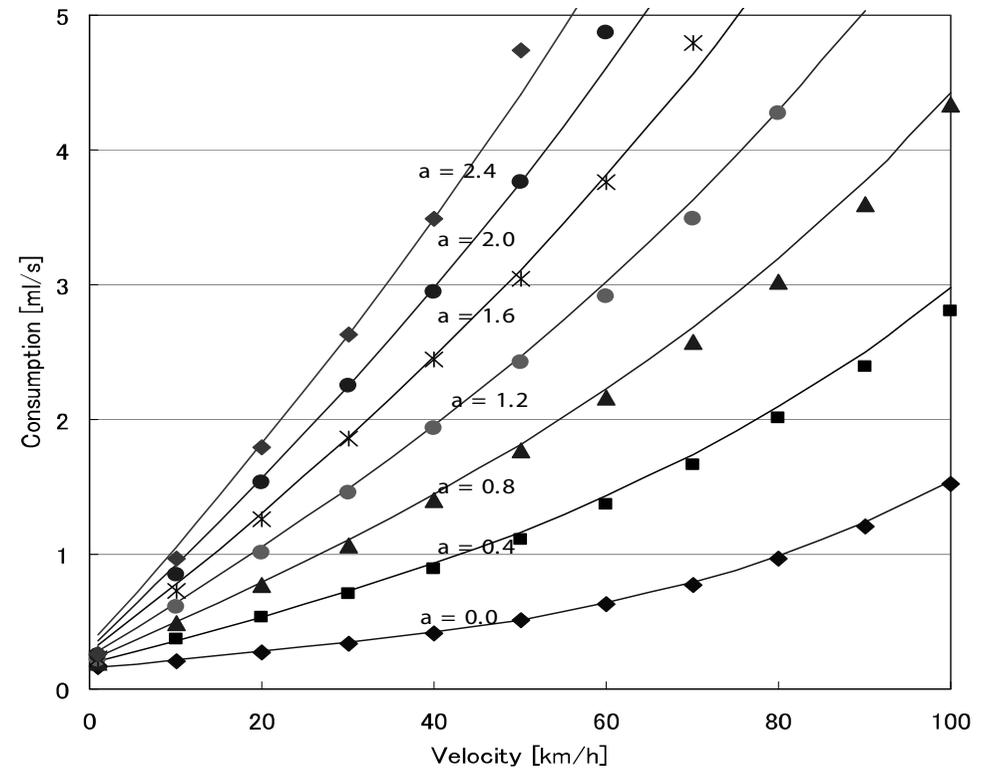


## 燃料消費の多項式近似

仮定: エンジンは最大効率線上で運転される(⇔理想的なCVTを使っている)



日産自動車 マーチの10-15モードカタログデータを参考に作成



## 瞬時燃料消費量

$$f_v(t) = f_{acl}(t) + f_{crs}(t)$$

$$f_{acl}(t) = \{a(t) + g \sin \theta(x_h)\} \{c_0 + c_1 v(t) + c_2 v^2(t)\}$$

$$f_{crs}(t) = b_0 + b_1 v(t) + b_2 v^2(t) + b_3 v^3(t)$$

$a(t)$ : 車両加速度

$v(t)$ : 車両速度

$c_i, b_i$ : 定数

$g$ : 重力加速度

## 最適制御問題

$$\text{Minimize } J \quad J[u(\tau)] = \int_t^{t+T} L(x(\tau), u(\tau), \tau) d\tau$$

subject to

$$\dot{x}(\tau) = f(x(\tau), u(\tau), \tau), \quad x(\tau_0) = x_0,$$

$$C(x(\tau), u(\tau)) = 0$$

瞬時燃費      道路に沿った加速度

$$L = w_1 \frac{f_V(\tau)}{v_h(\tau)} + \frac{1}{2} w_2 (a(\tau) + g \sin(\theta(x_h(\tau))))^2 + w_3 (v_h(\tau) - v_d)^2$$

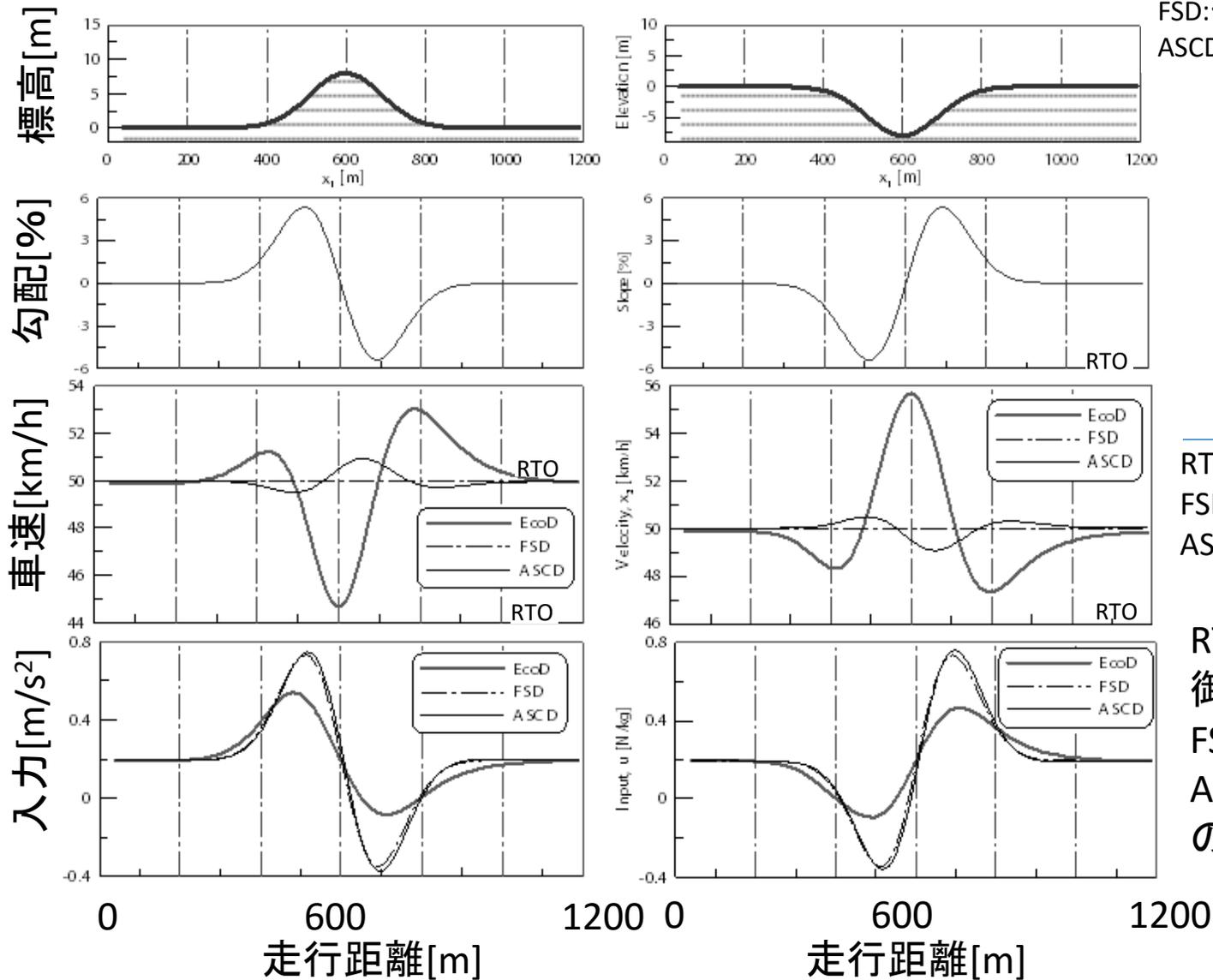
目標速度  
 $w_i$ : 重み,  $i=1,2,3$

$$C(u(\tau), u_d(\tau)) = \frac{1}{2} (u^2(\tau) + u_d^2(\tau) - u_{\max}^2)$$

許容する最大入力(加速度)  
ダミー入力

# 燃料消費の計算機シミュレーション(ホライゾン長10[s])

EcoD:実時間最適制御  
 FSD:一定速走行  
 ASCD:車速制御(目標値一定PI制御)

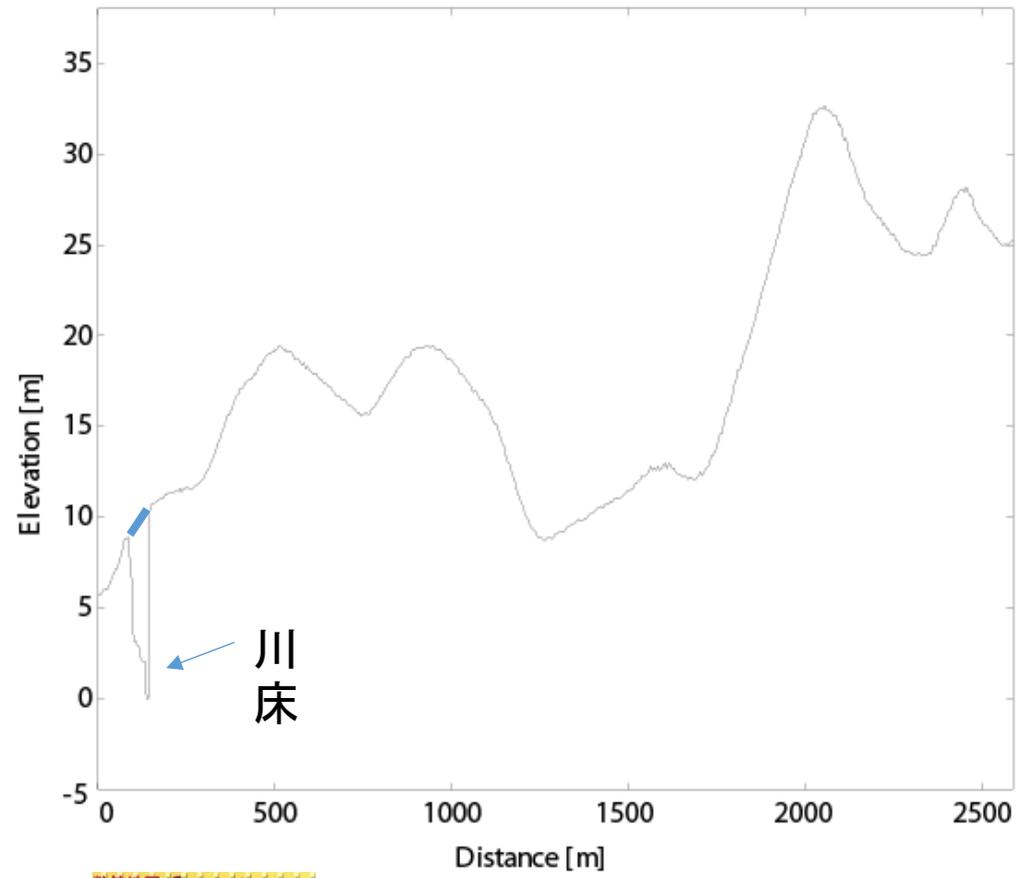


## 燃料消費 [ml]

	凸路	凹路
RTO	32.3(-9%)	32.4(-11%)
FSD	35.2(0%)	36.2(0%)
ASCD	35.6	35.3

RTO:実時間最適制御  
 FSD:一定速走行  
 ASCD:目標車速一定のPI制御

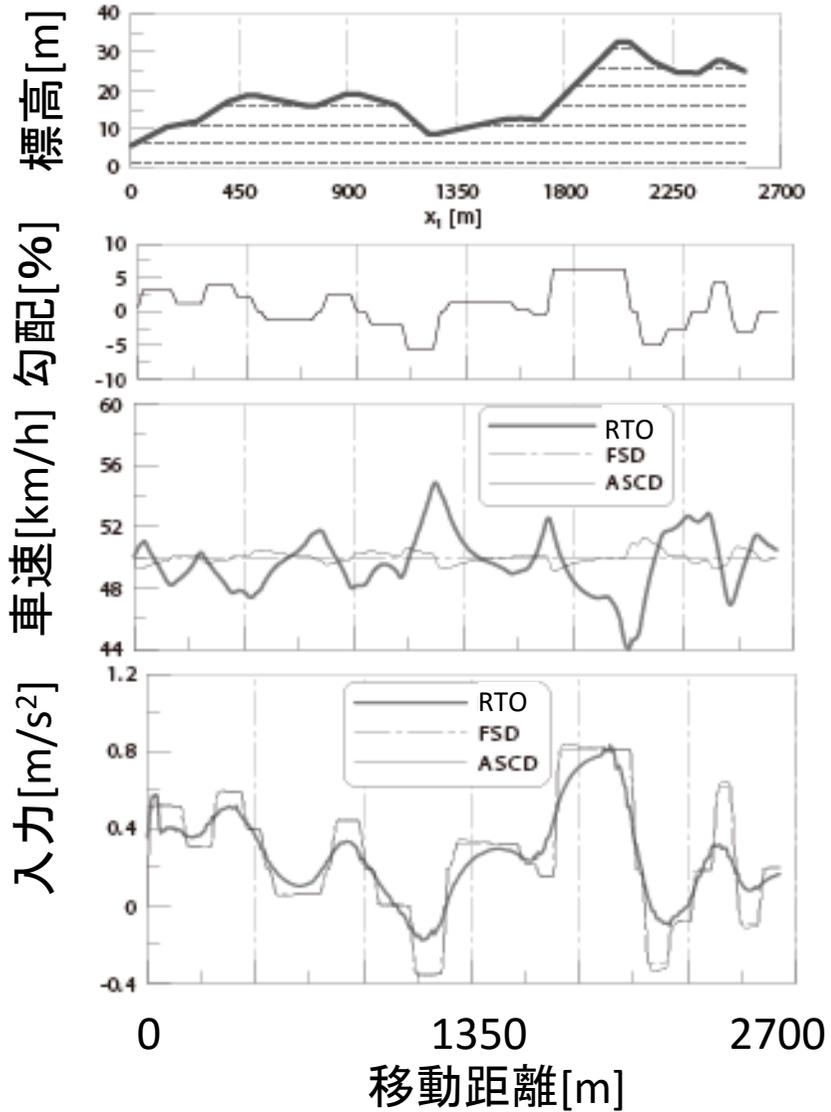
# 福岡市ユニバ通り勾配を用いた計算機シミュレーション



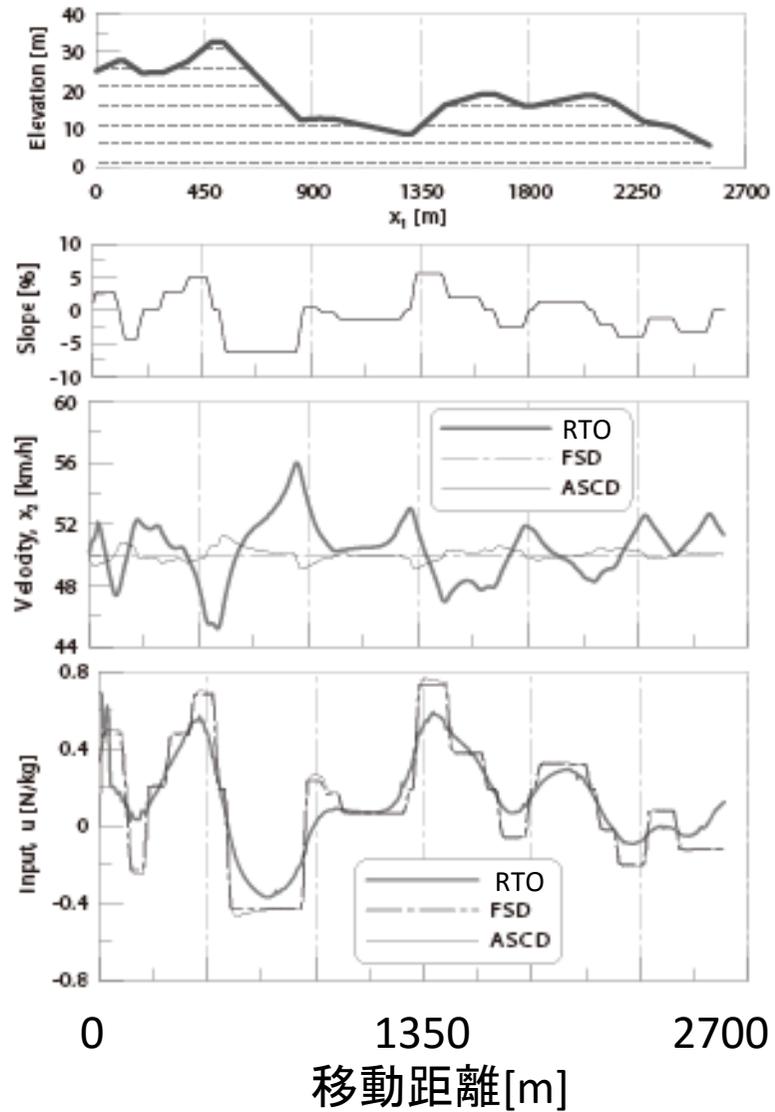
デジタルマップより得られた標高  
(数値地図 5mメッシュ 国土地理院)

# 計算機シミュレーション結果

北から南へ



南から北へ



## 燃料消費[ml]

	北から南(登り)	南から北(下り)
RTO	118.1 (-5%)	78.2(-6%)
FSD	123.5 (0%)	82.9(0%)
ASCD	124.3	84.1

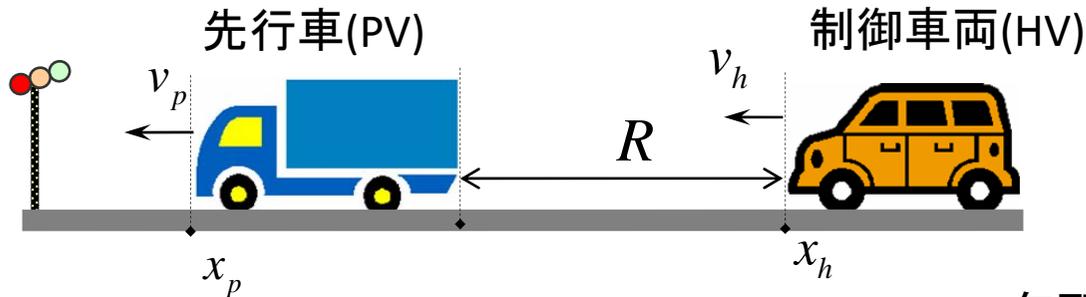
実時間最適化制御により:

- ・フィードバック制御より速度変動を大きく, 入力の振幅をより小さくチューニングすると,
- ・上り坂(下り坂)の手前から加速(減速)する
- ・坂の頂点(底)付近で入力が小さい

フィードバック制御  
では作れない特性

⇒ASCDより燃費よくチューニングできる

都市交通の中での性能評価, 先行車停止挙動のモデル化



車両の状態方程式: 質点近似

$$\dot{x} = f(x, u, q)$$

$$x = \begin{bmatrix} x_h \\ \dot{x}_h \\ x_p \\ \dot{x}_p \end{bmatrix}, f(x, u, q) = \begin{bmatrix} v_h \\ -\frac{1}{2m} C_D \rho_a A_v v_h^2 - g\mu - g\theta(x_h) + u_h \\ v_p \\ q \end{bmatrix}$$

勾配抵抗 (位置情報)

制御車両速度

制御車両加速度

操作量

先行車速度

先行車加速度

- 先行車の速度から疑似微分
- 赤信号では先行車が停止

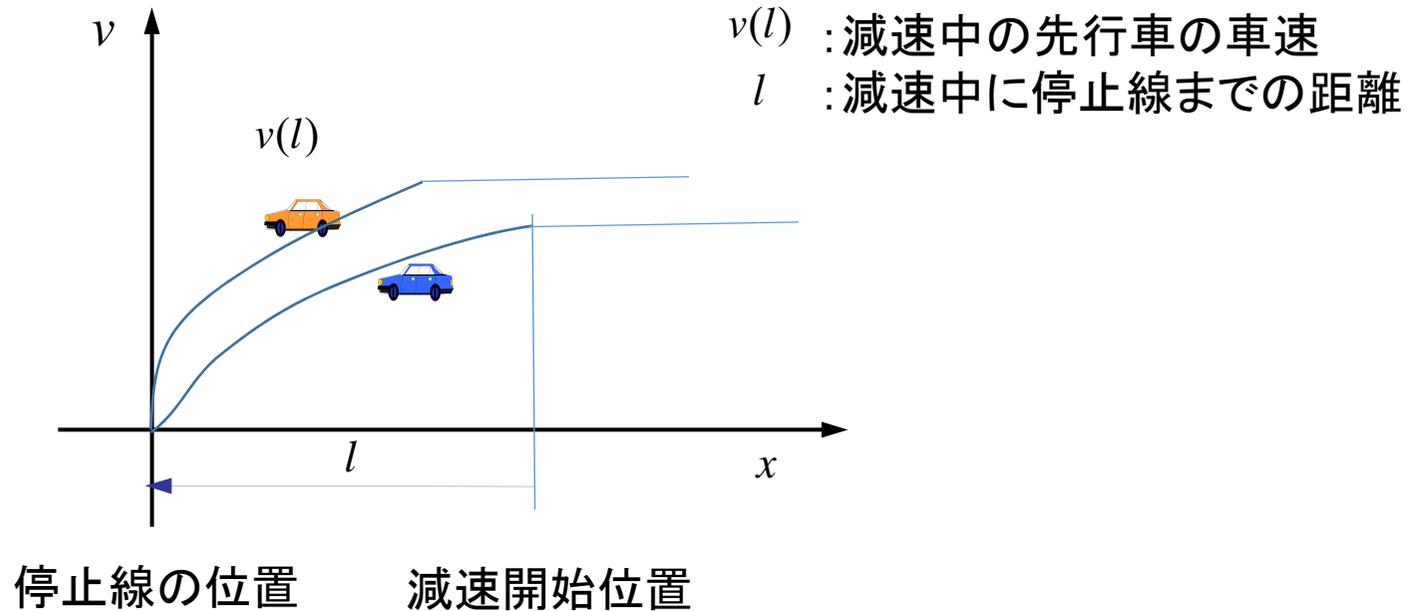
ここで

$$q = \begin{cases} a_p, & \text{先行車(PV)が減速以外の時} \\ a_b, & \text{先行車の減速} \end{cases}$$

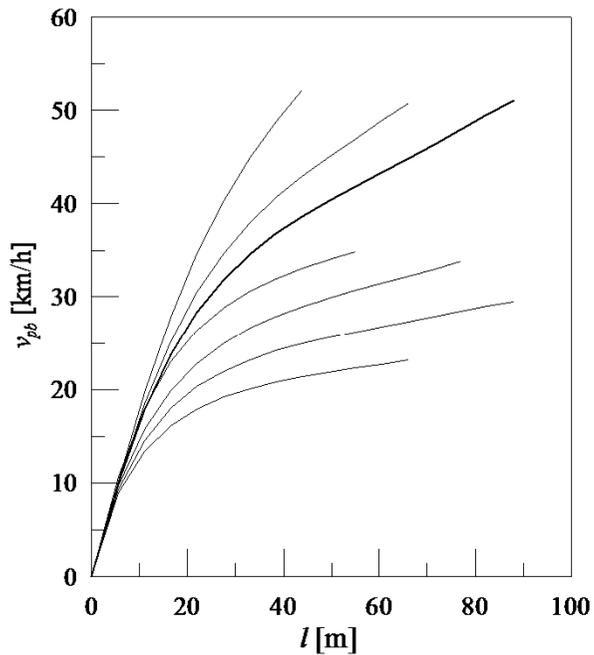
モデルを使用

# 先行車挙動のモデル化(信号停止)

赤信号での停止挙動のモデル化  $v(l)$  を  $v$  の関数として表現する.



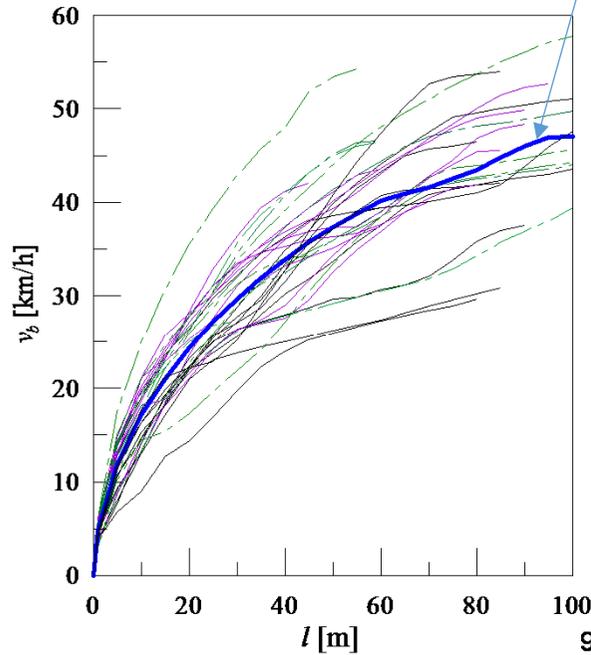
### 停止挙動モデル



$$a_b(l) = \bar{a}_b(l) \left( \frac{v_b(l)}{\bar{v}_b(l)} \right)^2$$

のバリデーション

### 実験データ



国道129号線(神奈川)  
における6人のドライ  
バーの停止挙動

平均速度を多項式近似

$$\bar{v}_b(l) = \sum_{i=0}^5 c_i l^i$$

$\bar{v}(l)$  に対応する加速度は

$$\bar{a}_b(l) = \frac{\partial \bar{v}_b(l)}{\partial l} \bar{v}_b(l)$$

一般の停止挙動についても

$$a_b(l) = \frac{\partial v_b(l)}{\partial l} v_b(l)$$

停止線までの距離を  $L_0$  とし

$$\frac{\partial \bar{v}_b(l)}{\partial l} \cong \frac{\bar{v}_b(l)}{L_0}, \quad \frac{\partial v_b(l)}{\partial l} \cong \frac{v_b(l)}{L_0}$$

と近似して  $L_0$  を消去すると

$$a_b(l) = \bar{a}_b(l) \left( \frac{v_b(l)}{\bar{v}_b(l)} \right)^2$$

## 評価関数, 拘束条件

$$J[u(\tau)] = \int_t^{t+T} L(x(\tau), u(\tau), \tau) d\tau$$

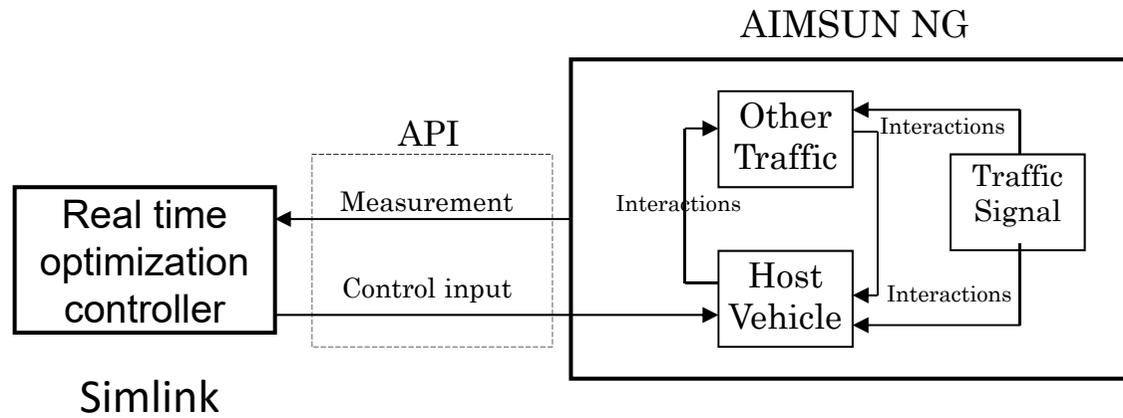
$$L = w_1 \frac{F_V(\tau)}{v_h(\tau)} + w_2 (R) R_e^2(\tau) + w_3 (v_h(\tau) - v_d)^2$$

$$C(u(\tau), u_d(\tau)) = \frac{1}{2} (u^2(\tau) + u_d^2(\tau) - u_{\max}^2) \quad \leftarrow \quad |u| \leq u_{\max}$$

$$R_e(\tau) = h_d v_h(\tau) - x_p(\tau) + x_h(\tau) + l_0 \quad \leftarrow \quad \text{車頭時間 } h_d \text{ に応じた車間距離}; \quad \text{マージン}$$

$$w_2(\tau) = \gamma e^{-\alpha R(\tau)} \quad \leftarrow \quad \text{遠い先行車の影響を減少させる}; \quad \alpha, \gamma: \quad \text{定数}$$

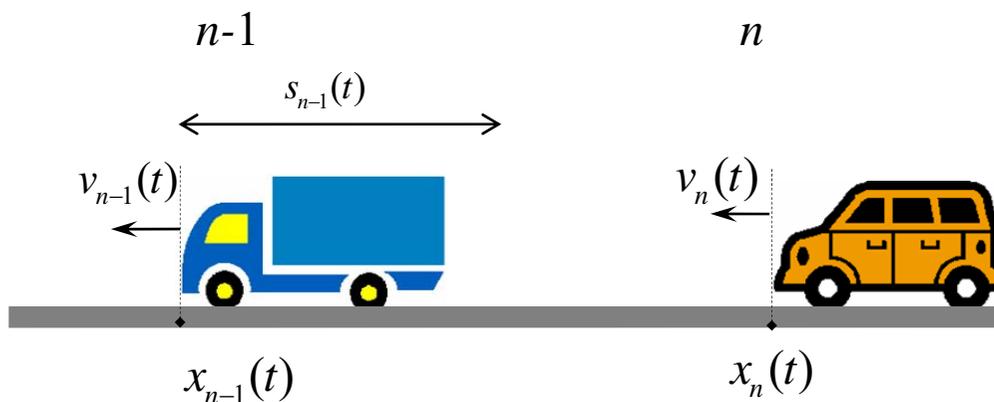
# 交通流シミュレータ(AIMSUN NG)を用いた計算機シミュレーション



ミクروسコピック交通  
流シミュレータ



# Gippsモデル



$$v_n(t + \tau) = \min \left[ \begin{array}{l} v_n(t) + 2.5a_n\tau\{1 - v_n(t)\} / V_n(0.025 + v_n(t)/V_n)^{1/2}, \\ b_n\tau + \left[ b_n^2 - b_n \left\{ 2(x_{n-1}(t) - s_{n-1} - x_n(t)) - v_n(t)\tau - v_{n-1}^2(t) / \hat{b} \right\} \right]^{1/2} \end{array} \right] \begin{array}{l} \leftarrow \text{自由走行} \\ \leftarrow \text{追従走行} \end{array}$$

$a_n$ : 車両 $n$ のドライバーが許容する最大加速度

$b_n$ : 車両 $n$ のドライバーが許容する最大減速度

$s_n(t)$ : 車両 $n$ の実効車長

$V_n$ : 車両 $n$ の実効車長目標速度

$x_n(t)$ : 車両 $n$ の位置

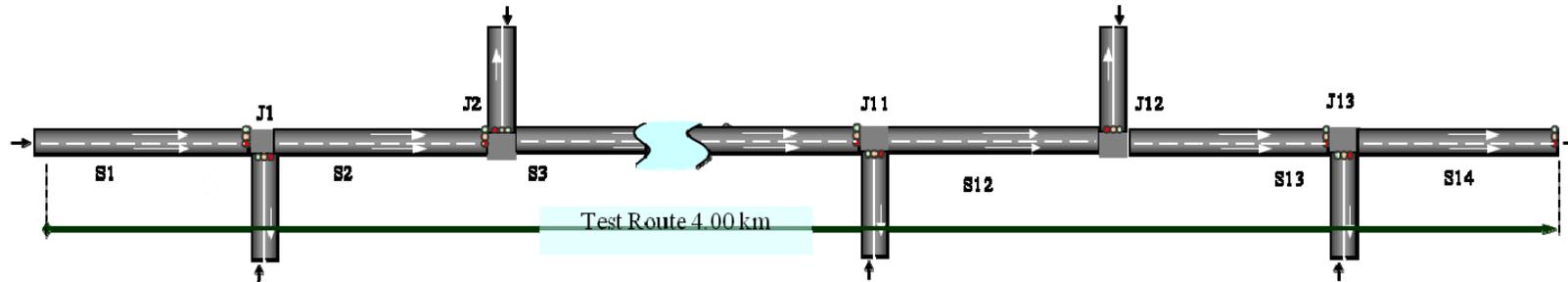
$v_n(t)$ : 車両 $n$ の速度

$\tau$ : ドライバーの反応時間

$\hat{b} := \min(-3.0, (b_n - 3)/2)$

## 道路条件

Simulation environment is created in AIMSUN Simulator



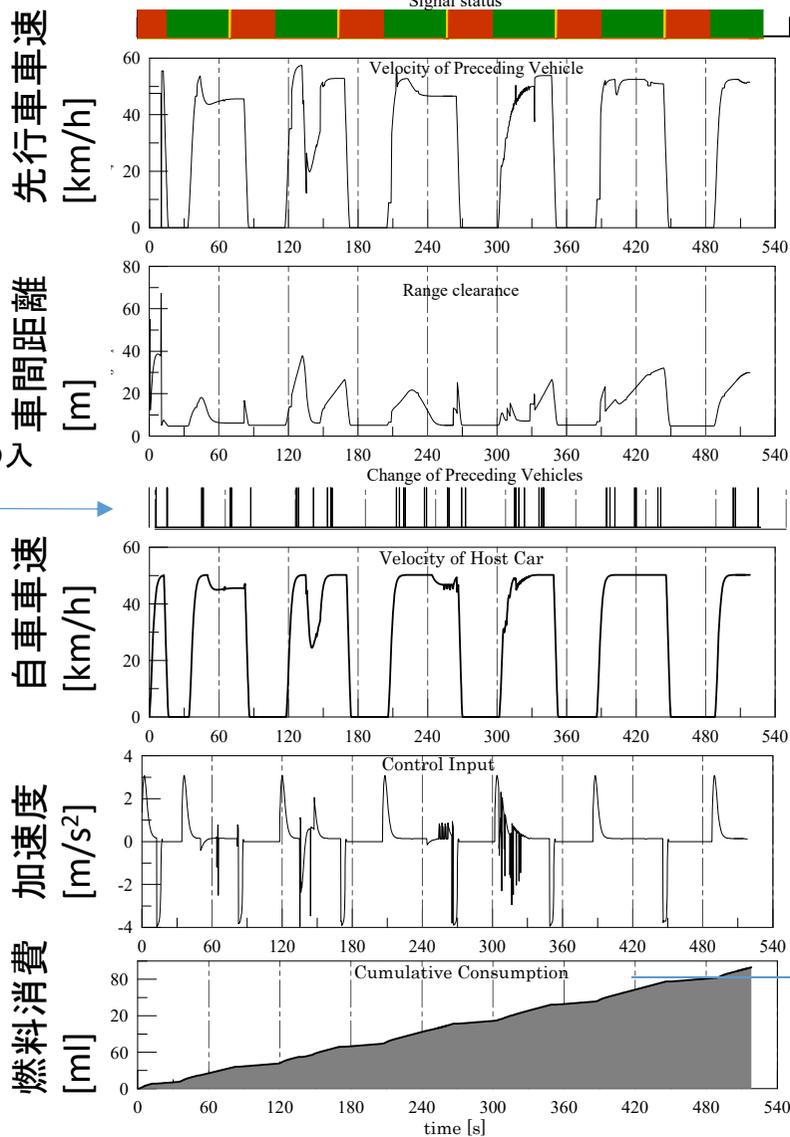
Number of Sections: 14  
Number of Junctions: 13  
Length of the Route : 4.00 [km]  
Number of Lanes:  
2 lanes in S1 to S8 and S11 to S14  
3 lanes in S9 and S10

Traffic signals:  
synchronous 90[s] cycle; 50 [s] Green 2 [s] Yellow  
Traffic Density: 2000 to 3000 [#Vehicle/h]  
Road Speed limit: 50 [km/h]

# 計算機シミュレーション結果の一例

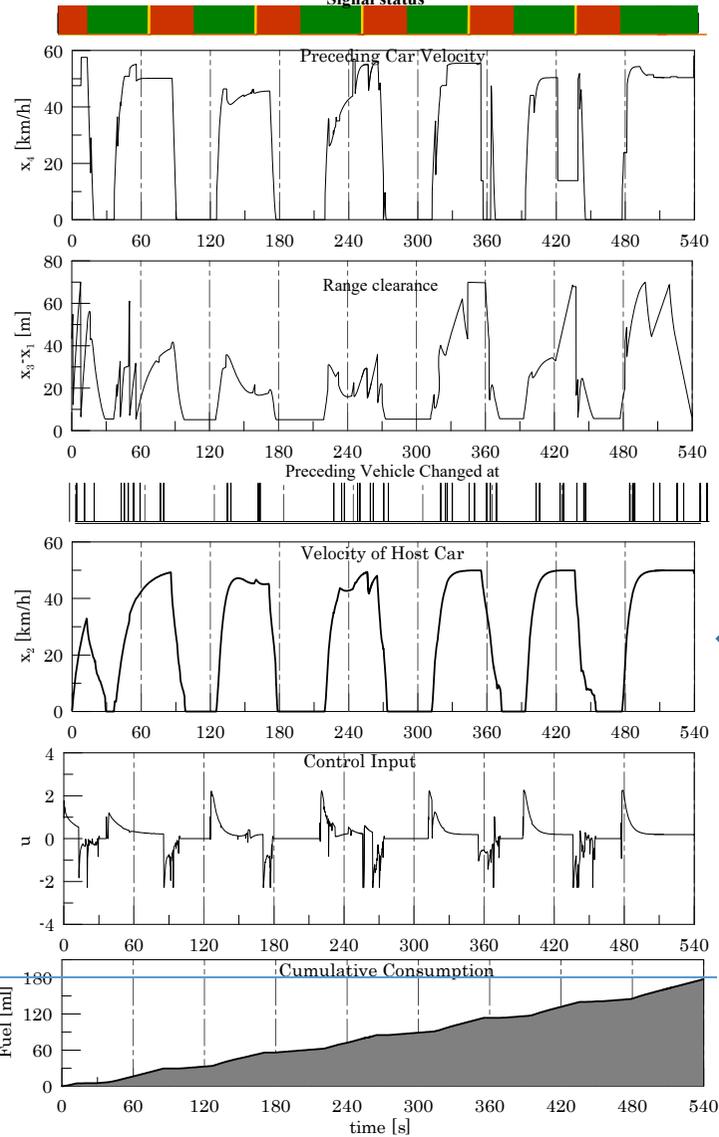
## Gipps

Signal status



## 実時間最適制御(RTO)

Signal status

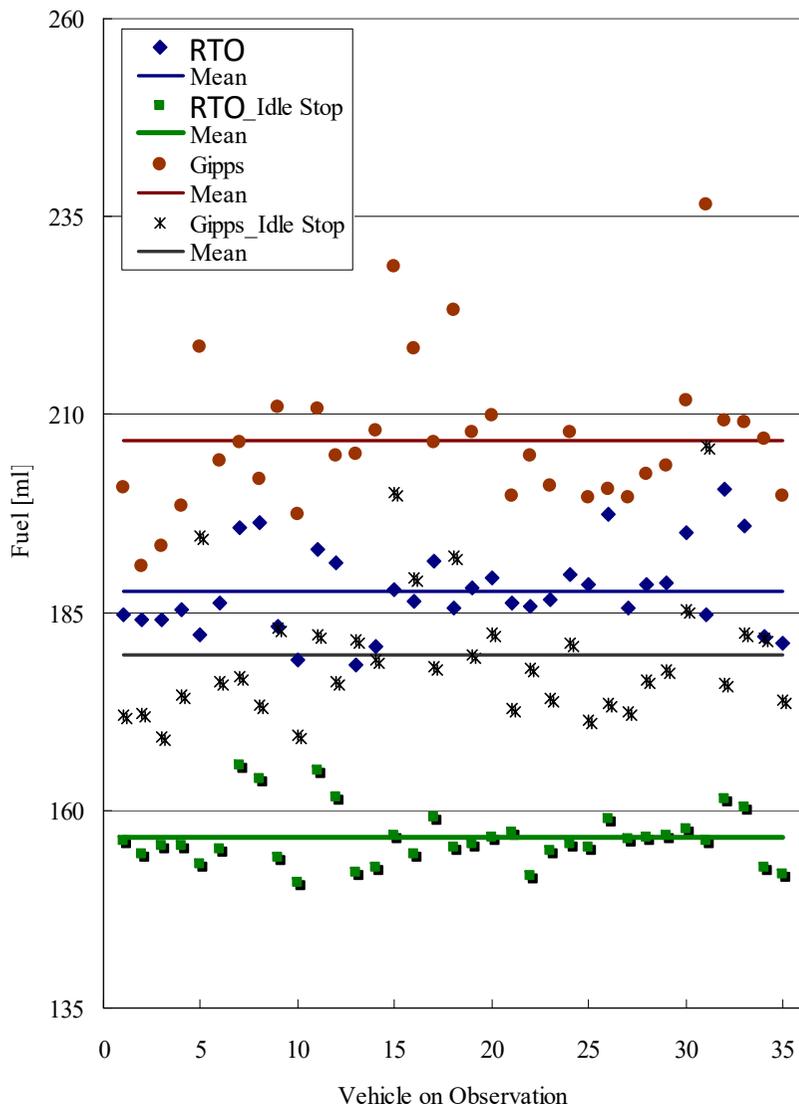


← 長め

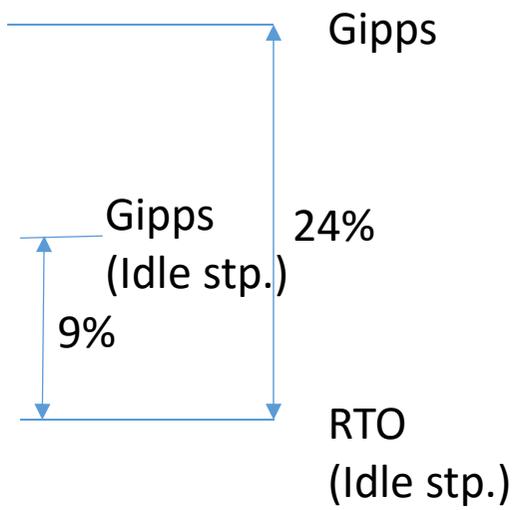
← 緩やか

← 小振幅

# 計算機シミュレーション(35ケース)

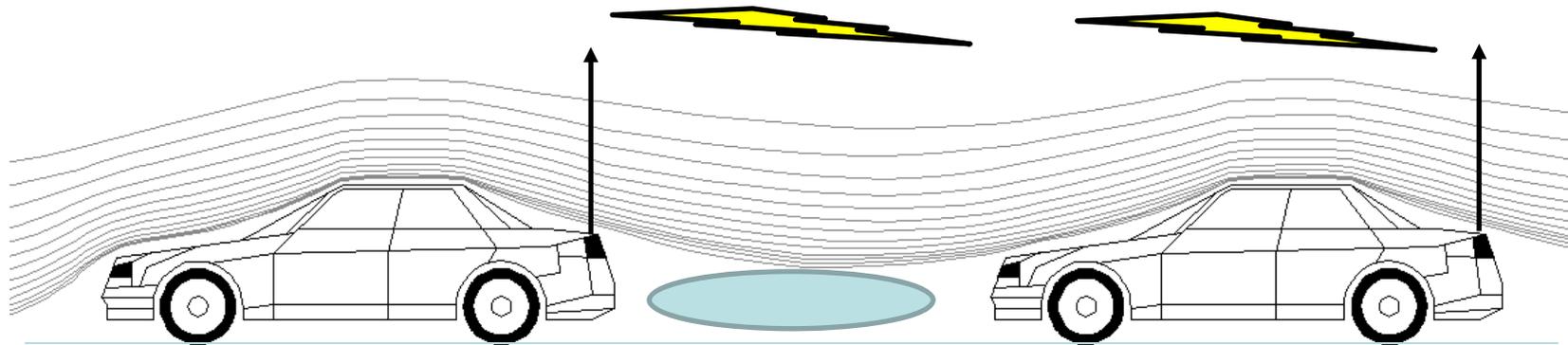


	RTO	Gipps	
Fuel [ml]	187.68	206.78	-9.24%
Fuel (idle stop)	156.44	178.97	-12.85%
Time	602.23	510.78	+17.9%
Stops	7.30	6.06	+1.24



# 省燃費運転への応用 その2

## 高速道路での隊列走行による省燃費化



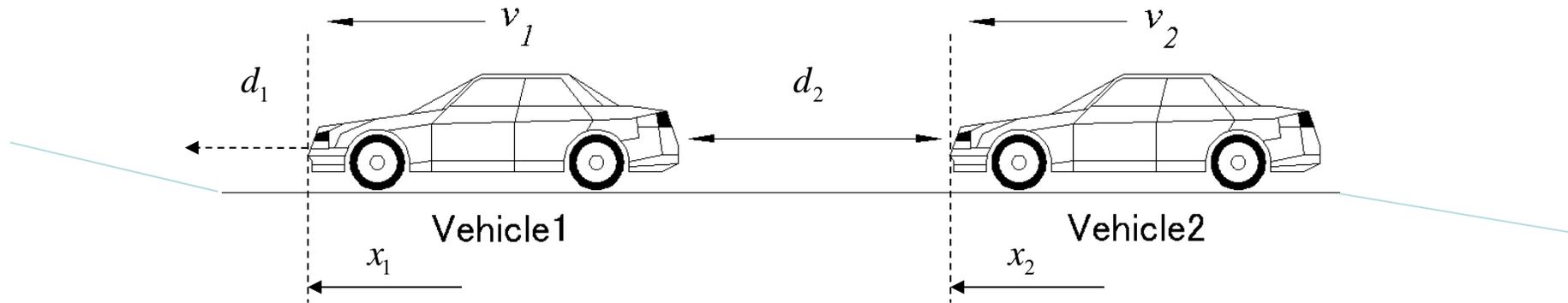
- スリップストリーム走法による省燃費化
- 隊列を構成する車両を**協調的に制御する**  
⇔  
**車群トータル**の燃費を最小化する制御を考える

## 提案する制御手法の特徴

基盤技術の進歩:ITS情報の積極利用, 車載CPUの高性能化, 最適制御問題の高速解法の進歩

- 車群トータル燃費を低減する.  
(車車間通信の利用)
- 空気抵抗係数の車間距離依存性
- 勾配, 道路形状情報  
を用いた最適化  
⇒非線形最適化問題に基づくモデル予測制御

# 車群のモデリング(1)



## 車両の運動方程式

$$\dot{v}_i = -\frac{1}{m_i} f_{ai} - \frac{1}{m_i} f_{gi} - \frac{1}{m_i} f_{\mu i} + a_i,$$

$$\dot{a}_i = -\tau_i a_i + \tau_i u_i,$$

$$f_{ai} = -\frac{1}{2} \rho A_i C_i(d_i) v_i^2, \quad f_{gi} = m_i g \sin(\theta(x_i))$$

車体の質点近似

エンジン・ブレーキの  
応答遅れ

勾配情報

空気抵抗の車間距離依存性

# 車群のモデリング(2)

## 車群(車両2台)の状態方程式

$$\dot{x} = f(x, u),$$

$$x = [x_1 \quad v_1 \quad a_1 \quad x_2 \quad v_2 \quad a_2]^T, u = [u_1 \quad u_2]^T$$

$$f(x, u) = \begin{bmatrix} v_1 \\ -\frac{1}{m_1}(f_{a1} + f_{g1} + f_{r1}) + a_1 \\ \frac{1}{\tau_1}(-a_1 + u_1) \\ v_2 \\ -\frac{1}{m_2}(f_{a2} + f_{g2} + f_{r2}) + a_2 \\ \frac{1}{\tau_2}(-a_2 + u_2) \end{bmatrix}$$

- 入力は車両1および車両2の加減速度指令値
- 車両1, 車両2の協調制御を考える

# 制御系の設計

## 最適制御問題

評価関数  $J$  を最小化する入力  $u$  を求める ( $T$  時間分)

$$\begin{aligned}
 & J = \int_t^{t+T} L dt \quad : \text{評価関数} \\
 & L = w_u L_u + w_d L_d + w_v L_v + w_s L_s \\
 & L_u = \frac{\sqrt{u_1^2} + u_1}{2} + \frac{\sqrt{u_2^2} + u_2}{2}, \quad L_d = \frac{1}{2} (d_2 - d^*)^2, \\
 & L_v = \frac{1}{2} (v_2 - v^*)^2, \quad L_s = \frac{1}{2} (v_1 - v_2)^2 \\
 & u_1^2 + u_2^2 \leq u_{\max}^2 \quad : \text{入力の拘束} \\
 & \dot{x} = f(x, u) \quad : \text{状態方程式}
 \end{aligned}$$

入力(加速)の評価  
 目標車間距離  
 相対速度の評価  
 目標車速

# 制御系の設計

## モデル予測制御系 (Receding Horizon Control)

Step1: 更新時刻 $t_1$  に初期状態として $x(t_1)$ を検出する.

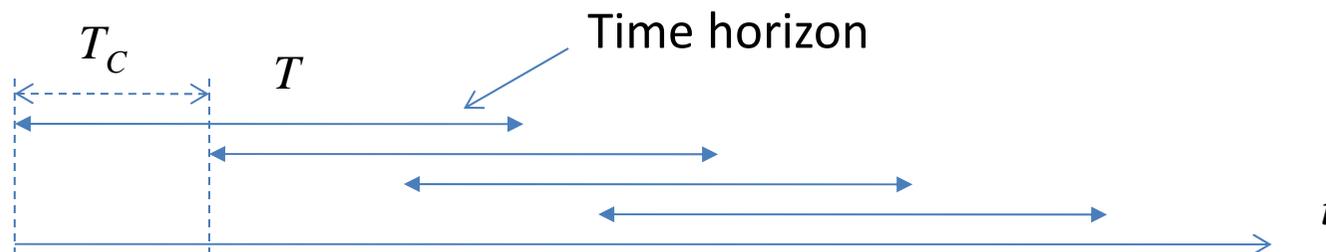
Step2:  $x(t_1)$  を初期状態とし, 最適制御問題を解き, 予測区  
間 $t_1 \leq t \leq t_1 + T$ の間の最適制御入力列

$$\{u^*(\tau) \mid t_1 \leq \tau \leq t_1 + T\}$$

を計算する.

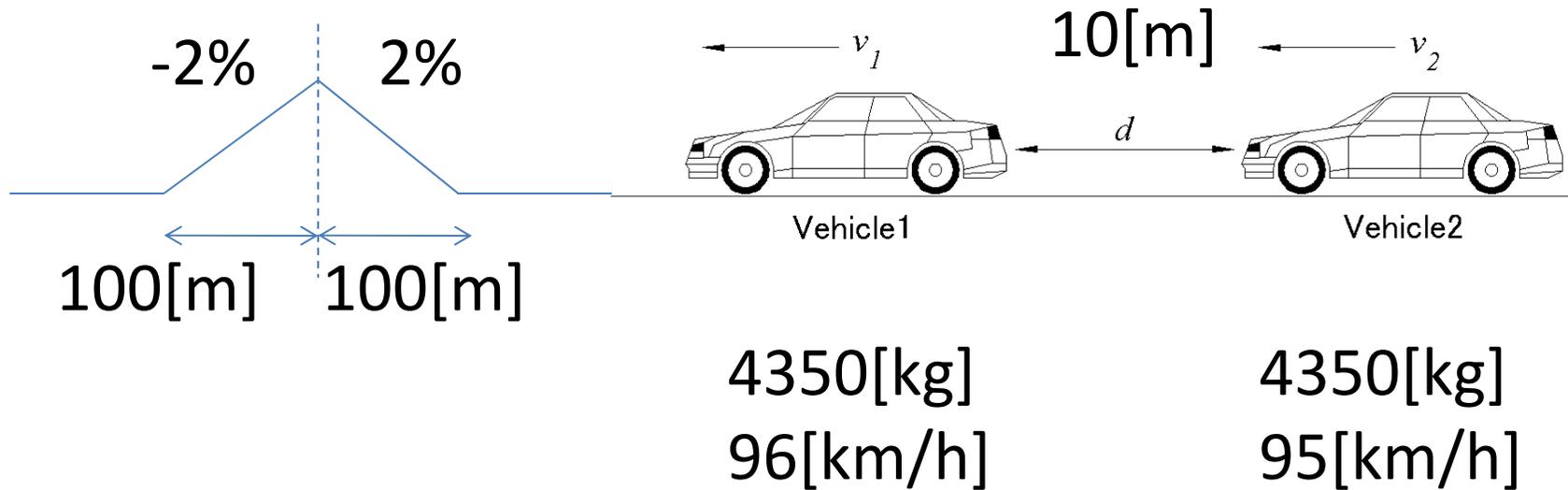
Step3: 現在時刻 $t$ から $t + T_c$ の間, 制御入力を次のように発生する.

$$u(t) = u^*(t), \quad t_1 \leq t < t_1 + T_c$$



# 計算機シミュレーション

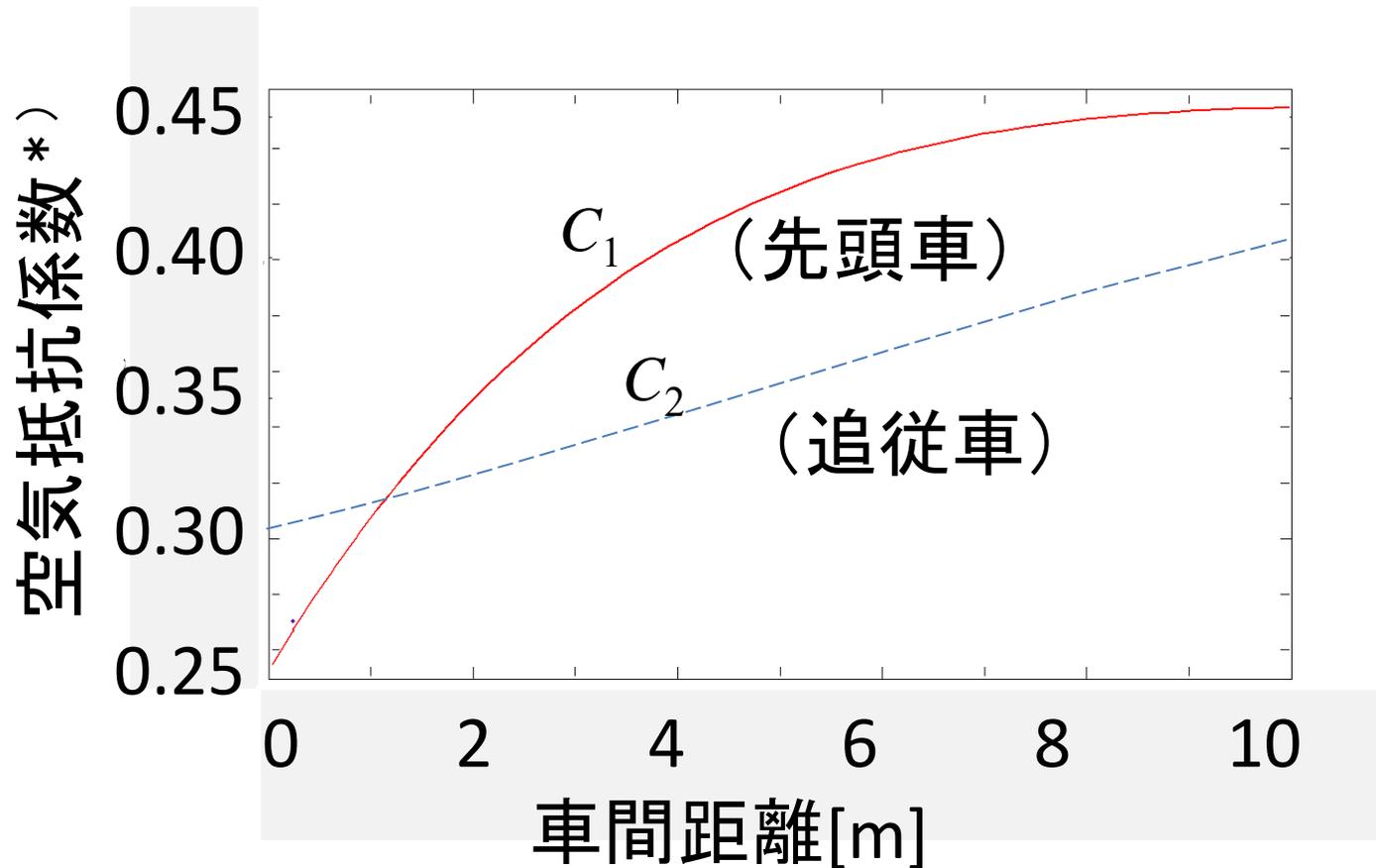
## 初期状態・パラメータ



目標車間距離: 2[m]

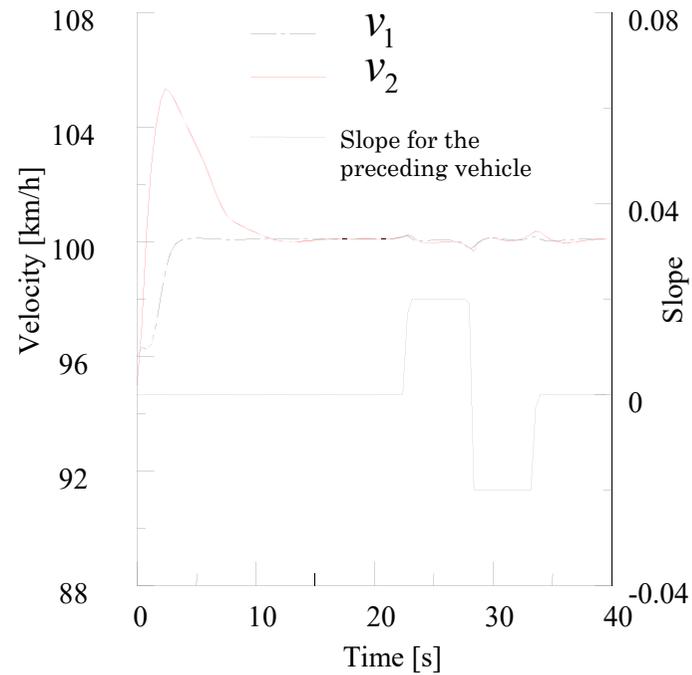
目標車速: 100[km/h]

# 計算機シミュレーション

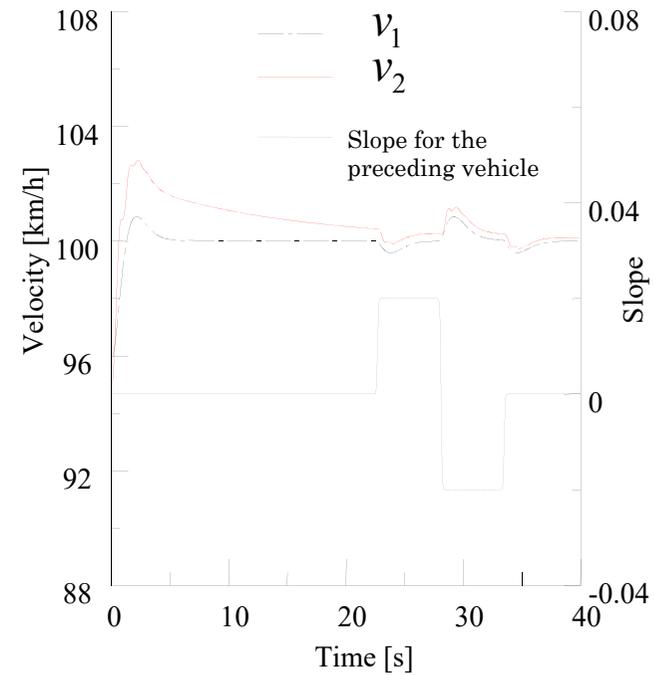


\* ) Michael Zabat, Nick Stabile, Stefano Frascaroli, Frederick Browand: The Aerodynamic Performance of Platoon: Final Report, California PATH Research Report UCB-ITS-PRR-95-35 A1-7 (1995)

# 計算機シミュレーション

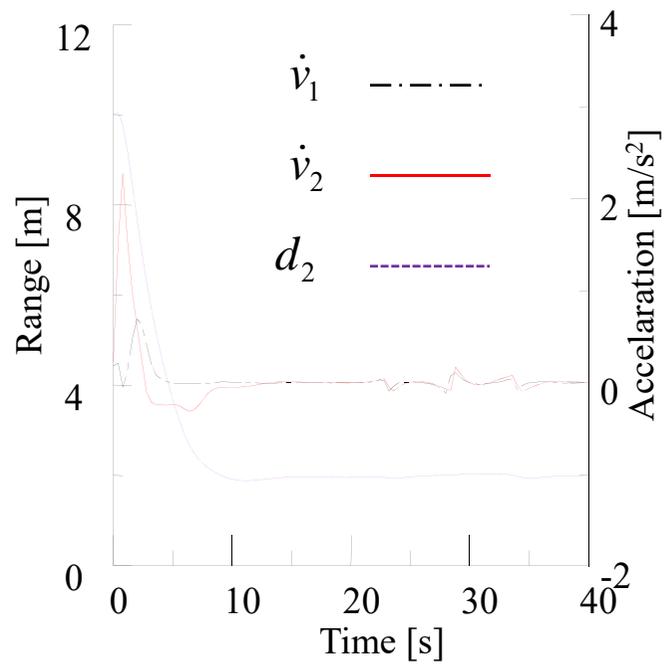


提案法

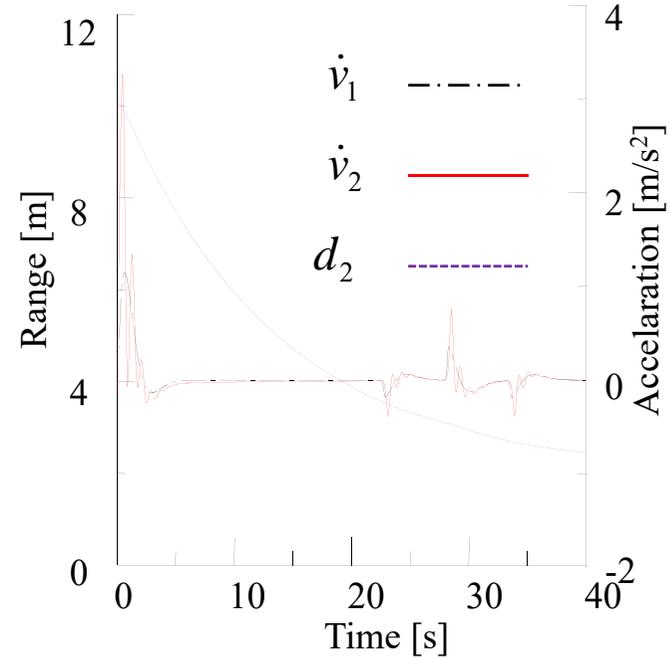


車頭時間一定制御  
(先頭車はASCD:PI速度制御)

# 計算機シミュレーション



提案法



車頭時間一定制御  
(先頭車はASCD:PI速度制御)

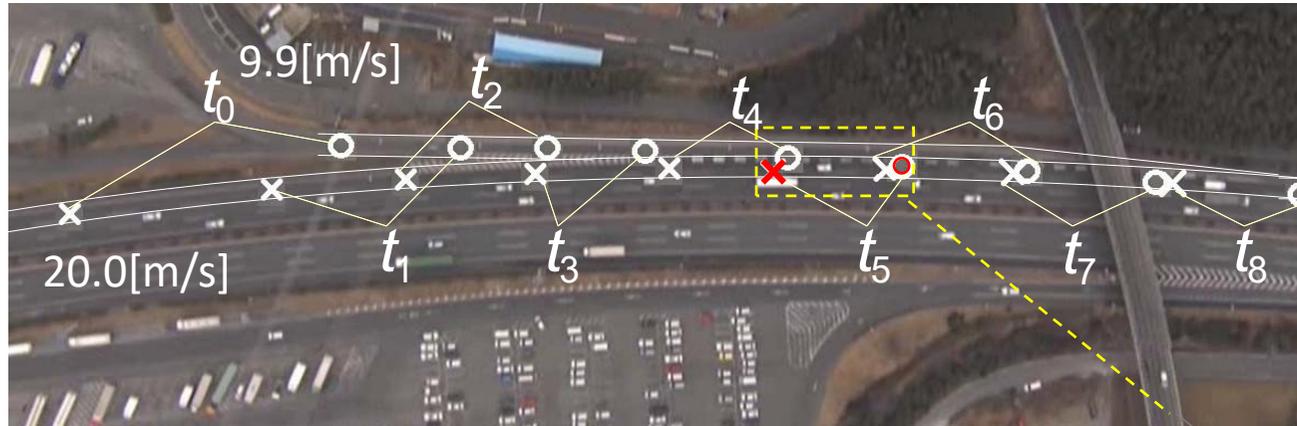
# 計算機シミュレーション

## エネルギー消費(提案手法比)

Time	提案法	車頭時間一定制御 (先頭ASCD)	ASCD 単独走行
$0 \leq t \leq 20$ (車群形成)	1.91[MJ]	1.99[MJ] (+4.1%)	2.01[MJ](+5.3%)
$20 < t \leq 40$ (勾配通過)	1.55 [MJ]	1.56[MJ] (+1.7%)	1.73[MJ](+11.7%)
$0 \leq t \leq 40$ (全区間)	3.46[MJ]	3.57[MJ] (+3.1%)	3.75[MJ](+8.2%)

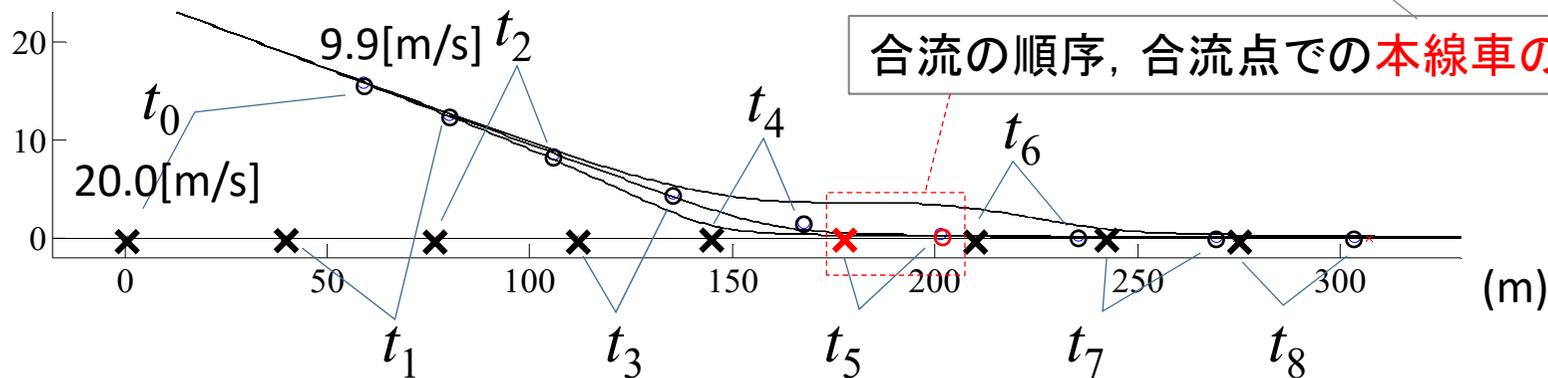
# 実時間最適制御による合流のシミュレート

仮説:合流の課程は、合流車両と本線車両の実時間同時最適化制御の結果である。



- : 合流車
- × : 本線車
- $t_i = 2i, i = 0, 1, \dots, 8$  [s]

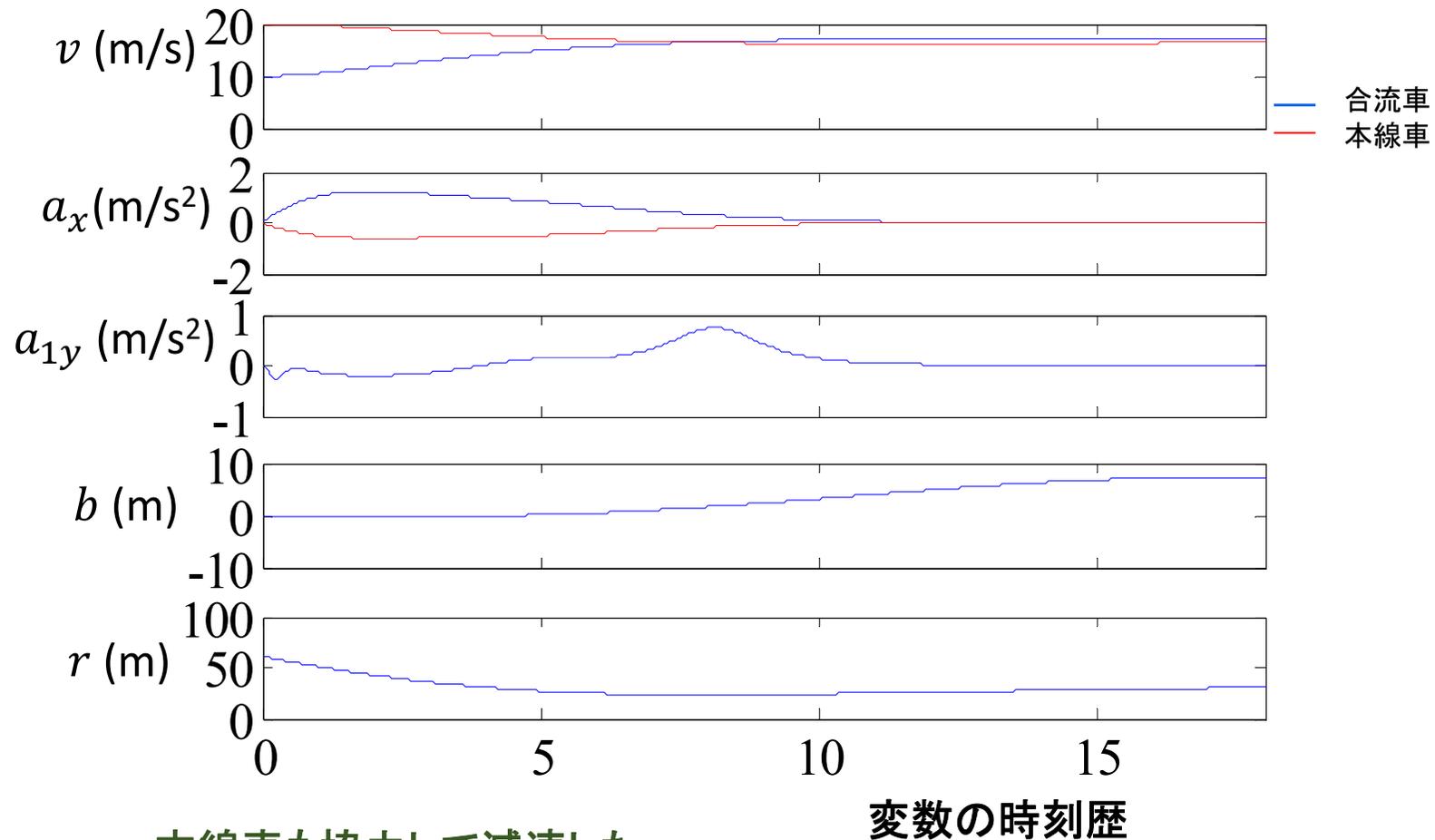
合流の空撮(九州道 基山ジャンクション)



実時間最適化制御による合流軌道の生成(計算機シミュレーション)

# 計算機シミュレーション

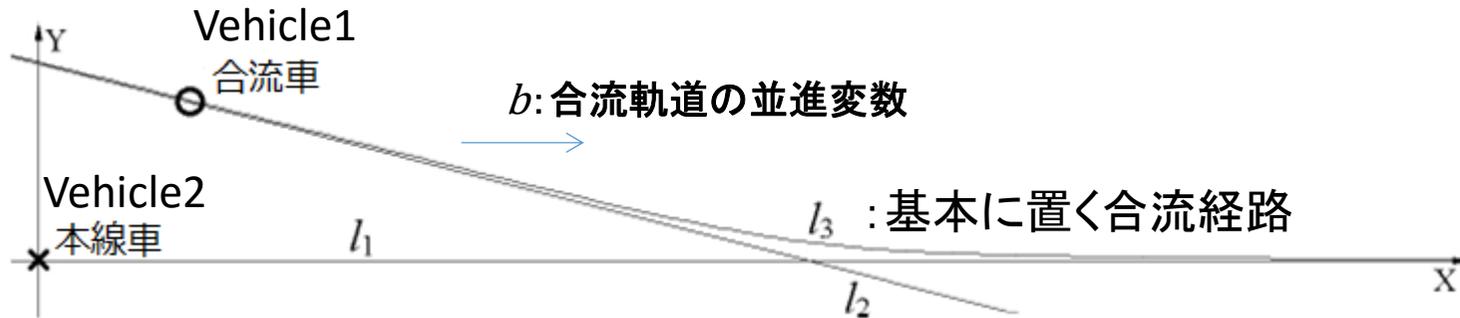
## 実際の合流の再現



- 本線車も協力して減速した。
- 加速度は穏やかだった。
- 本線車の加速度は合流車より強くない
- 合流の後本線車の速度も合流車の速度も目標車速に収束した。

# 問題の定式化(モデル化)

## 道路の近似



$$l_1: y = 0 \qquad l_2: y = k(x - \beta) \qquad l_3: y = \frac{k}{2} \left( (x - \beta) - \left( (x - \beta)^2 - \frac{\alpha}{k} \right)^{\frac{1}{2}} \right)$$

## 状態方程式 車両は質点近似

$$\frac{d}{dt} x = \begin{bmatrix} v_{1x} \\ v_{12} \\ v_b \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ a_{1x} \\ a_{12} \\ a_b \end{bmatrix}$$

$i=1$  合流車  
 $i=2$  本線車

合流軌道の修正

$$x = [x_{1x} \quad x_{2x} \quad b \quad v_{1x} \quad v_{2x} \quad v_b]^T$$

$$a = [a_{1x} \quad a_{2x} \quad a_b]^T$$

## 拘束条件

### 合流車の位置

$$x_{1y} = \frac{k}{2} \left( (x_{1x} - \beta - b) - \left( (x_{1x} - \beta - b)^2 - \frac{\alpha}{k} \right)^{\frac{1}{2}} \right)$$

### 入力

$$\begin{cases} a_{1xmin} \leq a_{1x} \leq a_{1xmax} \\ a_{2xmin} \leq a_{2x} \leq a_{2xmax} \\ a_{bmin} \leq a_b \leq a_{bmax} \end{cases}$$

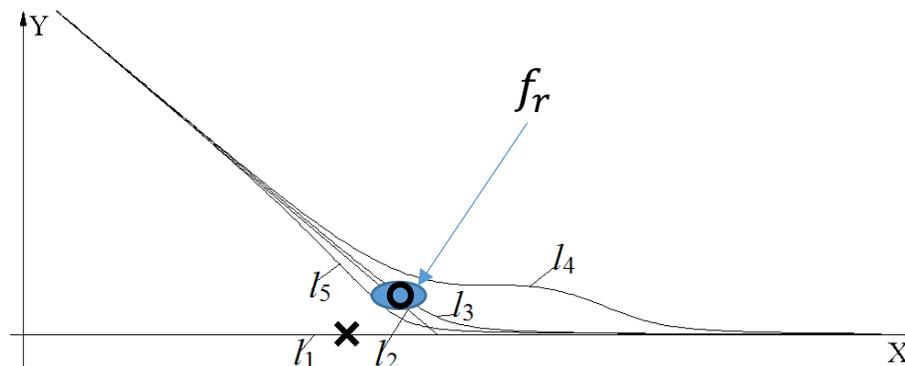
本線車両の加速度は小さめに

# 問題の定式化(最適制御問題による表現)

## 最適制御問題

$$J = \int_t^{t+T} L(x(\tau), a(\tau), \tau) d\tau$$

$$\min_a J$$



拘束条件:自動車が発生しうる加速度・減速度の範囲内で運動

## 評価関数

$$L = \frac{\omega_1}{f_r}$$

衝突を防止する。

$$f_r = ((x_{1x} - x_{2x})^2/a_1^2 + x_{1y}^2/a_2^2)^{0.5} - 1 \leq 0$$

進入禁止領域

$-\omega_2 \log(b - b_{min} f_1(x_1)) - \omega_3 \log(b_{max} f_2(x_2) - b)$  合流車が $l_4$ と $l_5$ の間を走る。

$+\omega_4 (v_1 - v_{1d})^2 + \omega_5 (v_2 - v_{2d})^2$  目標車速で走る。

合流車両は

- ・合流前後は車線の中央を走る。
- ・加速車線内で合流経路を調整する。

$+\omega_6 a_1^2 + \omega_7 a_2^2 + \omega_8 a_b^2$  加速度を小さく、滑らかにする。

$$f_i(x) = k_{bi} \left( \frac{1}{1 + \exp(-\alpha_{bi}(x_1 - \beta_{bli}))} + \frac{1}{1 + \exp(\alpha_{bi}(x_1 - \beta_{bri}))} - 1 \right), (i = 1, 2)$$

# 計算機シミュレーションパラメータ

初期条件

$x_{1x}$	$x_{2x}$	$v_1$	$v_2$
59.0m	0m	9.9m/s	20.0m/s

合流車加速度の重み

設定したパラメータ

進入禁止領域は円:  $a_1 = a_2 = 14$

重み設定

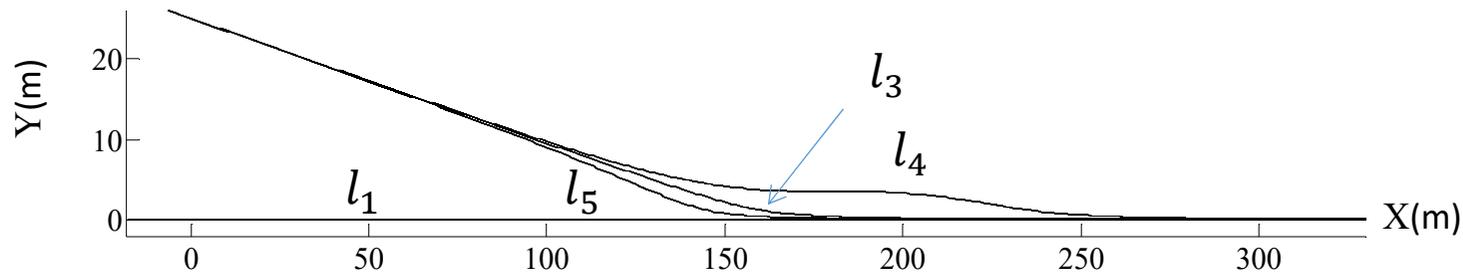
$\omega_1$	$\omega$	$\omega_3$	$\omega_4$	$\omega_5$	$\omega_6$	$\omega_7$	$\omega_8$
0.076	0.001	0.0011	0.004	0.004	0.024	0.0241	0.001

本線車加速度の重み

チューニング

形状パラメータ

$k$	$\beta$	$\alpha$	$b_{min}$	$b_{max}$	$k_{b1}$	$k_{b2}$	$\alpha_{b1}$	$\alpha_{b2}$	$\beta_{bl1}$	$\beta_{bl2}$	$\beta_{br1}$	$\beta_{br2}$	$v_{1d}$	$v_{2d}$
-0.156	159.8	40.0	-13.07	80.10	2.15	1.00	0.05	0.05	140	180	180	840	16.7	16.7



道路のモデル化結果

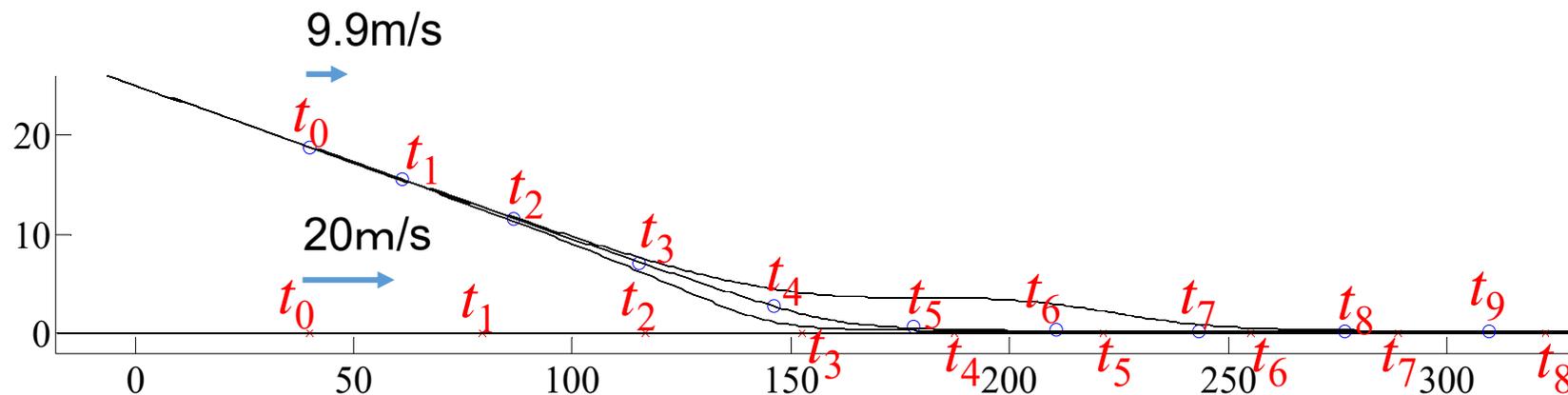
# 他の状況での計算機シミュレーション

## 本線車が先行車になると思われるケース

初期条件

$x_{1x}$	$x_{2x}$	$v_1$	$v_2$
40.0m	40.0m	9.9m/s	<u>20.0m/s</u>

進入禁止領域は円:  $a_1 = a_2 = 14$



シミュレーションの合流経路

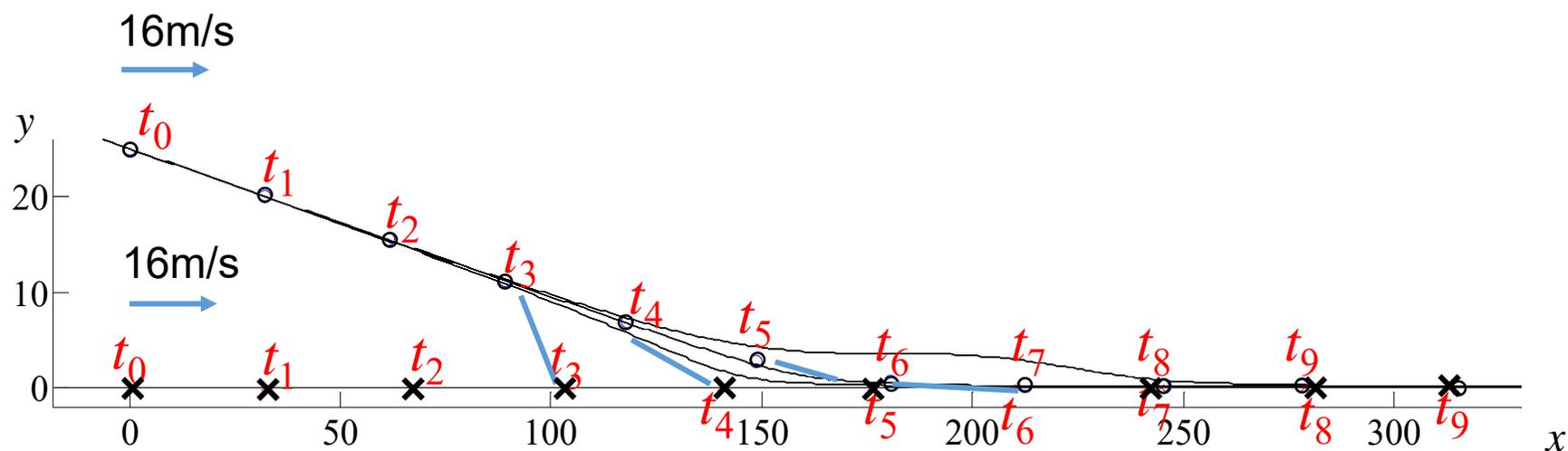
- 合流は成功した。
- 合流車はあらかじめ予測されたとおり、本線車の後ろに合流した。

# 他の状況での計算機シミュレーション

## 初期値のままでは衝突するケース

初期値

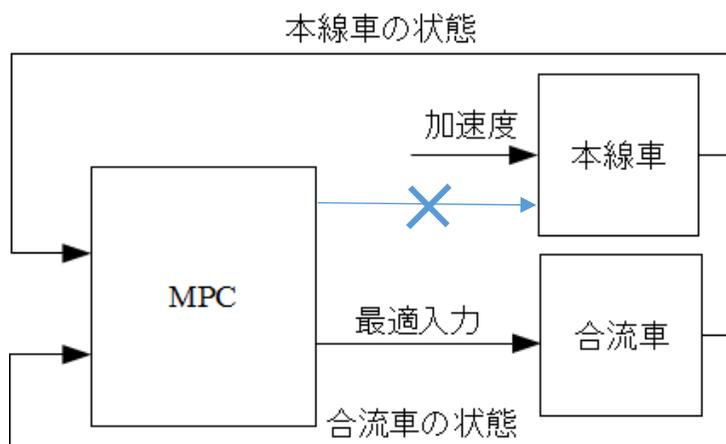
$x_{1x}$	$x_{2x}$	$v_1$	$v_2$
0m	0m	16.0m/s	16.0m/s



ケース3のシミュレーションの合流経路

- 合流車が減速して、本線車の後ろに合流して衝突回避した。

# ロバスト性検証：予測がはずれる場合



## 実際の合流？

- 合流車で最適解を計算
- 本線車の最適挙動は合流車で本線車の運動予測値として使用
- 本線車は最適解のとうりには運動しない
- フィードバックの作用で衝突を防ぐ



# 計算機シミュレーション

本線車が等加速度運動で衝突コースをとる場合

初期値

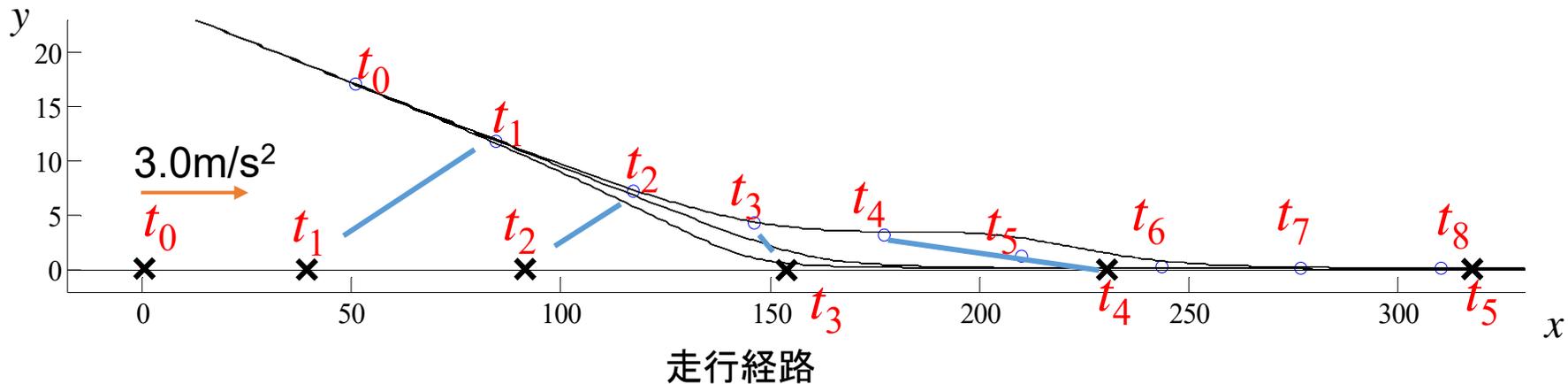
$a_{2x}$	$x_{1x}$	$x_{2x}$	$v_1$	$v_2$
3.0m/s <sup>2</sup>	51.0m	0m	16.7m/s	16.7m/s

進入禁止領域は楕円:  $a_1 = 16, a_2 = 2$   
 シミュレーションのサンプル時間  $h = 0.01$  (s),  
 予測ホライズン  $T = 3$  (s)

重み設定

$\omega_1$	$\omega_2$	$\omega_3$	$\omega_4$	$\omega_5$	$\omega_6$	$\omega_7$	$\omega_8$
0.9	0.001	0.0011	0.004	0.004	0.001	1.0	0.001

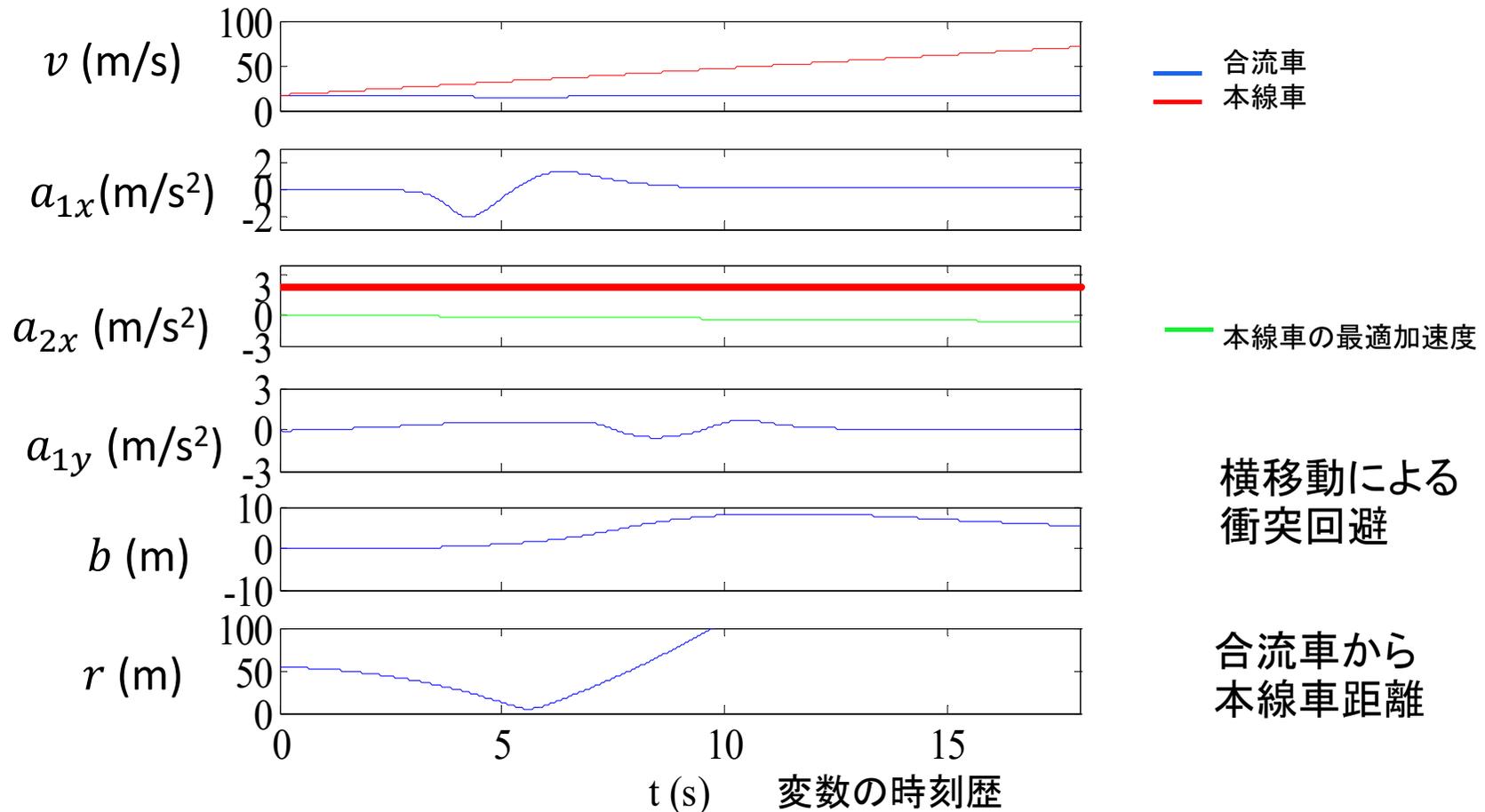
シミュレーション結果



- 合流車は車間距離を保つために減速, 横方向に移動。

# 計算機シミュレーション

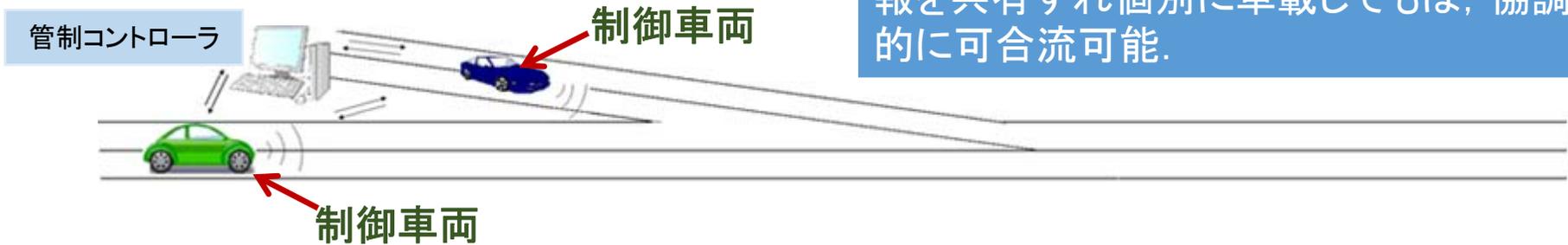
本線車は等加速度運動で衝突コースをとる場合



- 最適化された加速度と実際の加速度の偏差は3.0(m/s<sup>2</sup>)から少しずつ大きくなっていった。
- 加速度は穏やかだった。

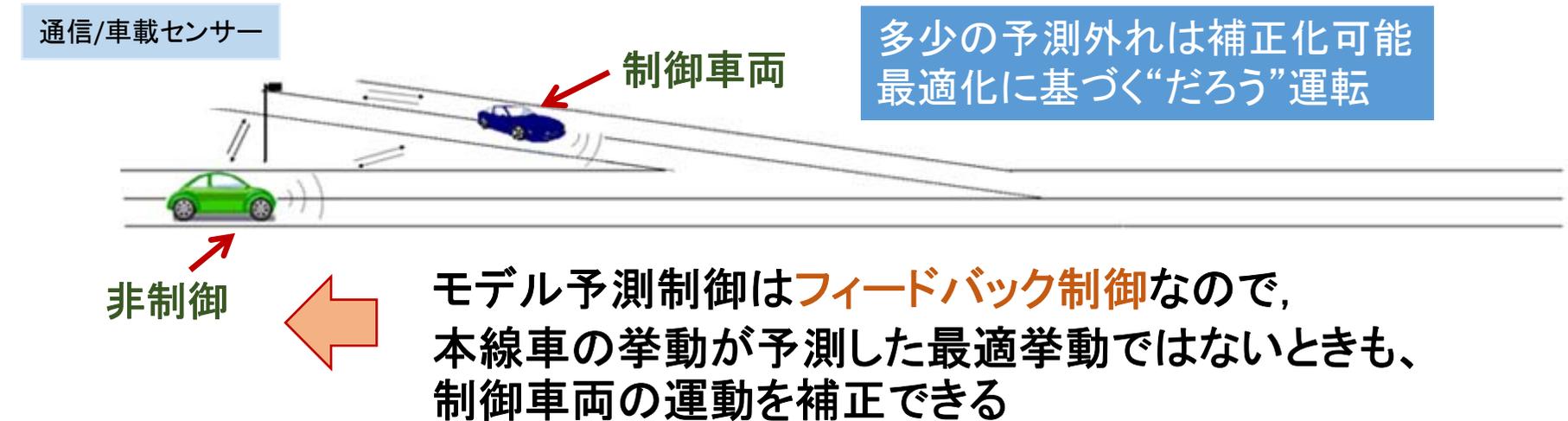
仮定: 実時間最適制御の解を使って車両の動きが最適化・予測できる

## 1. 合流のインフラによる管制



おなじ合流ルール(制御則)のもとに, 情報を共有すれば個別に車載してもば, 協調的に可合流可能.

## 2. インフラ情報を利用した車載型自動合流システム



多少の予測外れは補正化可能  
最適化に基づく“だろう”運転

# まとめ：自動車制御の目指すところ

