

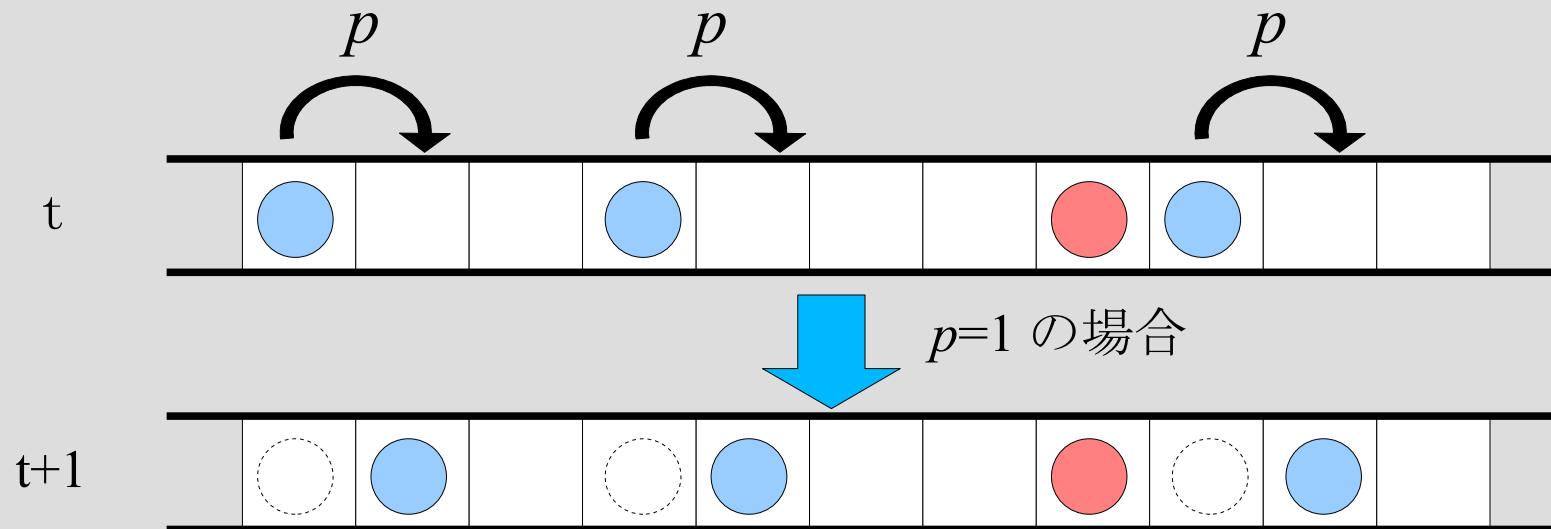
Google Earth を用いた交通渋滞 シミュレーションシステムの開発

篠原 修二

交通流モデル 1/6

- Asymmetric Simple Exclusion Process (ASEP)

⇒ 離散化された Cellular Automaton(CA) モデル

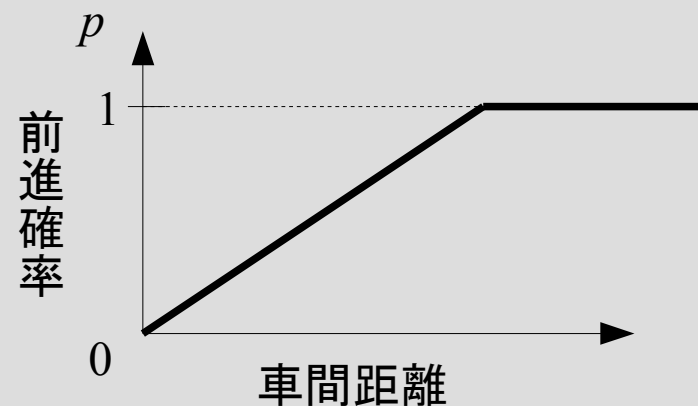
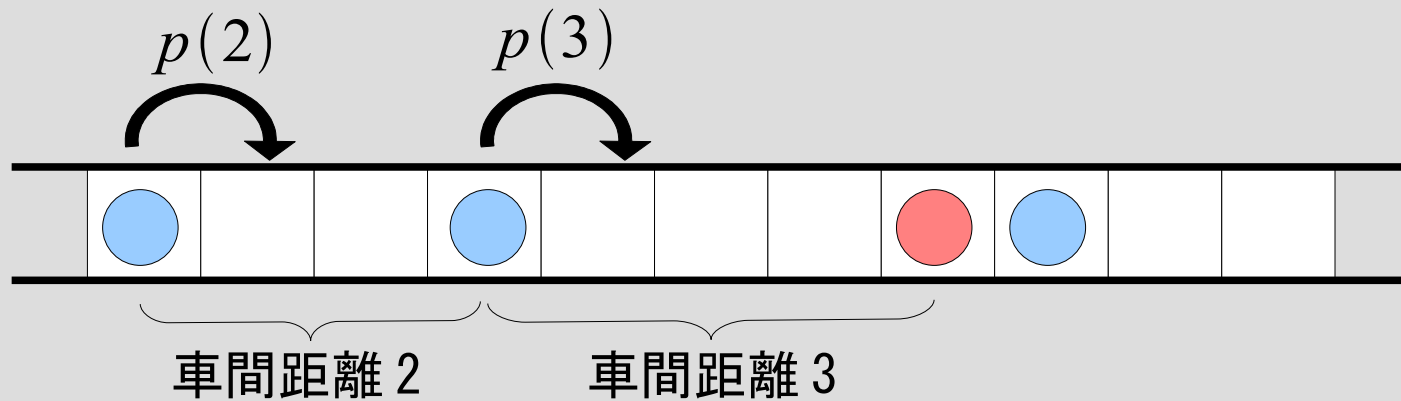


- 道路を Cell に分割し, 空間を離散化
- 各 Cell には, 最大 1 台の車のみが存在可能
- 前方の Cell が空いている時, 確率 $p(=const)$ で前進

交通流モデル 2/6

- Zero Range Process (ZRP)

⇒ 前進確率 p は前の車との車間距離に依存

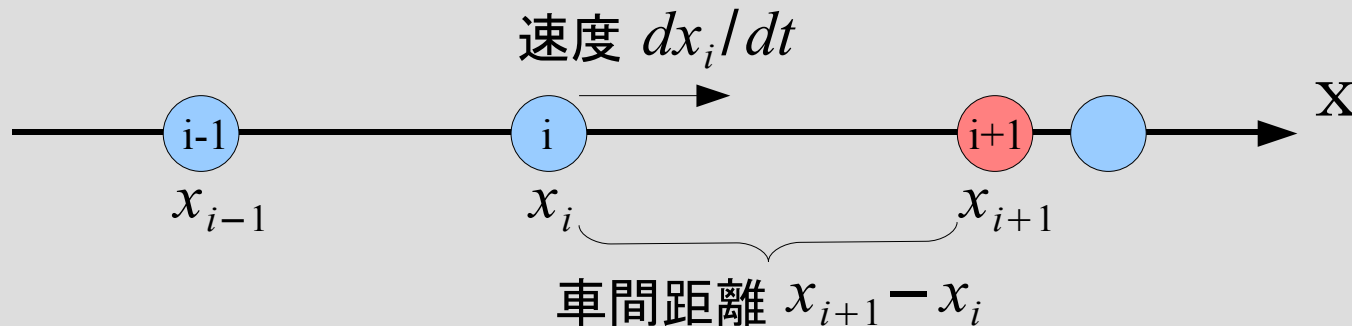


交通流モデル 3/6

- Optimal Velocity (OV) モデル

⇒ 連続かつ決定論的モデル

⇒ 各車両は、前の車との車間距離に応じた最適速度 V に近づけるように速度を調整



$$d^2 x_i / dt^2 = a [V(x_{i+1} - x_i) - dx_i / dt]$$

$x_i = x_i(t)$: 時刻 t における車 i の位置

V : Optimal Velocity Function

a : パラメータ

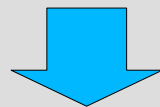
交通流モデル 4/6

- Stochastic Optimal Velocity(SOV) モデル
⇒ OV モデルの離散化、確率モデル化

■ 離散化

$$d^2 x_i / dt^2 = a [V(x_{i+1} - x_i) - dx_i / dt]$$

OV モデル



離散化

$$v_i^{t+\Delta t} = (1 - a \Delta t) v_i^t + a \Delta t V(\Delta x_i^t)$$



$\Delta t \rightarrow 1$

$$v_i^{t+1} = (1 - a) v_i^t + a V(\Delta x_i^t)$$

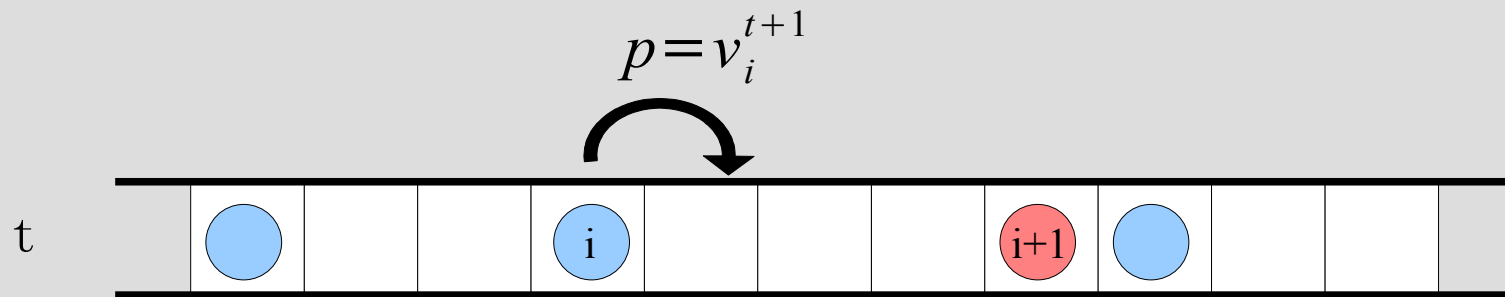
v_i^t : 時刻 t における車 i の速度
 Δx_i^t : 車間距離

交通流モデル 5/6

Stochastic Optimal Velocity(SOV) モデル (続き)

確率モデル化

→ 前方の Cell が空いている時, 確率 $p(= v_i^{t+1})$ で前進
ただし $\forall i, t, 0 \leq v_i^t \leq 1, 0 \leq V(\Delta x_i^t) \leq 1$



整理すると

$$v_i^{t+1} = (1-a)v_i^t + aV(\Delta x_i^t)$$

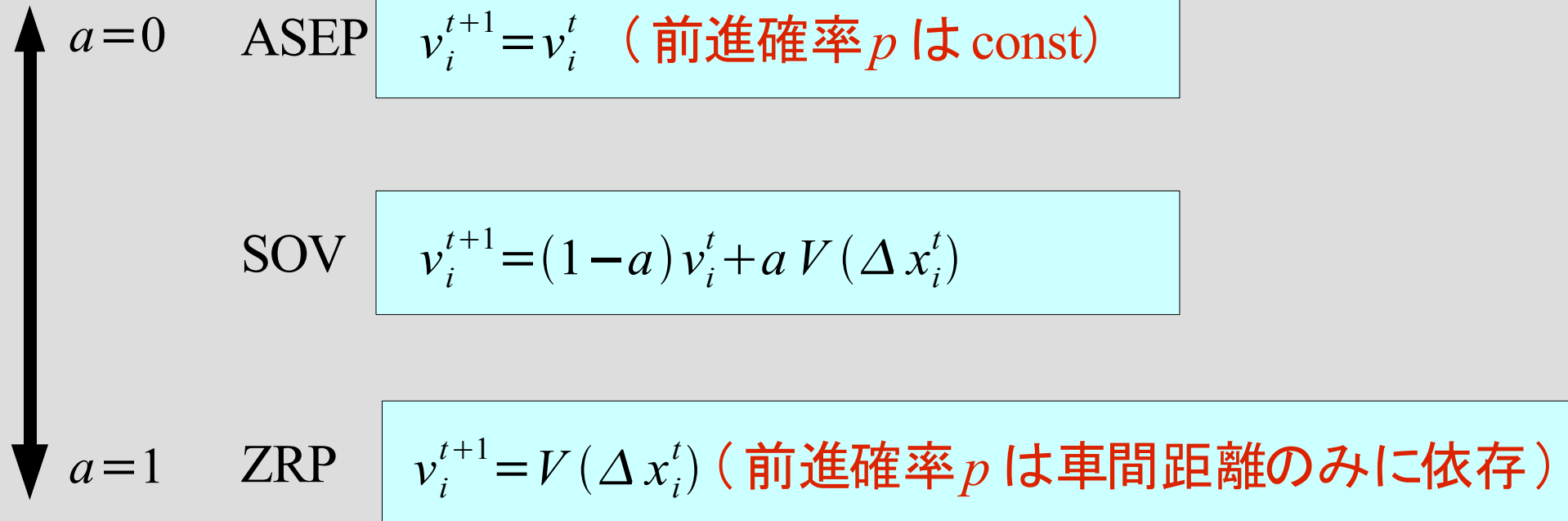
$$x_i^{t+1} = x_i^t + 1 \text{ with } p = v_i^{t+1}$$

$$\Delta x_i^t = x_{i+1}^t - x_i^t - 1$$

交通流モデル 6/6

- Stochastic Optimal Velocity(SOV) モデル (続き)

- SOV と ASEP および ZRP の関係



Masahiro Kanai, Katsuhiro Nishinari and Tetsuji Tokihiro, "Stochastic optimal velocity model and its long-lived metastability", *Phys. Rev. E*, 72, 035102(2005)

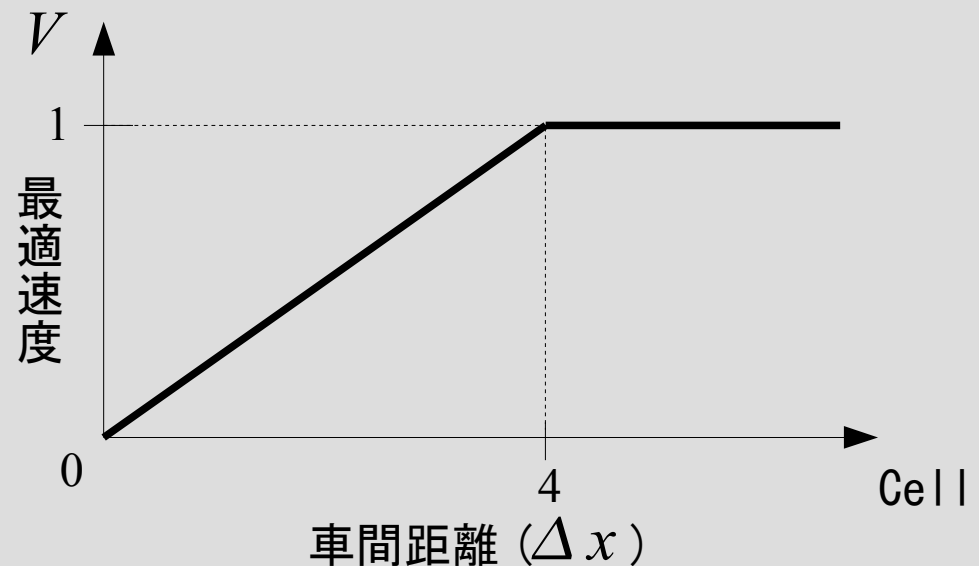
実世界とシミュレーションの対応

例：高速道路

	実世界	シミュレーション
最高速度	100km/h	1cell/step(p=1の時)
距離	7.5m	1cell
時間	0.27sec = (7.5m)/(100km/h)	1step

•Optimal Velocity Function

$$V(\Delta x) = \begin{cases} x/4 (x \leq 4) \\ 1 (x \geq 4) \end{cases}$$

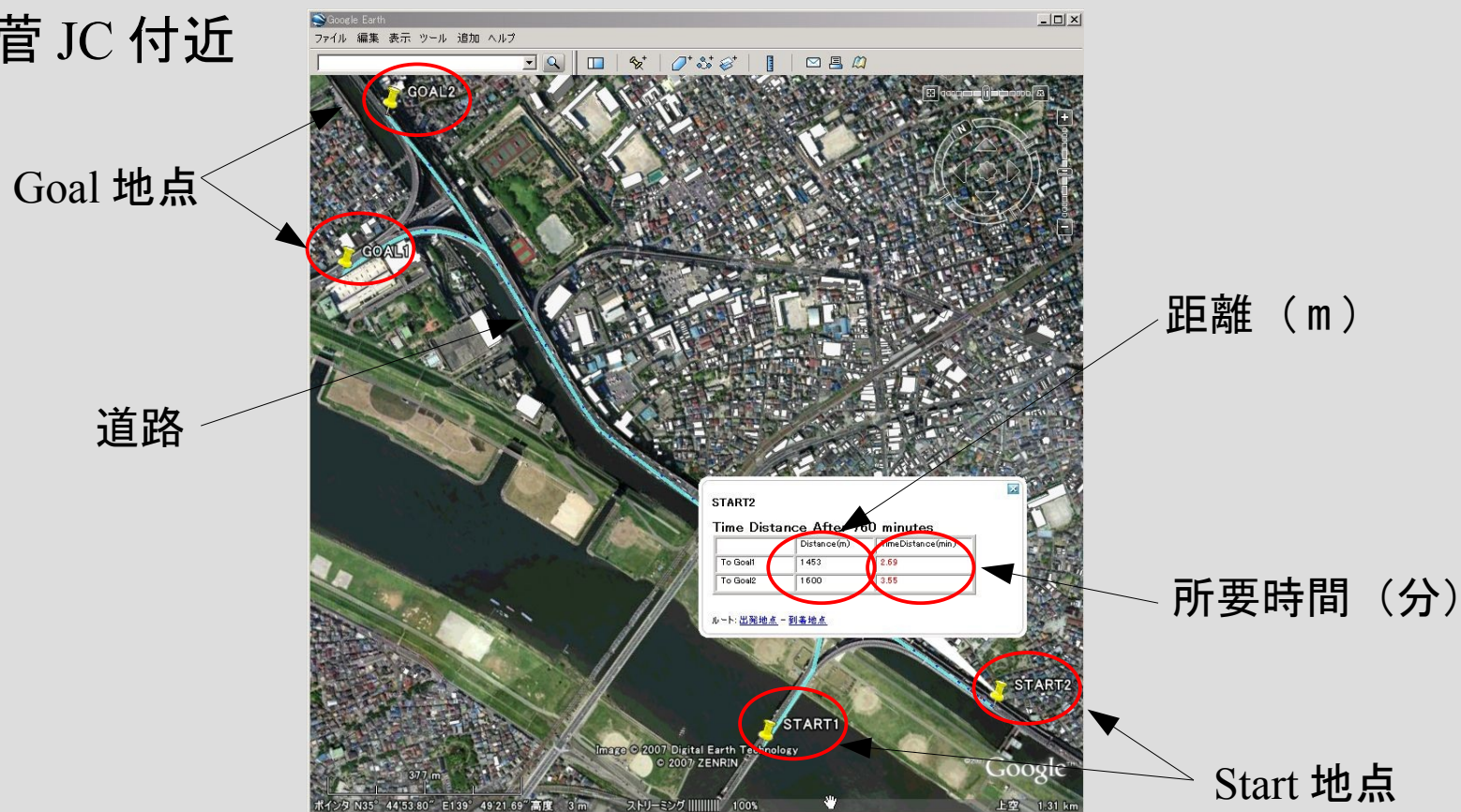


システム概要 1/3

目的

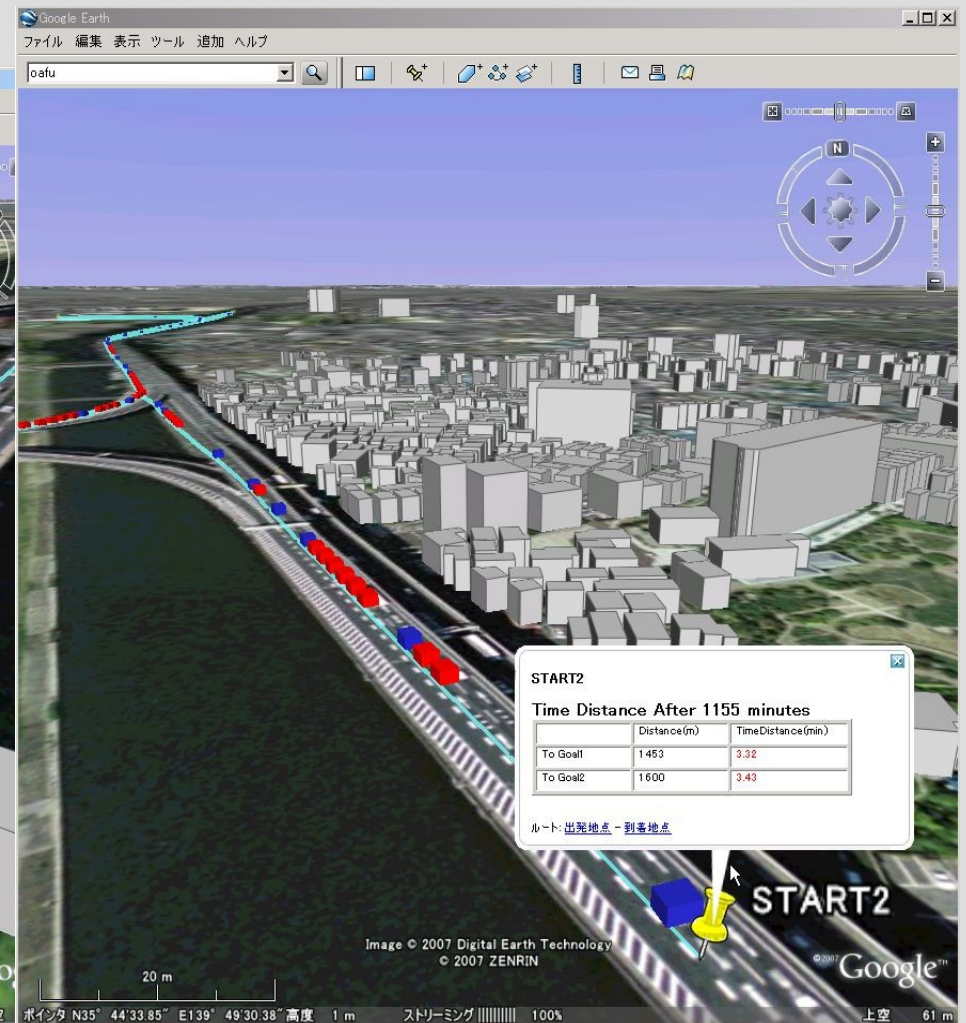
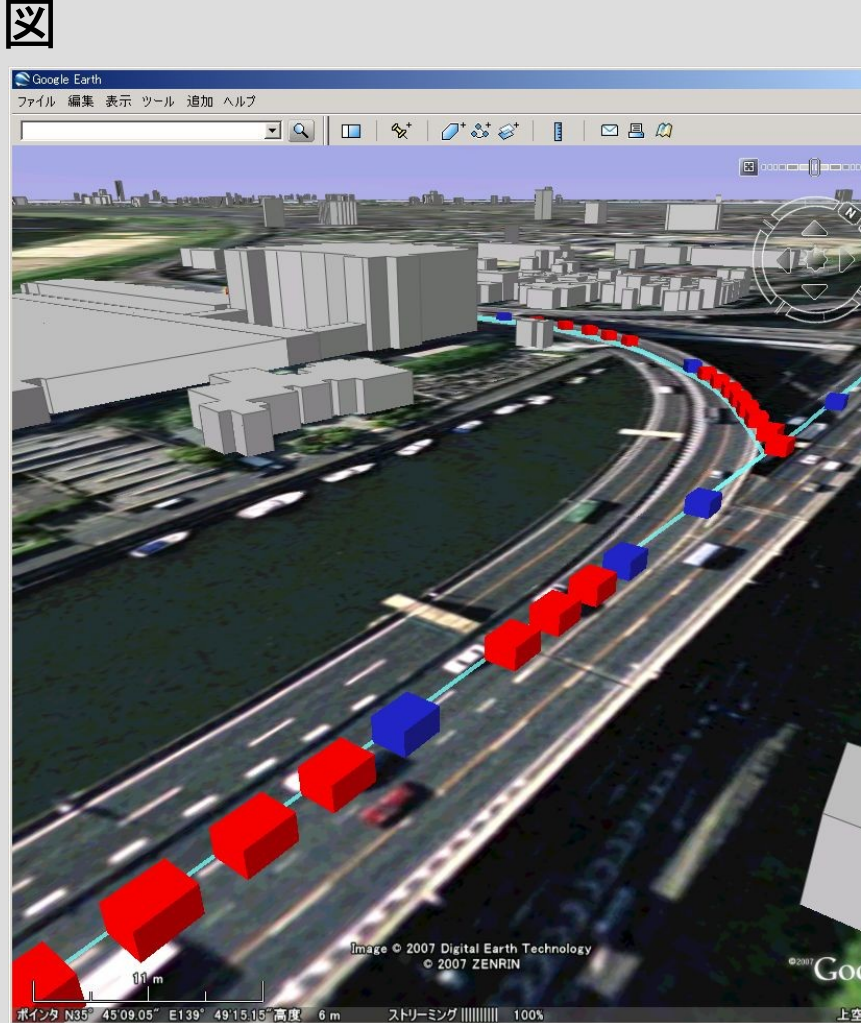
GoogleEarth 上で SOV モデルに従った交通流シミュレーションを行うことで、交通渋滞を予測し任意の二点間の所要時間を推定する。

●例：小菅 JC 付近



システム概要 2/3

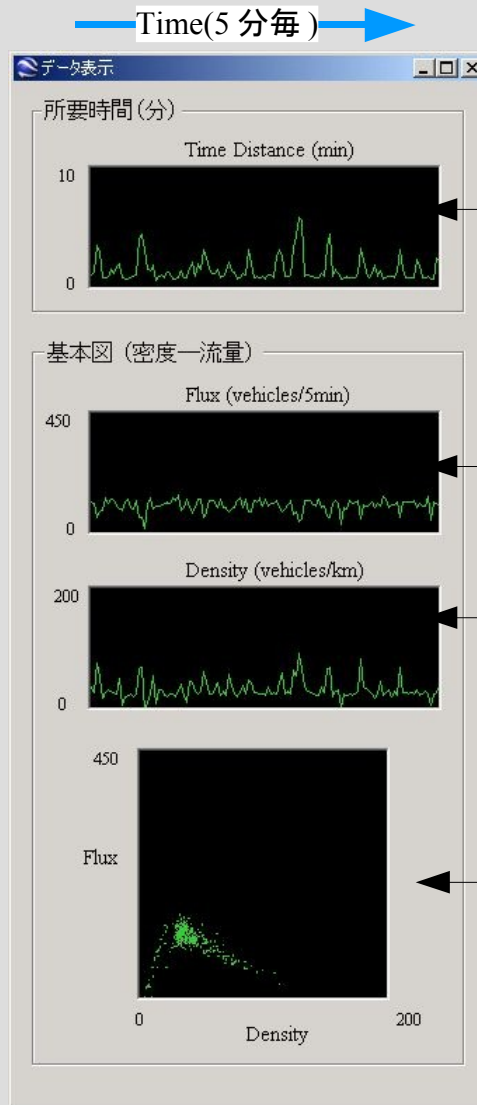
拡大図



■ 渋滞中車両

システム概要 3/3

データ表示



推定所要時間 (分)

流量 (vehicles/5min) : 5分間にある地点を通過した車の台数

密度 (vehicles/km) : 道路上のある区間1kmの間に存在する車の台数

基本図 (密度-流量)

処理概要 1/5

ルート情報
(kml 形式)

交通情報
(初期・境界条件)

車両表示部



GoogleEarth



計算部

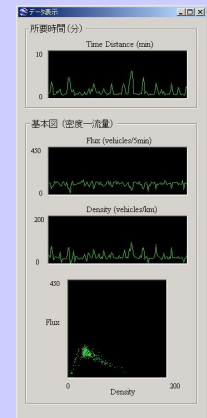
経路 Cell 分割, 距離計算

車両位置計算 (SOV)

所要時間推定

車両情報 kml 変換

データ表示部



処理概要 2/5

入力：ルート情報

•kml の具体例

詳しくは http://code.google.com/apis/kml/documentation/kml_tags_21.html

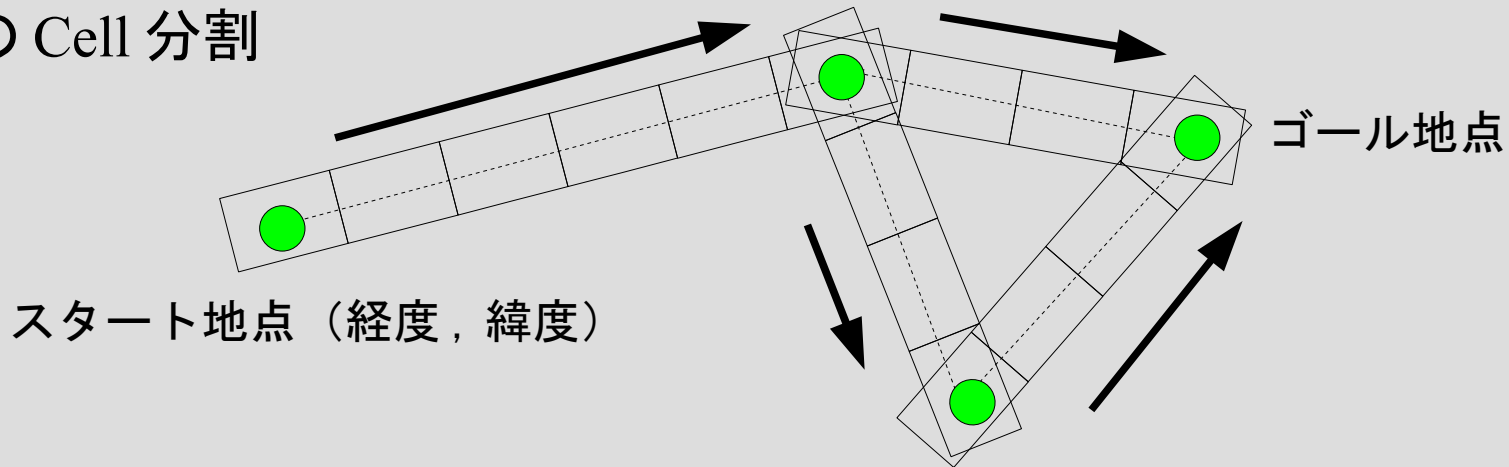
```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.1">
<Document>
<name>Route.kml</name>
<Folder>
<Placemark>
<name>Route1</name>
<Style>
<LineStyle>
<color>c8ffff7f</color>
<width>5</width>
</LineStyle>
</Style>
<LineString>
<extrude>1</extrude>
<tessellate>1</tessellate>
<coordinates>
139. 820528286111, 35. 7431262305556, 0 139. 82065615
4639, 35. 7431744022247, 0
...
</coordinates>
</LineString>
</Placemark>
...
```

経路は点（経度・緯度・標高）
の羅列で表現

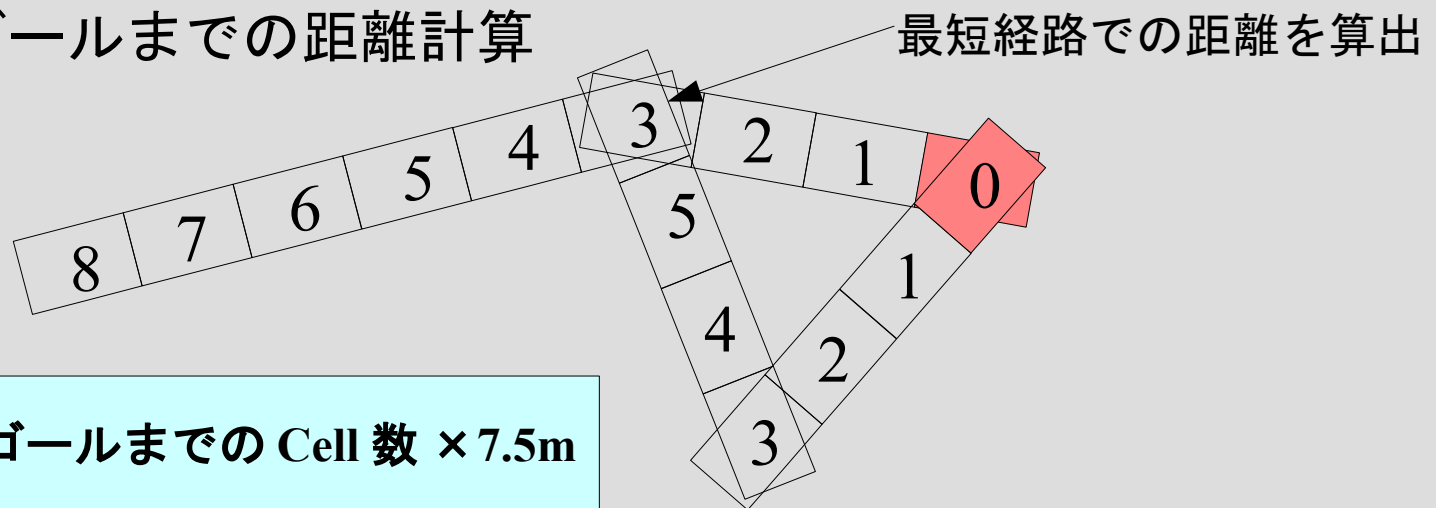
処理概要 3/5

道路 Cell 分割, 距離計算

経路の Cell 分割



各 Cell からゴールまでの距離計算



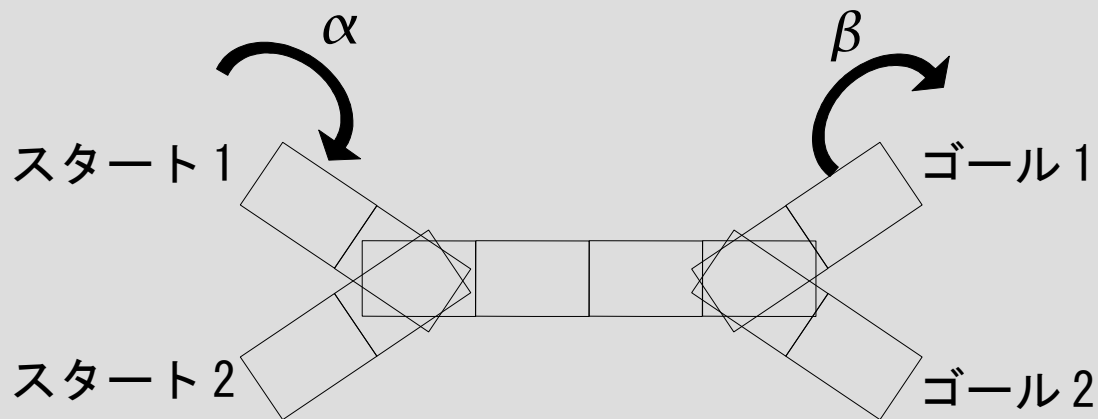
実際の距離 = ゴールまでの Cell 数 × 7.5m

処理概要 4/5

車両位置計算 (SOV)

- 決めなければならない初期・境界条件

- ⇒ 初期の車両の配置
- ⇒ スタート地点毎の α , およびゴール地点毎の β
- ⇒ 各車両の目的地 (ゴール) の割合



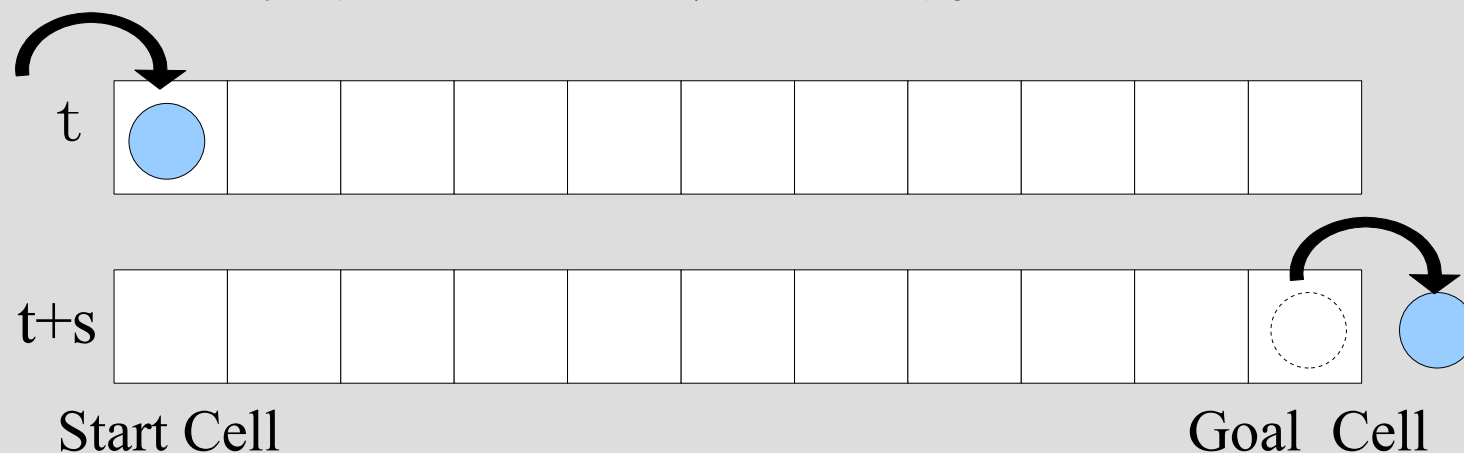
α スタート Cell が空の場合, 新しく車が進入してくる確率

β ゴール Cell に車が存在する場合, その車が退出する確率

処理概要 5/5

所要時間推定

時刻 t の所要時間 : 時刻 t にスタート地点に進入した車がゴール地点から退出するまでに要するステップ数 $\times 0.27$ (秒) .



$$\text{所要時間 (t)} = s \times 0.27 \text{ 秒}$$

ただし推定時間は、時刻 t の前後 5 分間にスタート地点を出発した車の所要時間の平均値とする。

正確な予測へ向けての課題 1/2

初期・境界条件

- 現時点で入手可能な情報にはどのようなものがあるか？
- 今**スタート地点にいる車がゴールに到達するまでの時間を**今**予測する
(今ゴールに到達した車が実際に要した時間ではない)
⇒ **未来の予測には未来の情報 (境界条件) が必要.**
- α , β とは何に対応するか？
 - α : スタート Cell が空の場合, 新しく車が進入してくる確率
⇒ ? (正確には不明)
⇒ **スタート地点の流入率 (単位時間あたりの進入車の割合) で近似.**
 - β : ゴール Cell に車が存在する場合, その車が退出する確率
⇒ **ゴール地点を通過する車の速度.**
- 過去の履歴から, 未来の α , β をどのように推定するか？

正確な予測へ向けての課題 2/2

ドライバーの行動

- 複数のゴールや分岐が存在する場合，各車両の行き先の割合はどのように推定するのか？
- ゴールまでの経路が複数存在する場合，ドライバーは最短経路をたどるのか？