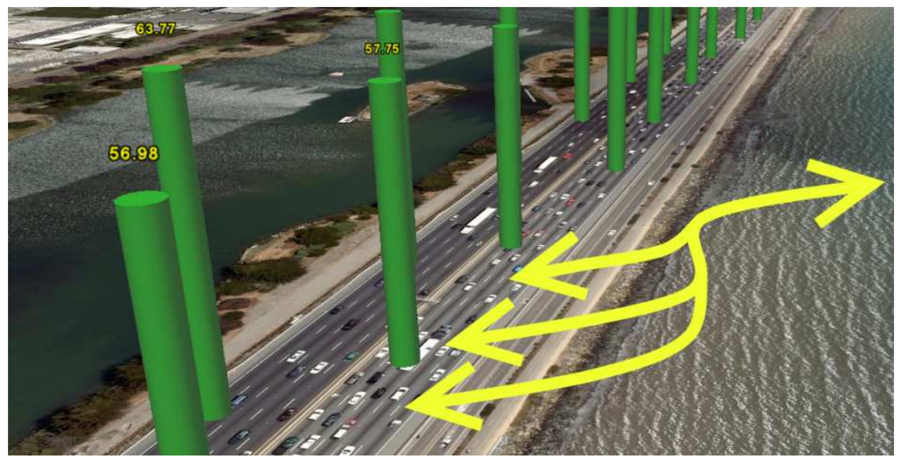
Optimization formulations for inverse problems with applications to Mobile Sensing



Christian Claudel

UC Berkeley

Department of Electrical Engineering and Computer Sciences

Los Alamos, 03/23//2010

Mobile sensing for solving societal challenges

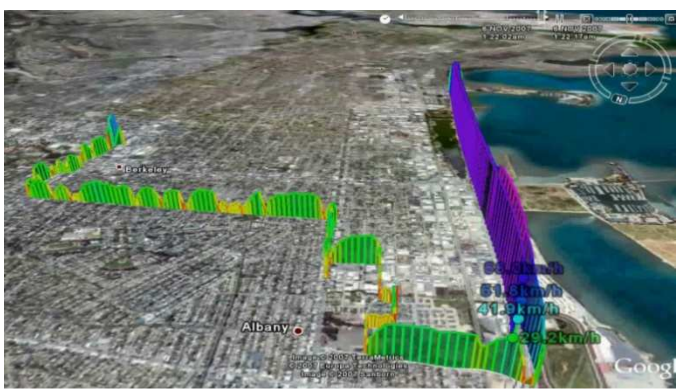
Fusion of mobile pollution data with models, for air pollution monitoring

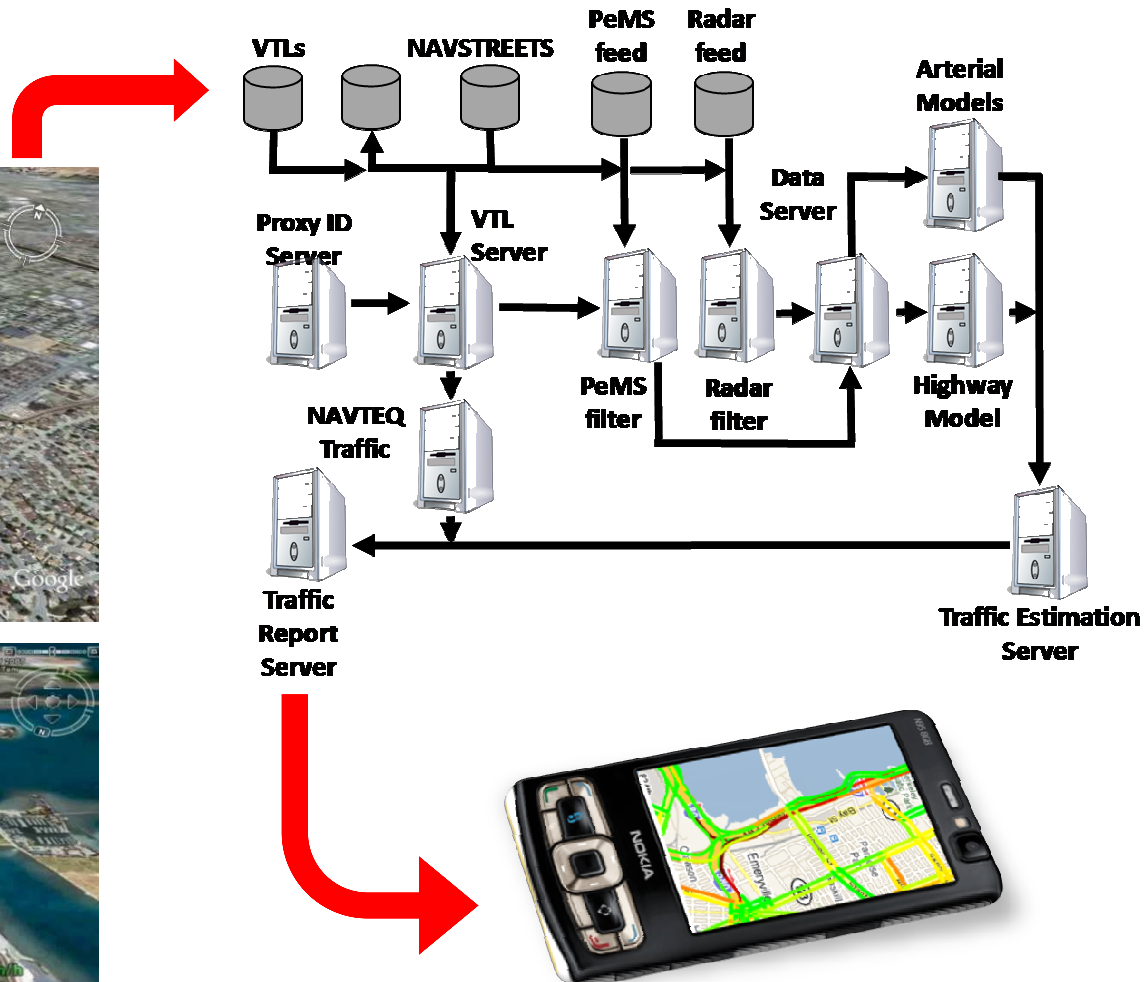
source: Seto UC Berkeley

source: DHS/NASA

Mobile sensing for solving societal challenges

Participatory sensing for traffic flow monitoring: fusion of fixed and mobile data into traffic models





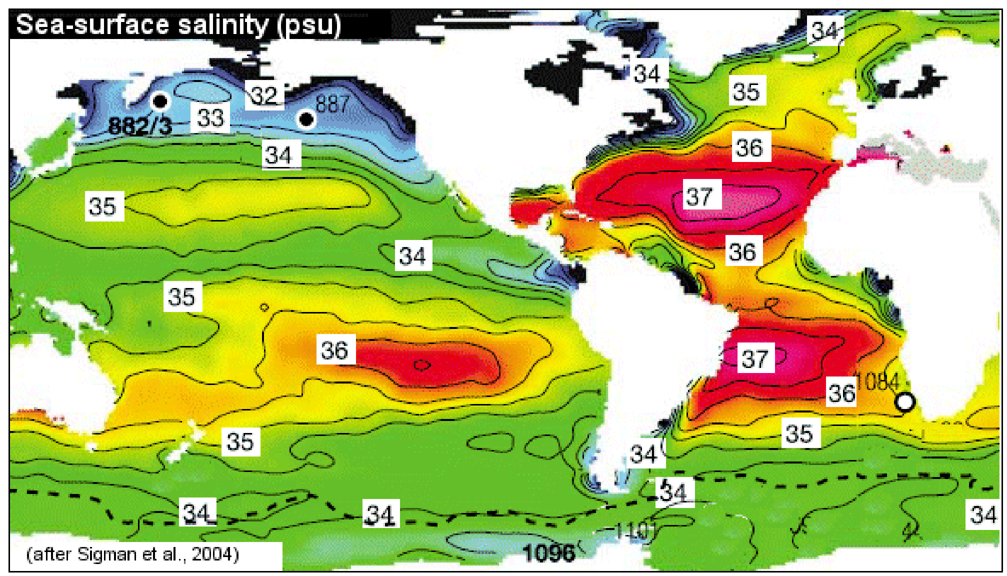
Mobile sensing for solving societal challenges

Impact of desalination on water ecosystems: how can we assess it?

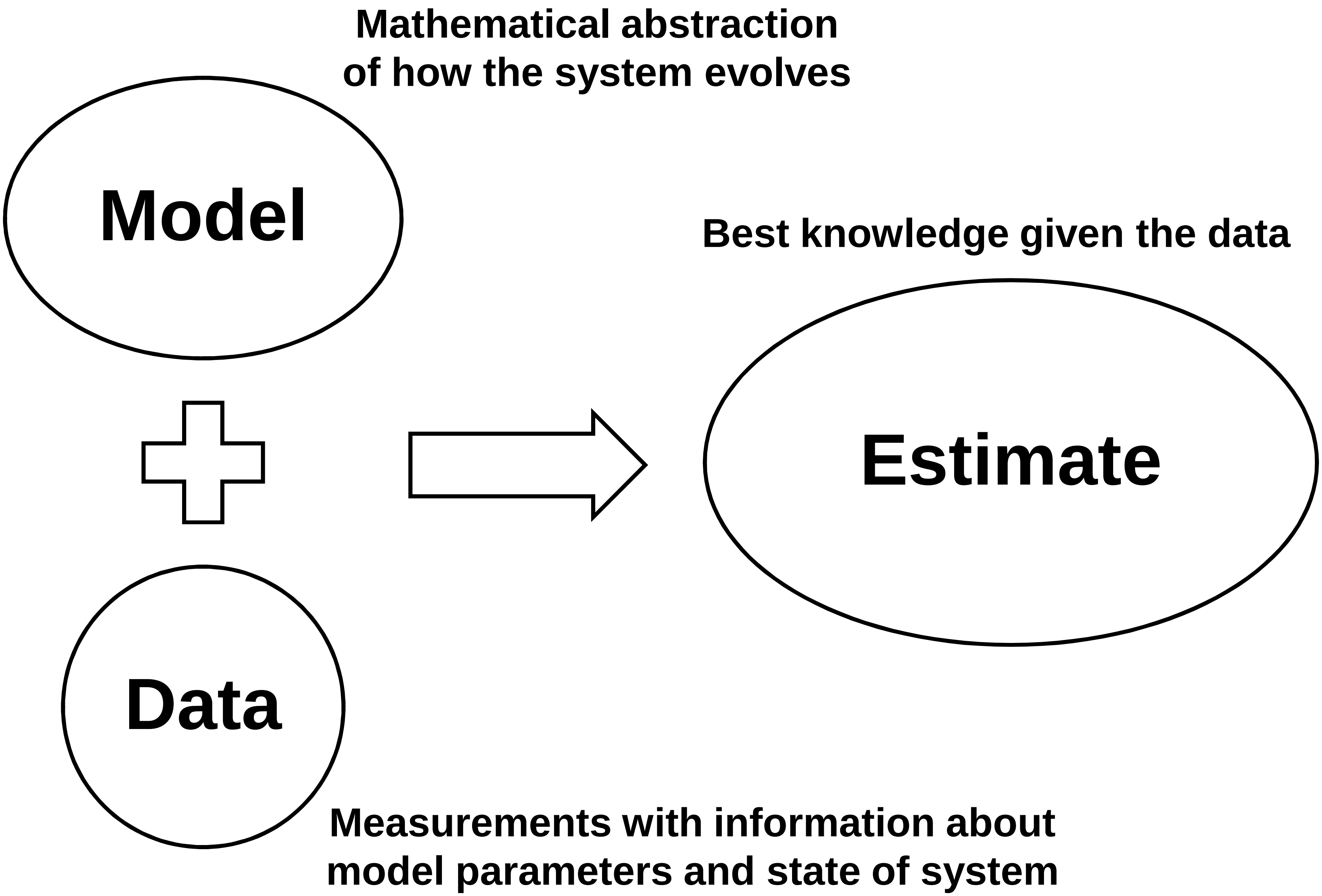
World's largest desalt plant opened

Siraj Wahab | Arab News

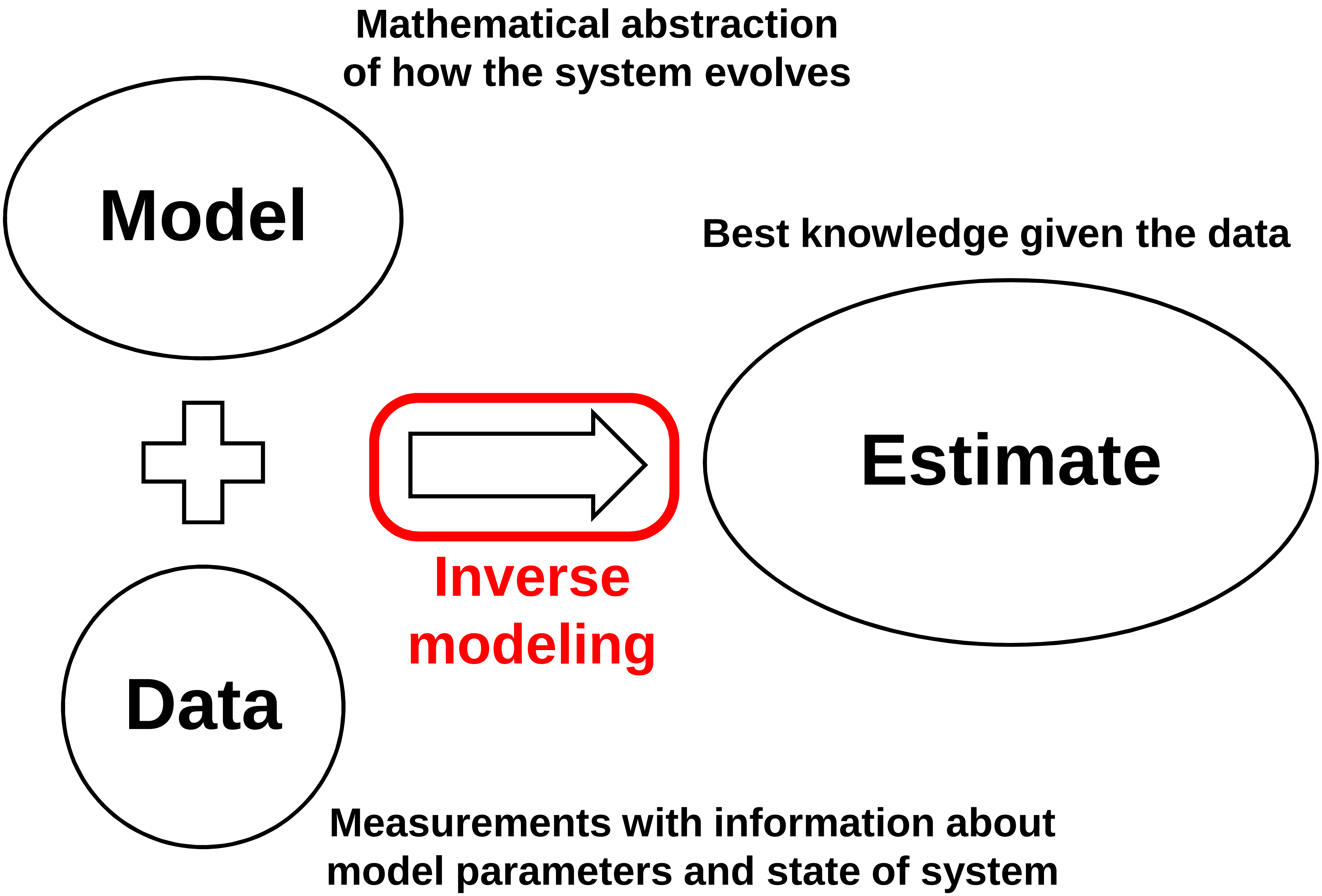
JUBAIL: Custodian of the Two Holy Mosques King Abdullah yesterday launched massive development projects worth SR54 billion in the Eastern Province's newest industrial zone called Jubail-II. The new city is



Overview


Mathematical abstraction
of how the system evolves
Model
Best knowledge given the data
Estimate
Data
Measurements with information about
model parameters and state of system

Overview


Mathematical abstraction
of how the system evolves
Model
Best knowledge given the data
Estimate
Inverse
modeling
Data
Measurements with information about
model parameters and state of system

Outline

Inverse modeling problems

- **Problem description**
- **Introductory example**

Inverse modeling problems involving Hamilton-Jacobi equations

- **Background**
- **Problem definition**
- **Optimization formulations for inverse modeling**
- **Parameter uncertainty**

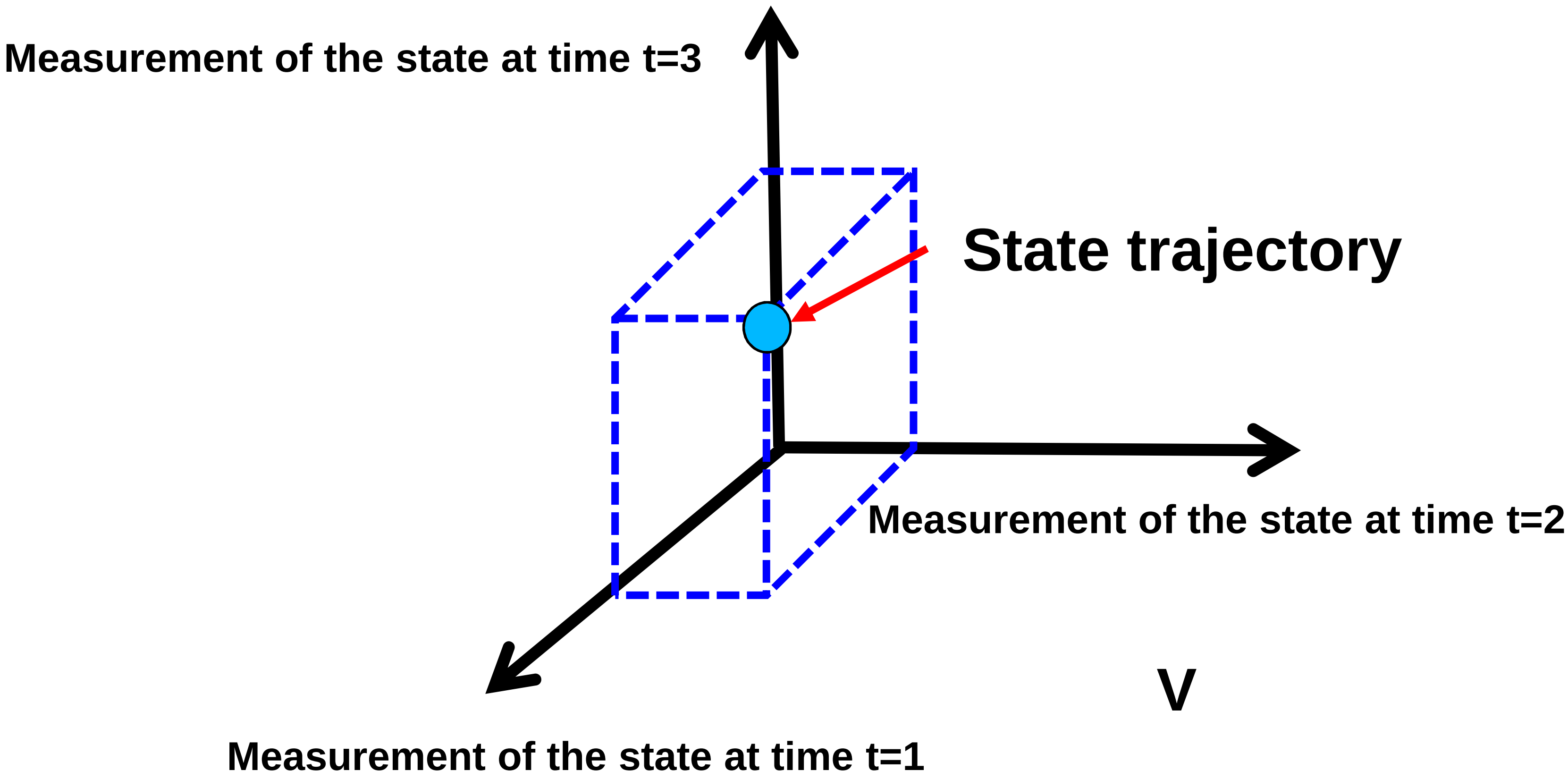
Sensing applications in traffic flow

- **Fault detection in the PeMS sensor network**
- **Other applications: racing with Google Maps**

Conclusion

Inverse modeling problems

Assume that any evolution trajectory of a system can be represented as a point in a (high dimensional) vector space V

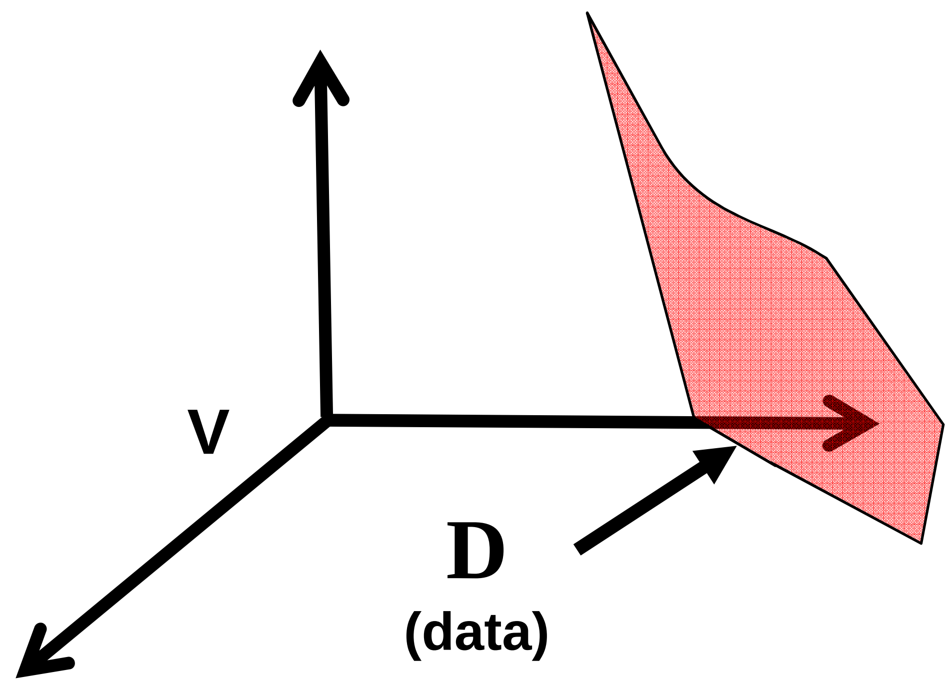


Inverse modeling problems

The measurement data corresponds to a set of "possible state trajectories" D of the system (i.e. trajectories that are compatible with the measurement data)

D is not a single point because of:

- **Measurement uncertainty**
- **Missing measurements**

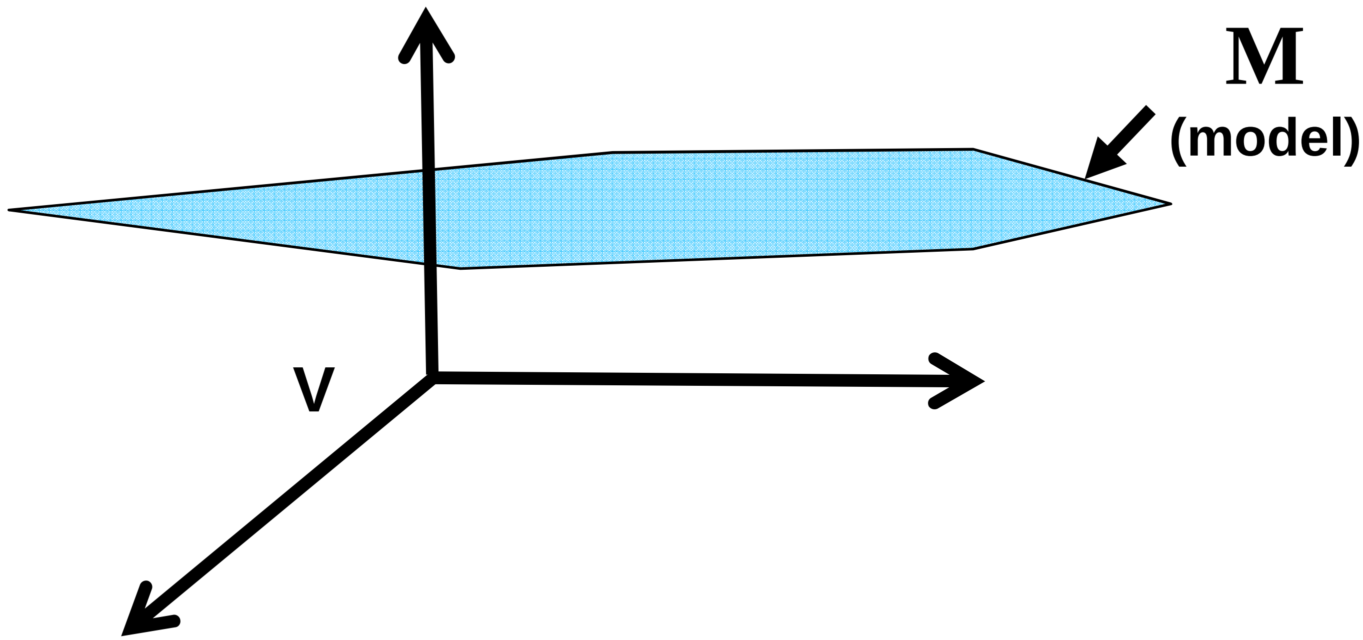


Inverse modeling problems

The model yields a set of "possible model trajectories" M (i.e. trajectories that are compatible with the model)

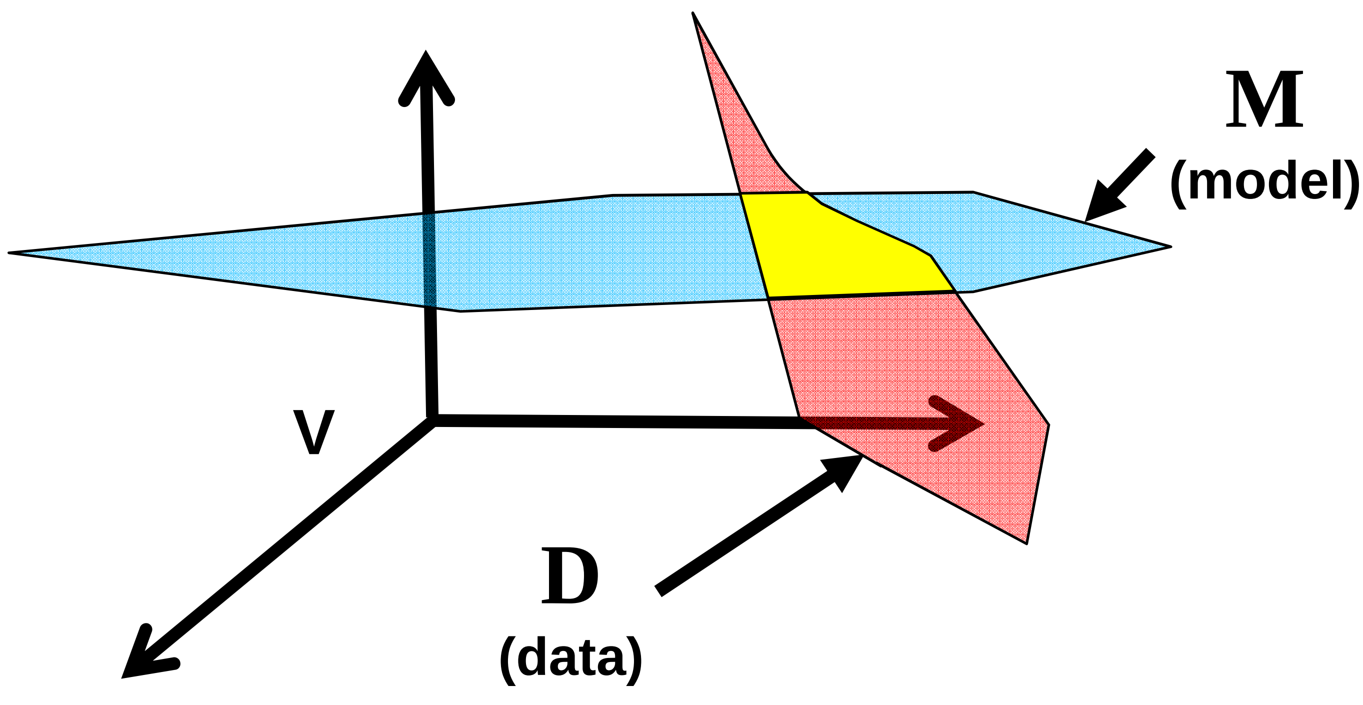
M is not a single point since:

- **The evolution trajectory depends upon the initial condition**
- **The model parameters may be uncertain**



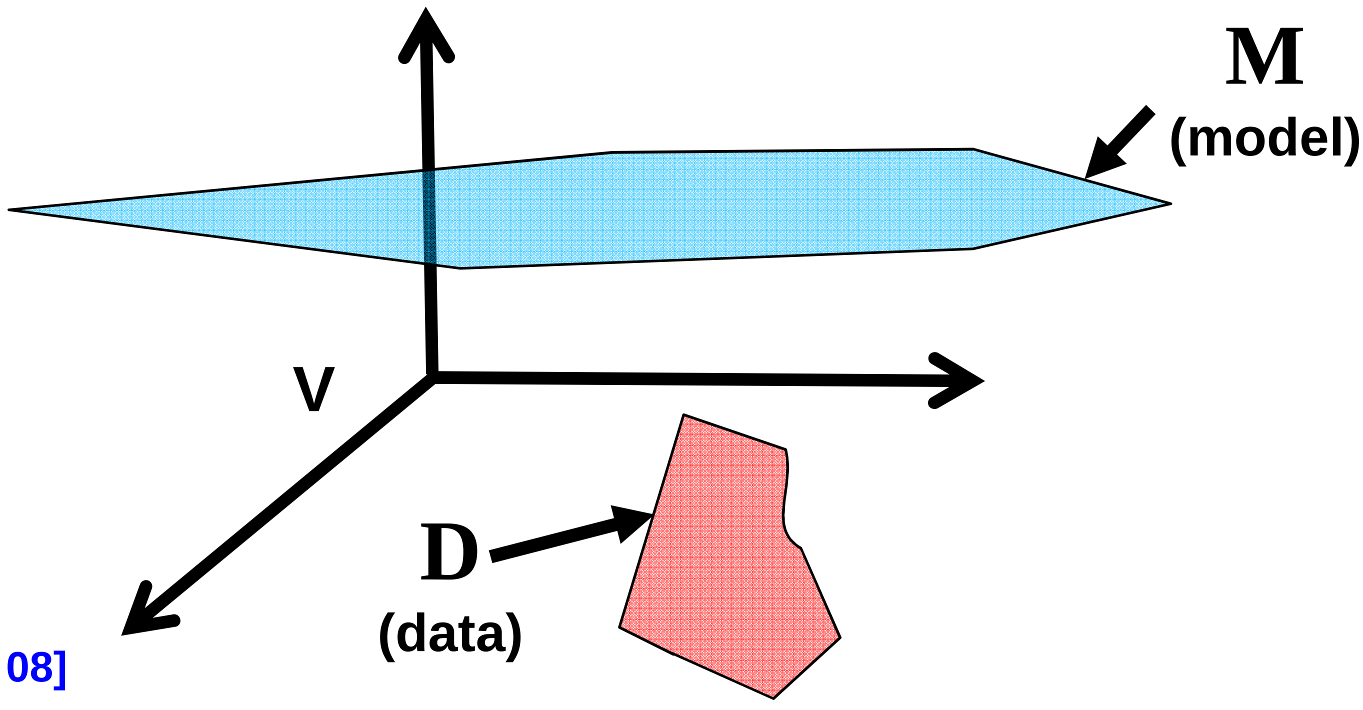
Case 1: compatibility between data and model

If both sets intersect, there exist state trajectories which are both compatible with the model and the data



Case 2: incompatibility between data and model

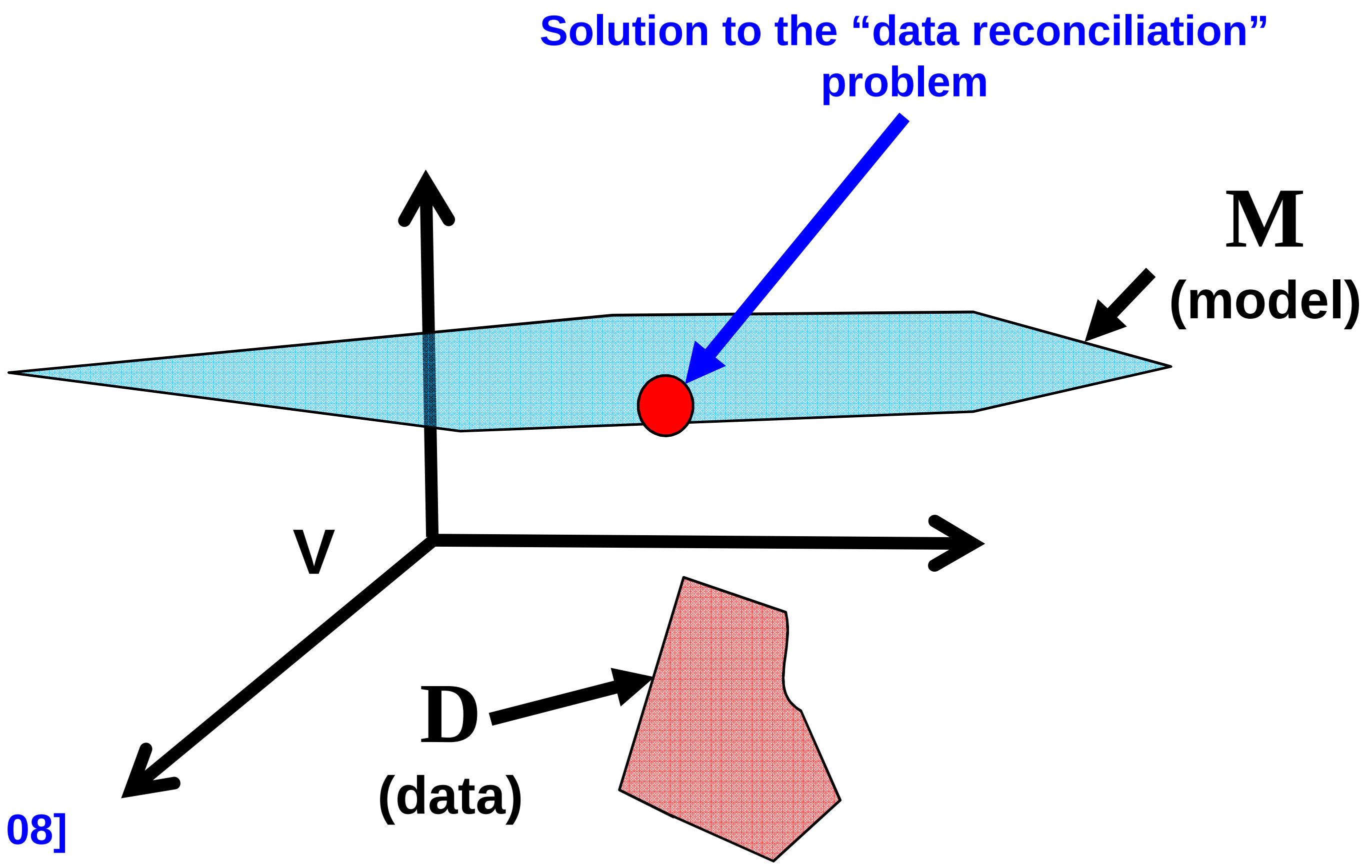
If both sets do not intersect, there are no state trajectories which are both compatible with the model and the data. Two problems might still be of interest



[Evensen 07]
[Wu, Litrico, Bayen 08]

Case 2.a: incompatibility between data and model

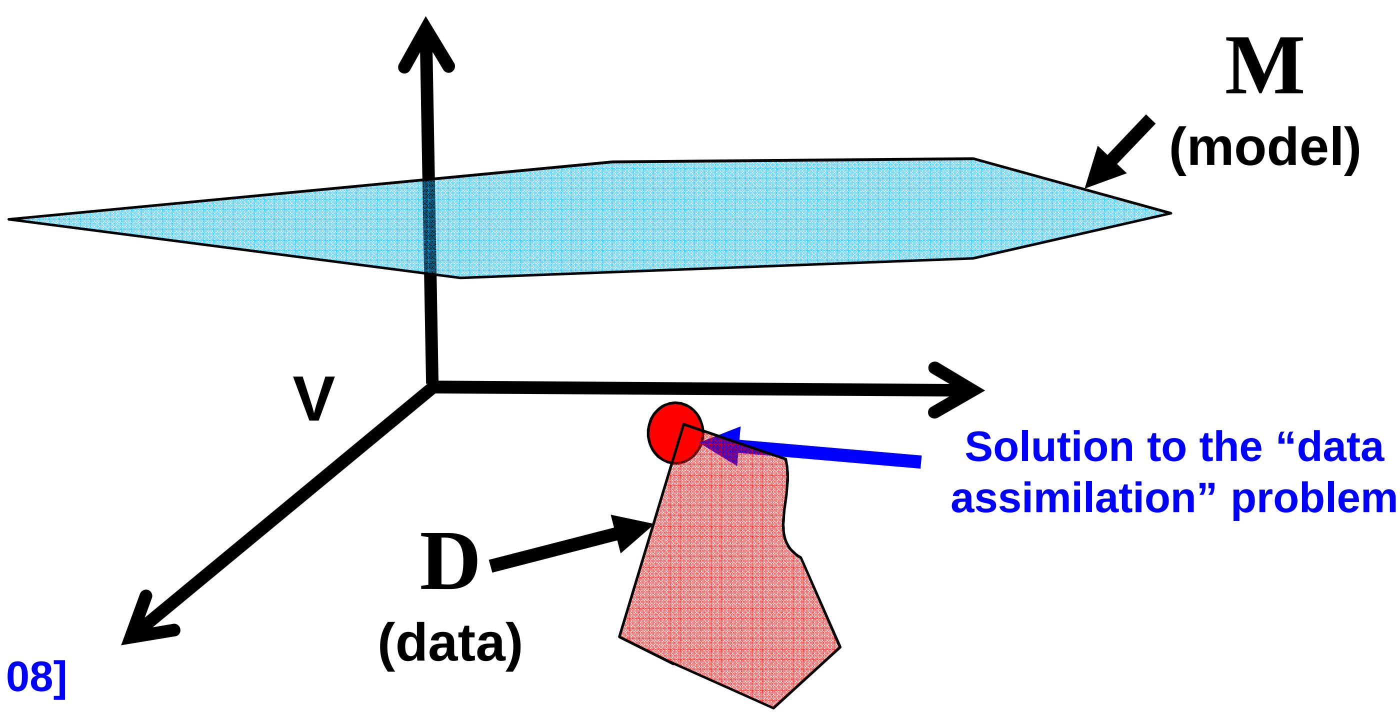
Data reconciliation problem: finding the trajectory satisfying the model closest to the data



[Evensen 07]
[Wu, Litrico, Bayen 08]

Case 2.b: incompatibility between data and model

Data assimilation problem: **finding the trajectory satisfying the data closest to be a solution of the model**

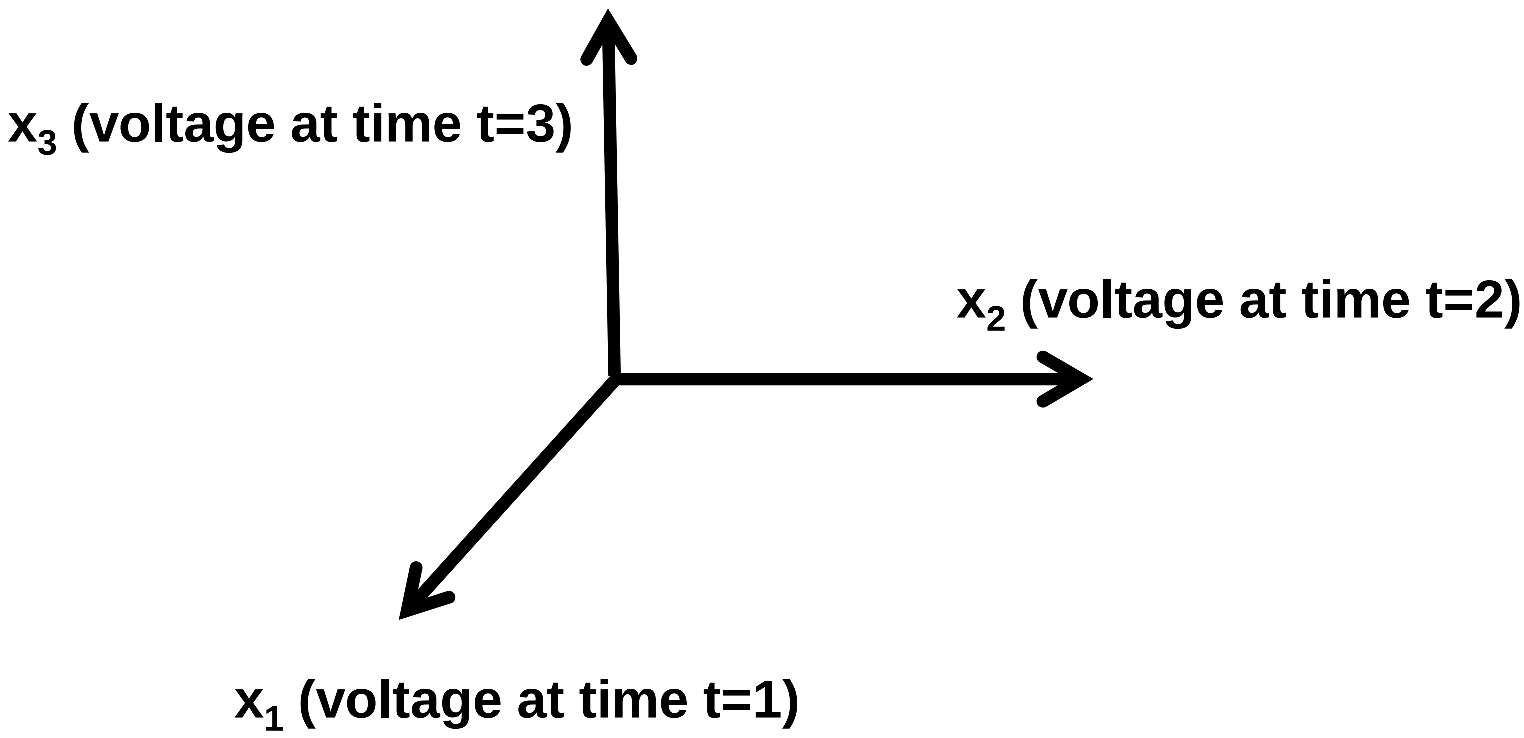


[Evensen 07]
[Wu, Litrico, Bayen 08]

Introductory example

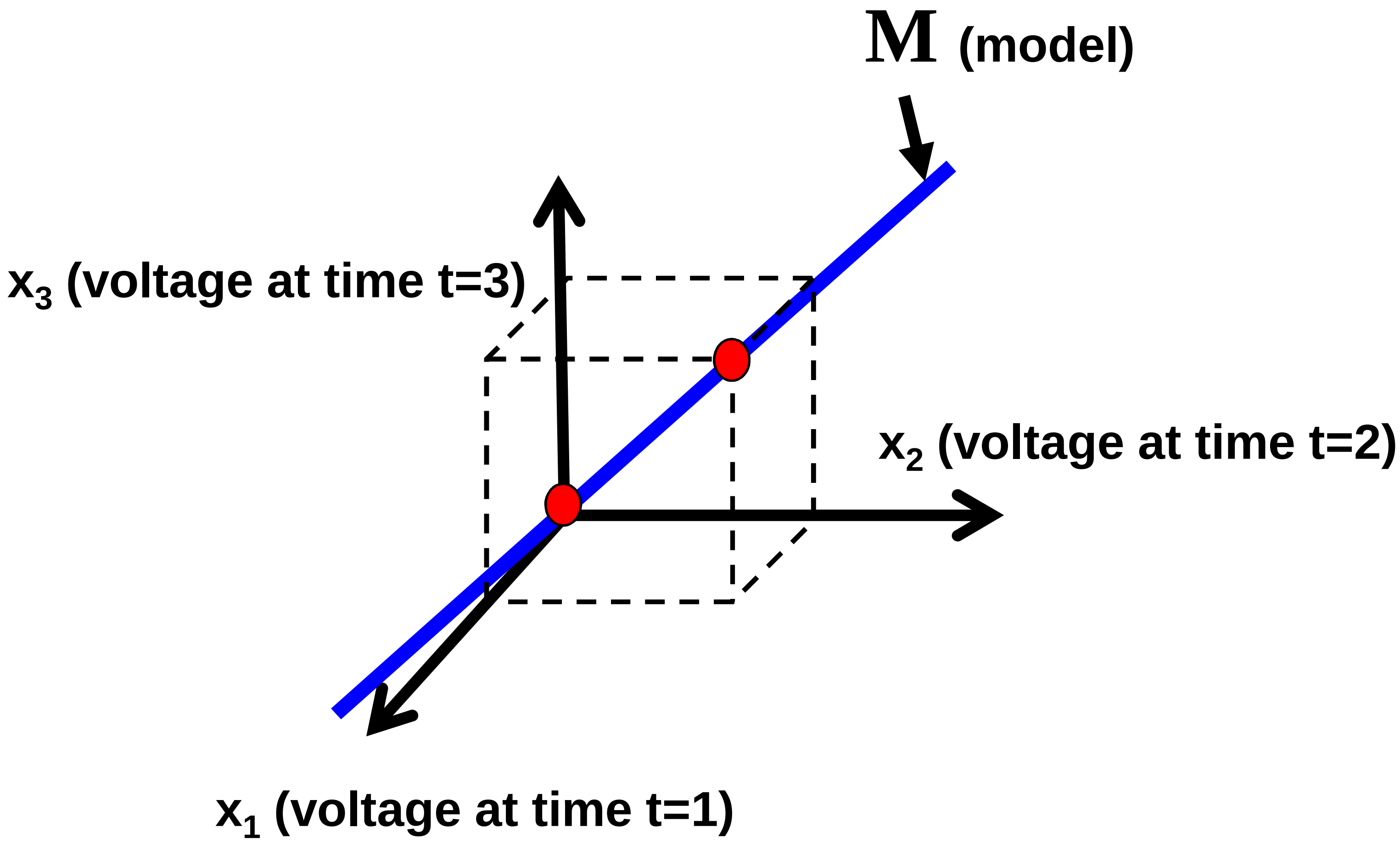
Consider the measurement of a battery's voltage x over (discrete) time. If n discrete measurements are considered, the evolution trajectory space V is R^n

For n=3, the state trajectory is defined by (x_1,x_2,x_3)



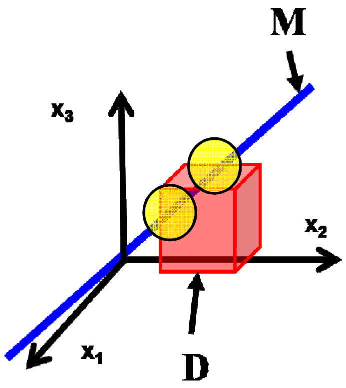
Introductory example

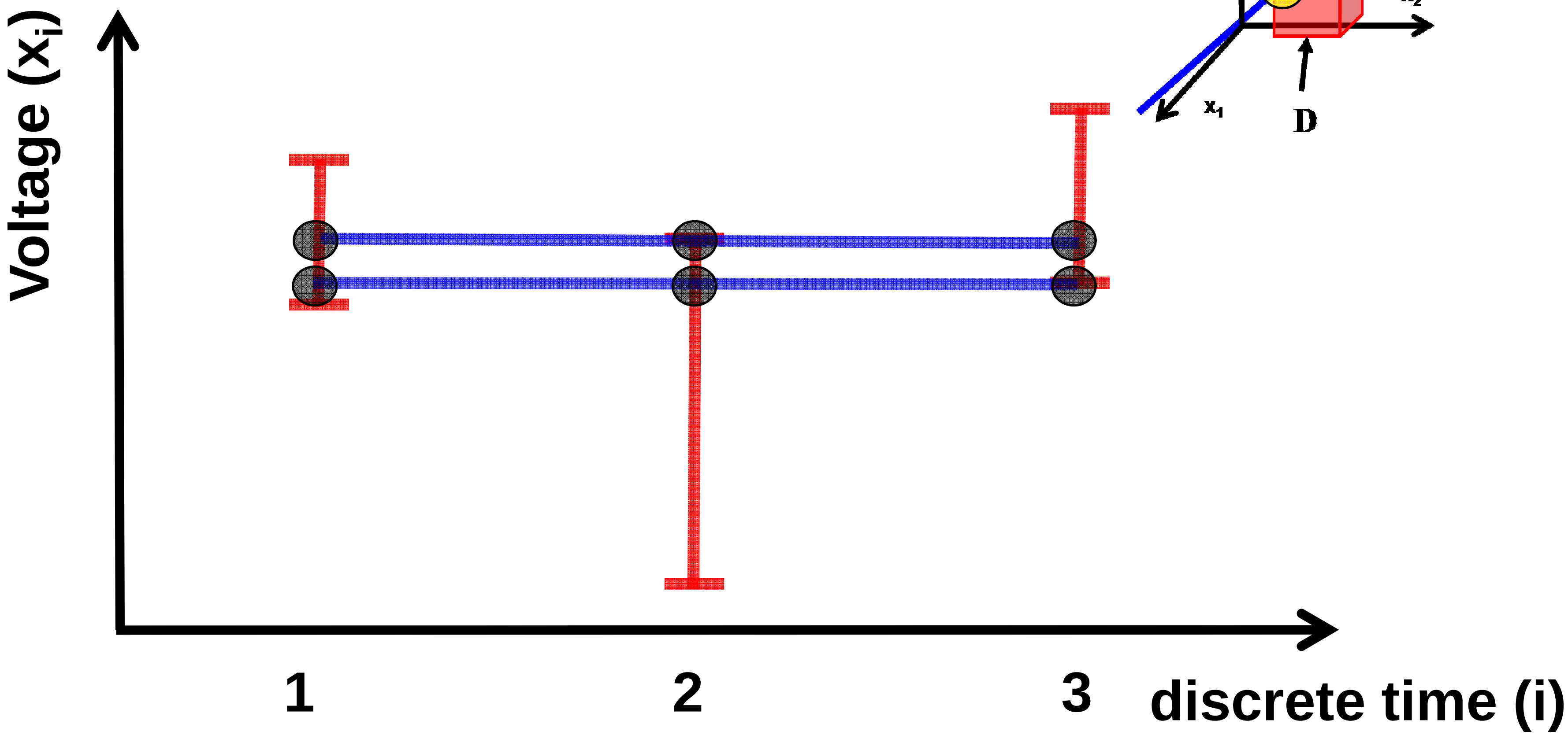
Consider the simple discrete time linear model: $x_{k+1} = x_k$



Case 1: compatible data

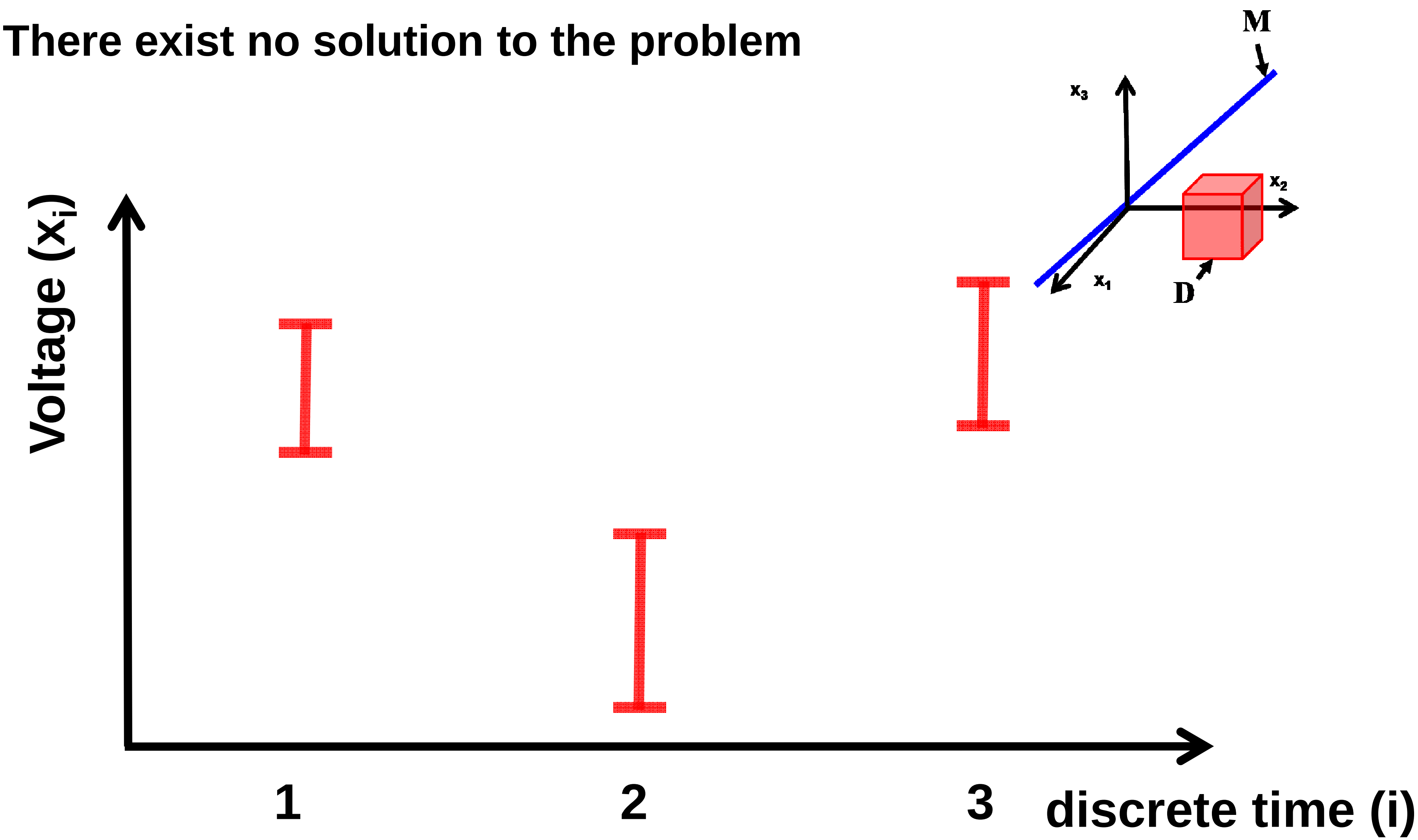
There can exist more than one solution to the problem



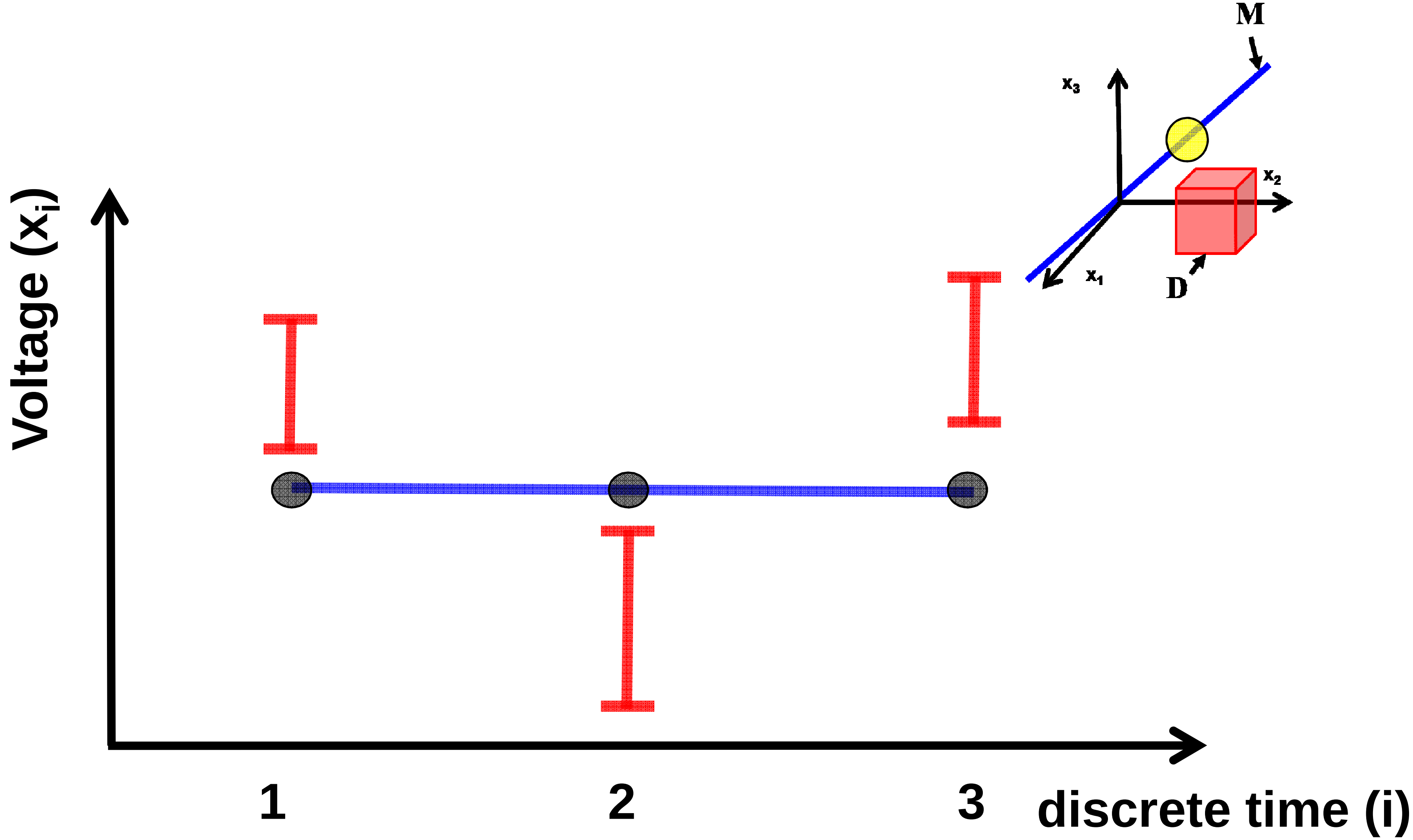


Case 2: incompatible data

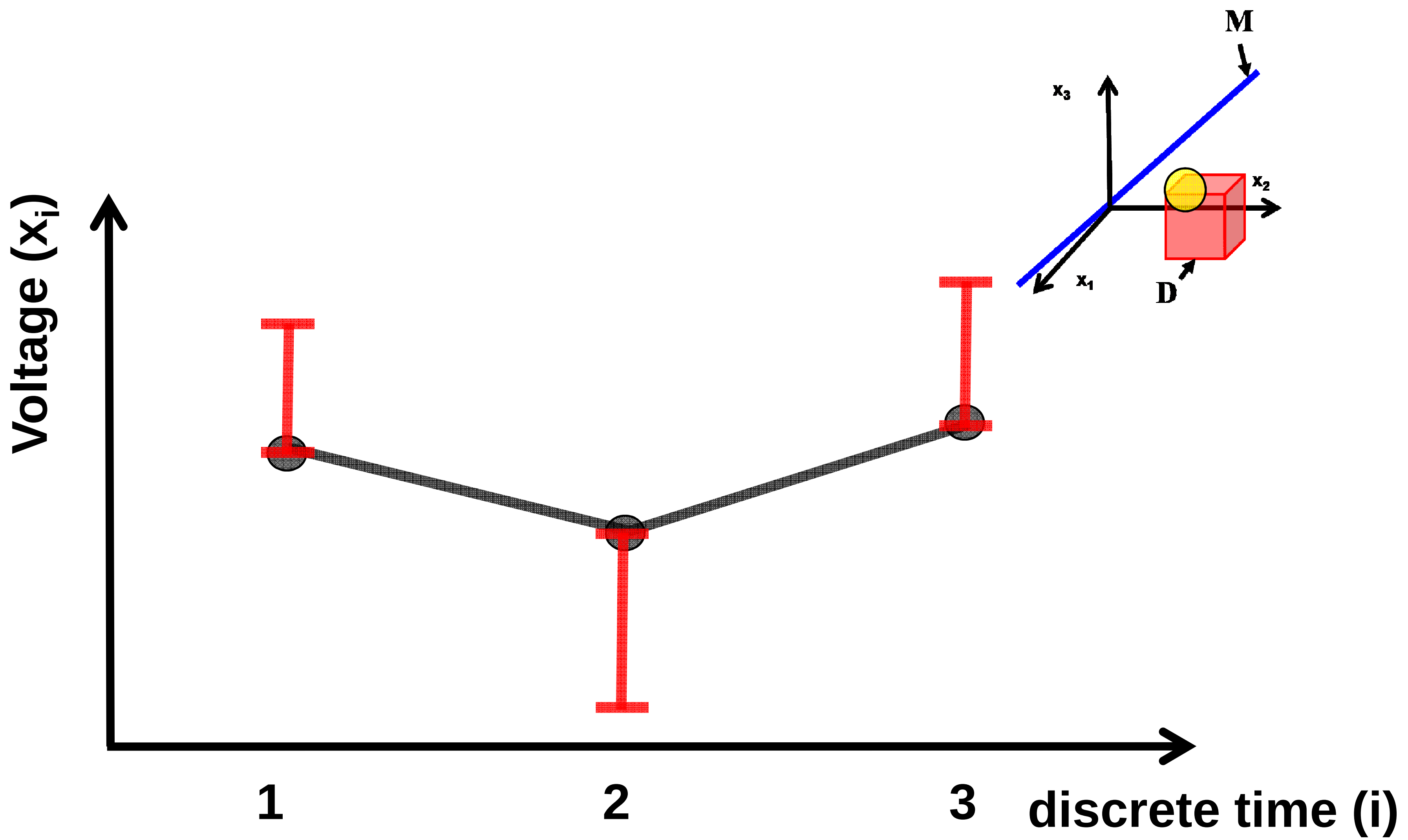
There exist no solution to the problem



Case 2.a: data reconciliation



Case 2.b: data assimilation



Outline

Inverse modeling problems

- **Problem description**
- **Introductory example**

Inverse modeling problems involving Hamilton-Jacobi equations

- **Background**
- **Problem definition**
- **Optimization formulations for inverse modeling**
- **Parameter uncertainty**

Sensing applications in traffic flow

- **Fault detection in the PeMS sensor network**
- **Other applications: racing with Google Maps**

Conclusion

Context of this work: Mobile Millennium project

Mobile Millennium project: estimating traffic conditions on highways and secondary roads using multiple sources of data:

Dedicated infrastructure:

- **Transponders (FasTrak)**
- **Speed radars**
- **Magnetometers**
- **Loop detectors**
- **Traffic cameras**

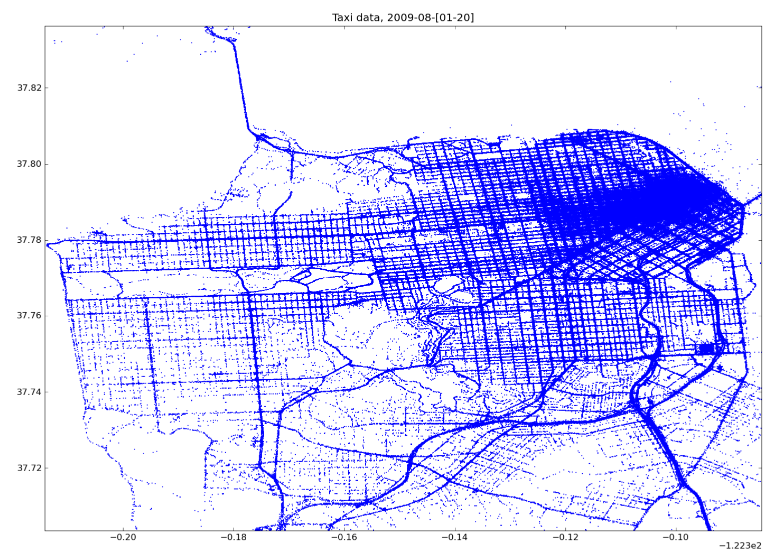
http://traffic.berkeley.edu

Context of this work: Mobile Millennium project

Mobile Millennium project: estimating traffic conditions on highways and secondary roads using multiple sources of data:

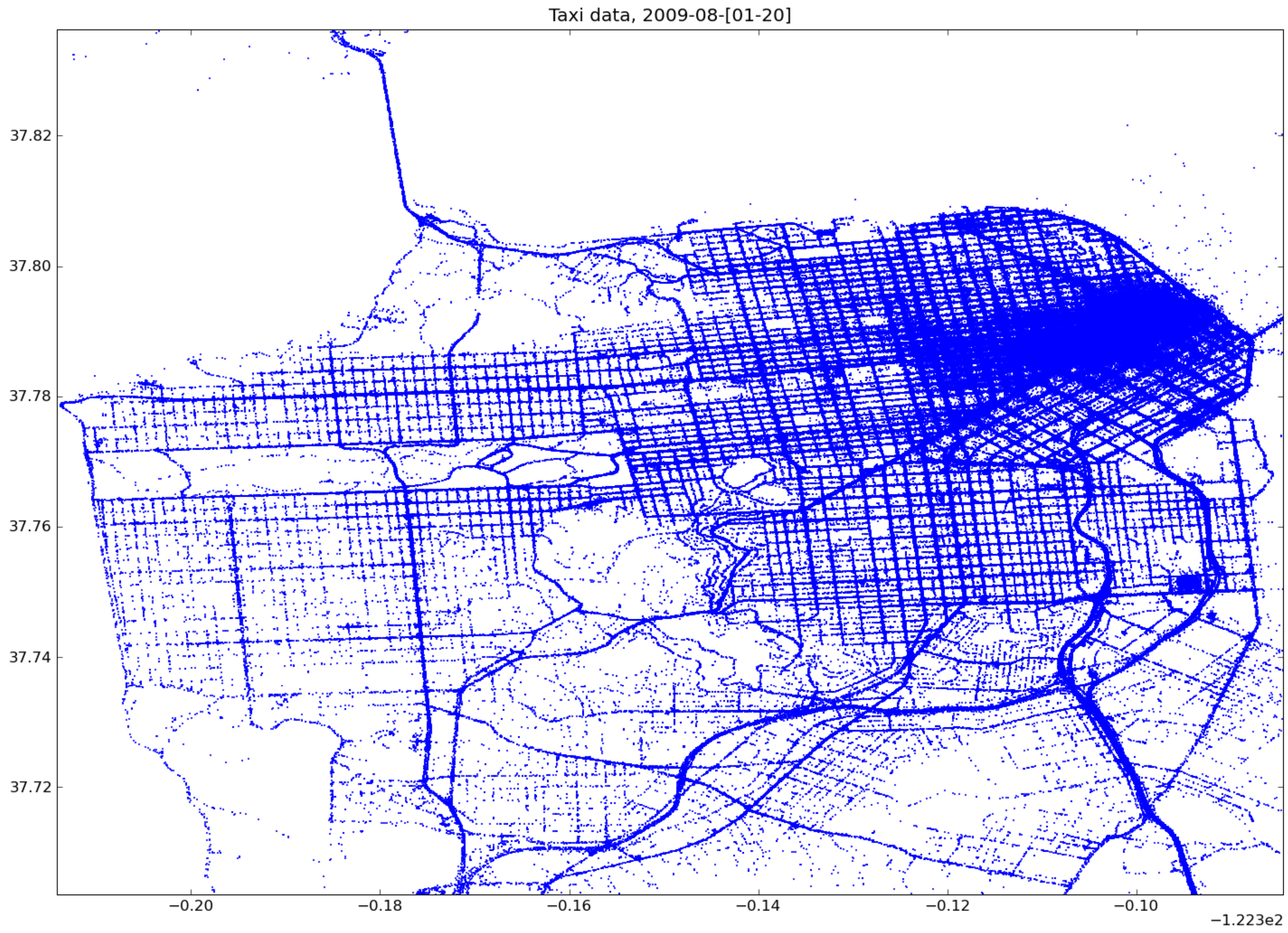
Participatory sensing

- **Phones**
- **Taxi location information**
- **Aftermarket devices**
- **Fleet location information**

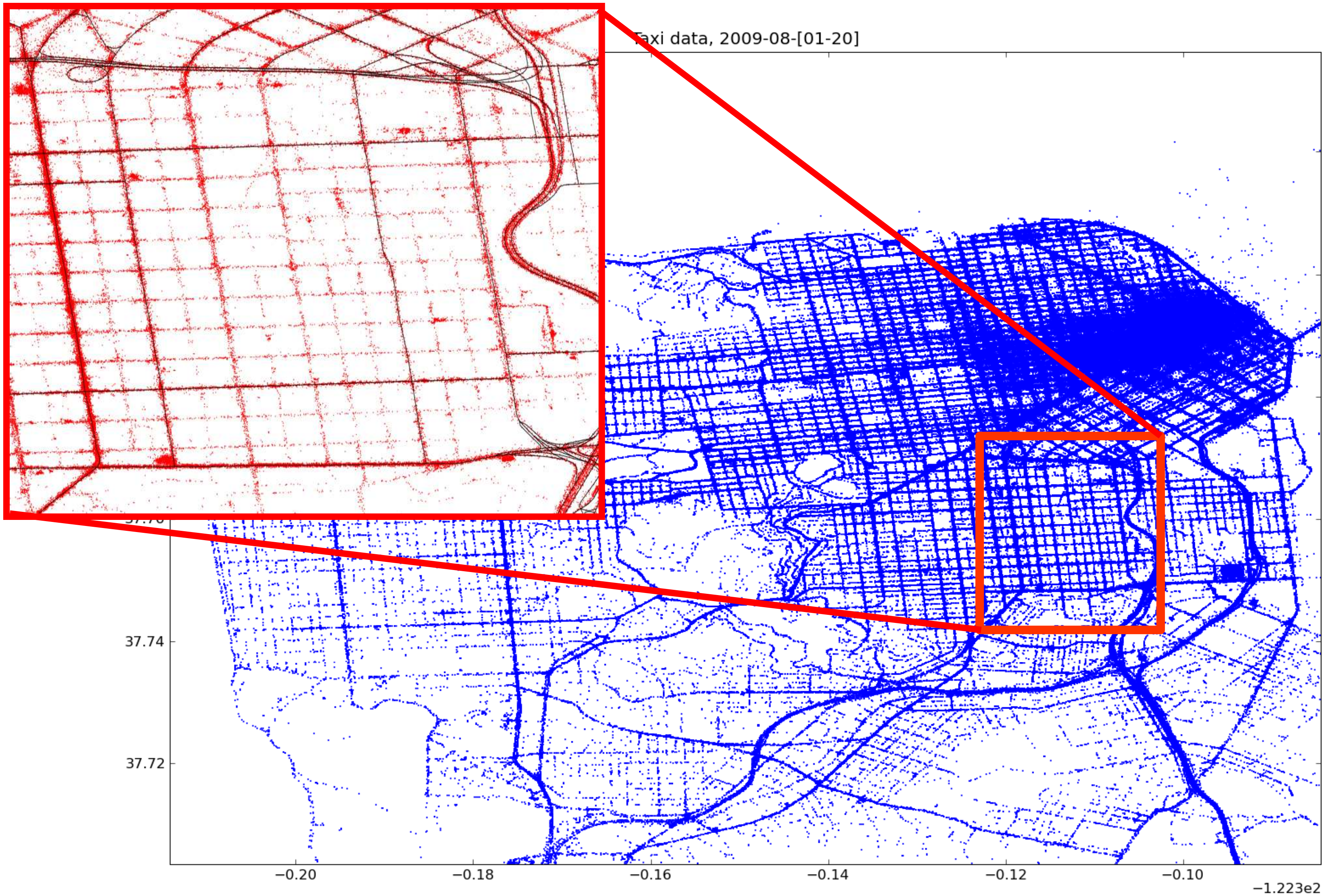


http://traffic.berkeley.edu

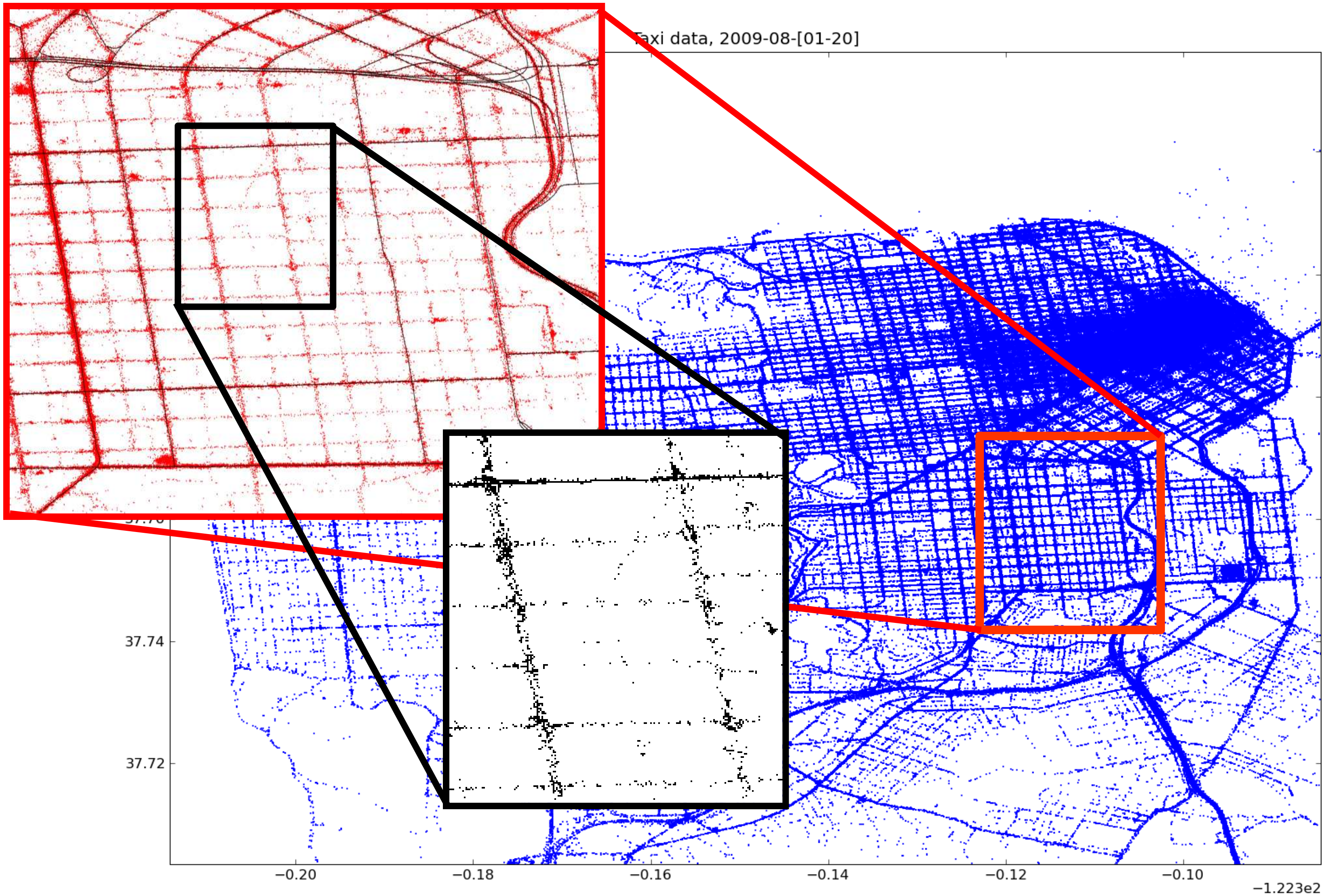
Context of this work: Mobile Millennium project



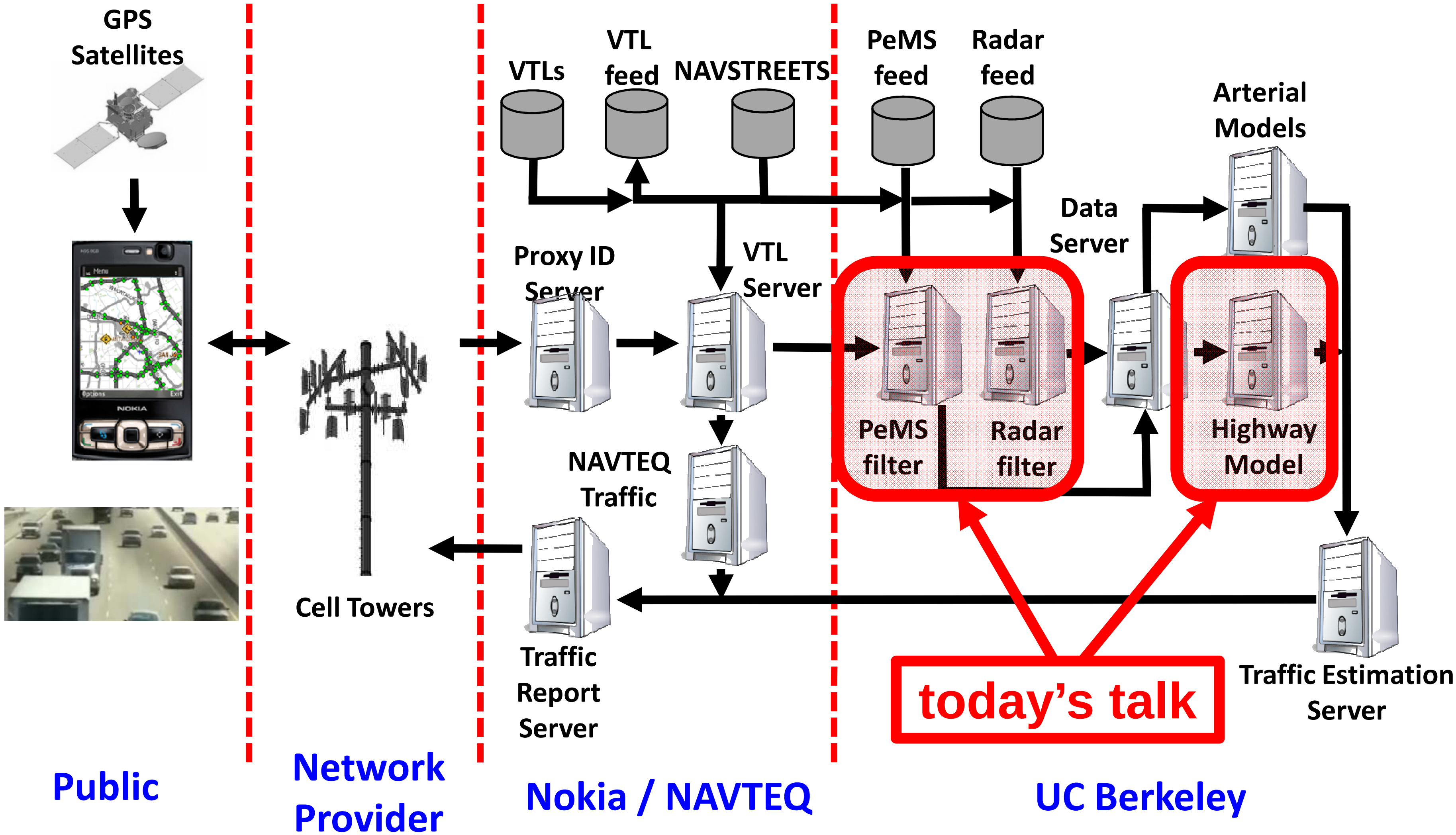
Context of this work: Mobile Millennium project



Context of this work: Mobile Millennium project



Architecture of the Mobile Millennium system



Real time traffic information system:

- more than 2 million data points a day (from the fixed infrastructure)
- more than 500 000 data points a day (from mobile sensing)
- more than 200 GB of data (cumulated)

Outline

Inverse modeling problems

- **Problem description**
- **Introductory example**

Inverse modeling problems involving Hamilton-Jacobi equations

- **Background**
- **Problem definition**
- **Optimization formulations for inverse modeling**
- **Parameter uncertainty**

Sensing applications in traffic flow

- **Fault detection in the PeMS sensor network**
- **Other applications: racing with Google Maps**

Conclusion

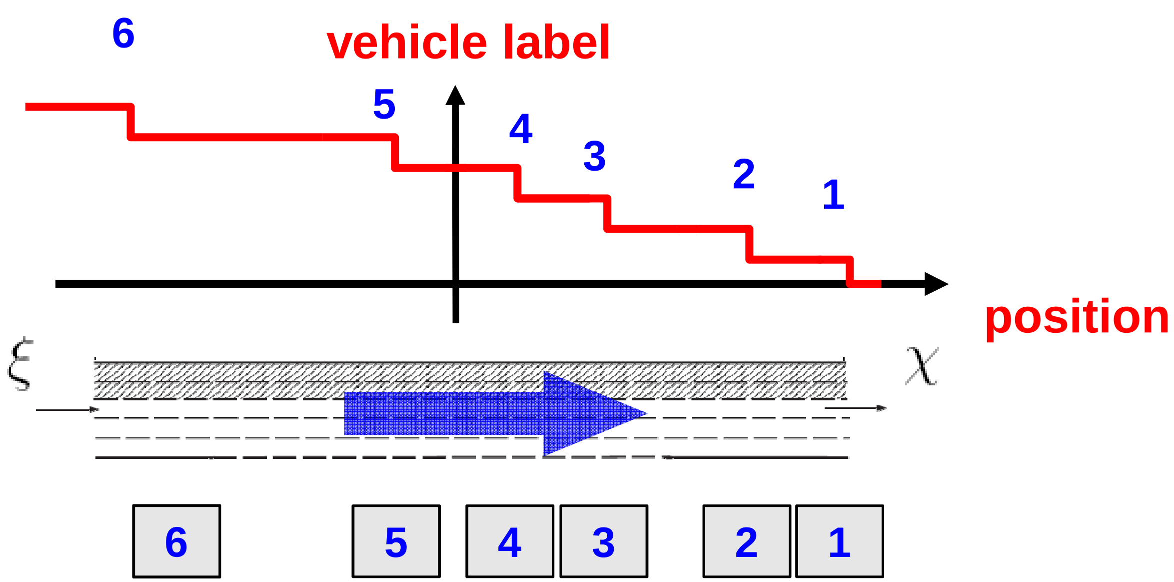
Problem description: state definition

<u>System:</u> highway section

The state of the system is described by the Moskowitz function M(t,x).

- We attribute consecutive labels n to the vehicles entering a stretch of highway

- The Moskowitz function is a continuous function satisfying M(t,x)=n



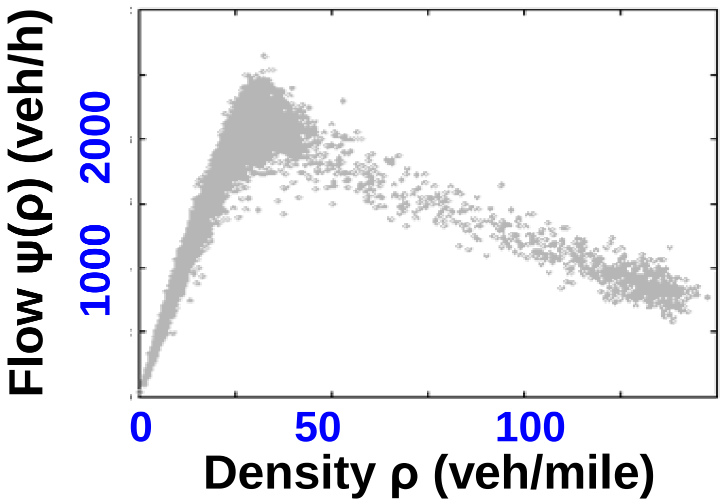
[Newell 93], [Daganzo 03,06]

Model: Hamilton-Jacobi PDE

The Moskowitz function satisfies the following Hamilton-Jacobi (HJ) partial differential equation (PDE), which can be derived from the Lighthill-Whitham-Richards (LWR) PDE

$$\frac{\partial \mathrm{M}(t,x)}{\partial t} - \psi\left(-\frac{\partial \mathrm{M}(t,x)}{\partial x}\right) = 0$$

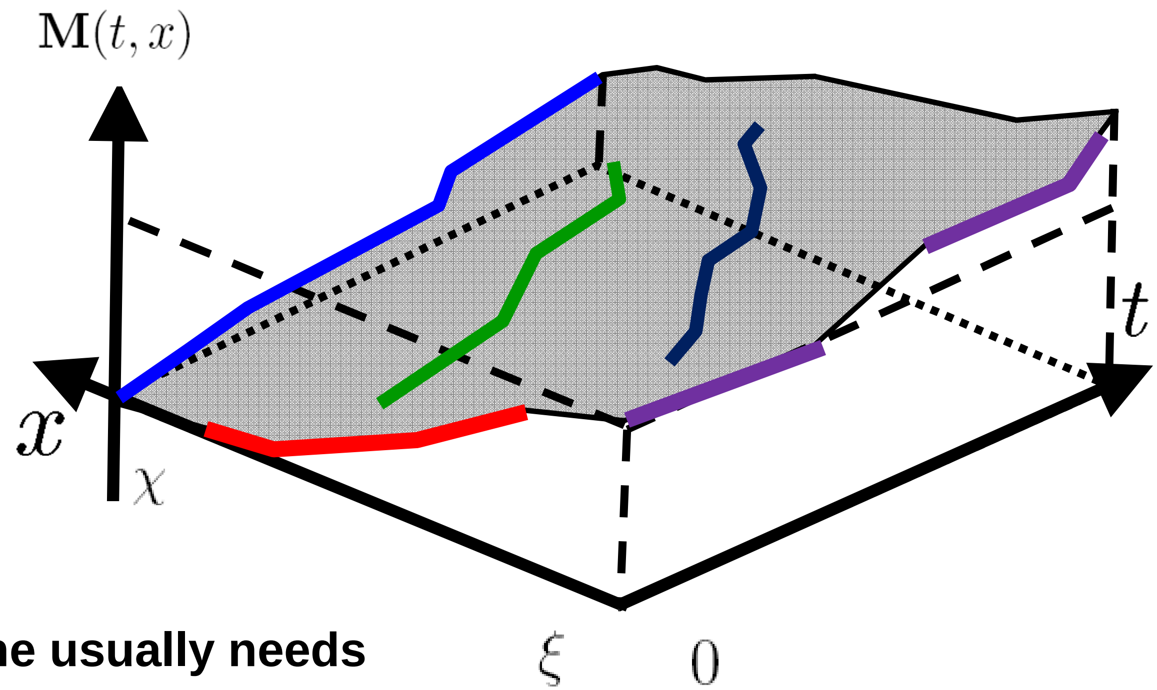
The function ψ is a parameter of the HJ PDE known as "flux function"



Source: PeMS

[Lighthill, Whitham 55], [Richards 56], [Newell 93], [Daganzo 03,06]

Data (boundary conditions)



To solve the PDE, one usually needs

- **Initial condition (satellite, camera)**
- **Upstream boundary condition (loop detector, radar)**
- **Downstream boundary condition (loop detector, radar)**

They are not always known, but we can use in addition:

- **Internal condition 1 (cellphone, taxi, fleet, aftermarket device)**
- **Internal condition 2 (cellphone, taxi, fleet, aftermarket device)**

Outline

Inverse modeling problems

- Problem description
- Introductory example

Inverse modeling problems involving Hamilton-Jacobi equations

- Background
- Problem definition
- Optimization formulations for inverse modeling
- Parameter uncertainty

Sensing applications in traffic flow

- Fault detection in the PeMS sensor network
- Other applications: racing with Google Maps

Conclusion

Model solution procedure

<u>Input:</u> **data (with error bounds), model parameter**

STEP 1: **Define the proper mathematical solution to the PDE using control theory**

STEP 2: **Express the boundary conditions by piecewise affine (PWA) functions**

STEP 3: **Express the solutions to the HJ PDE as semi-analytic functions**

STEP 4: **Derive the model compatibility conditions as a set of inequality constraints**

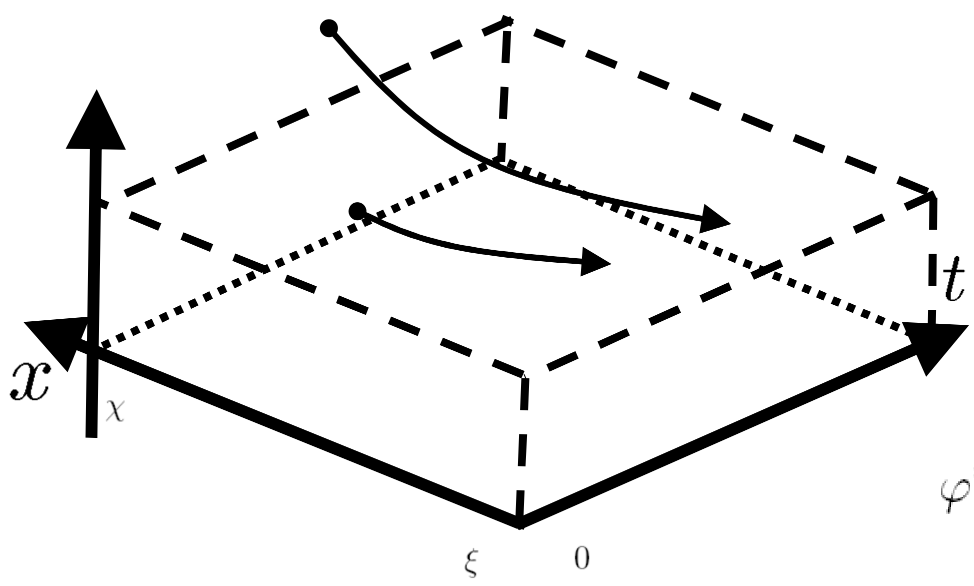
STEP 5: **Solve the problem numerically (LP, QP, GP)**

<u>Output:</u> **possible traffic states compatible with the model and the data**

[Claudel, Bayen, IEEE TAC 09 a,b] [Claudel, Bayen, SIAM SICON 10 (in review)]

STEP 1: Mathematical solution procedure

Control theoretic definition of the solution to the HJ PDE



Controlled dynamical system:

$$\begin{cases} \dot{x}(t) \in F(x(t)) \\ x(0) = x_0 \end{cases}$$

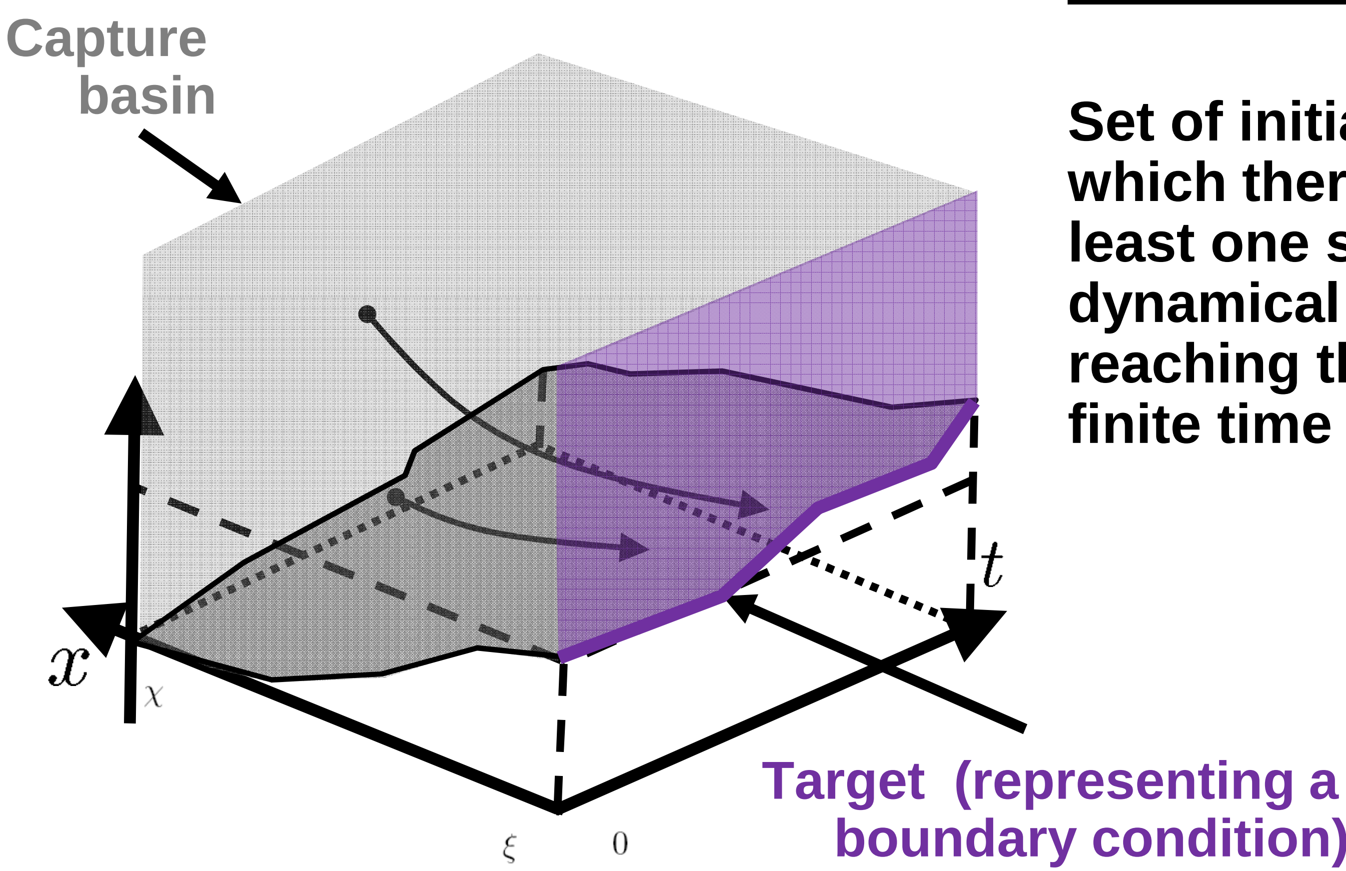
$$F := \begin{cases} \tau'(t) &= -1 \\ x'(t) &= u(t) \\ y'(t) &= -\varphi^*(u(t)) \end{cases}$$

$$\varphi^*(u) := \sup_{p \in \mathrm{Dom}(\psi)} [p \cdot u + \psi(p)]$$

[Aubin 91], [Cardaliaguet, Quincampoix, Saint Pierre 99], [Aubin, Bayen, Saint-Pierre 08]

STEP 1: Mathematical solution procedure

Control theoretic definition of the solution to the HJ PDE



$\mathcal{C} := \mathcal{E}pi(\mathbf{c})$

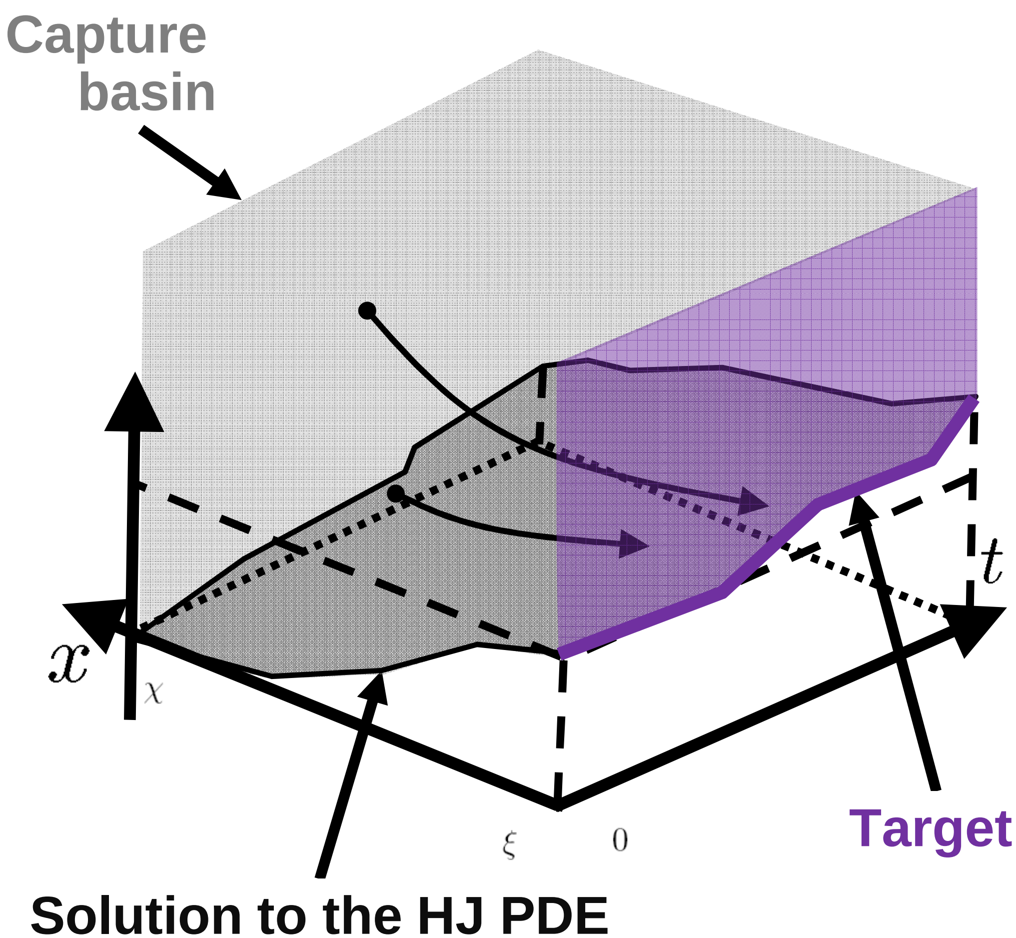
<u>Capture basin of a target:</u>

Set of initial points for which there exists at least one solution of the dynamical system reaching the target in finite time

[Aubin 91], [Cardaliaguet, Quincampoix, Saint Pierre 99], [Aubin, Bayen, Saint-Pierre 08]

STEP 1: Mathematical solution procedure

Control theoretic definition of the solution to the HJ PDE



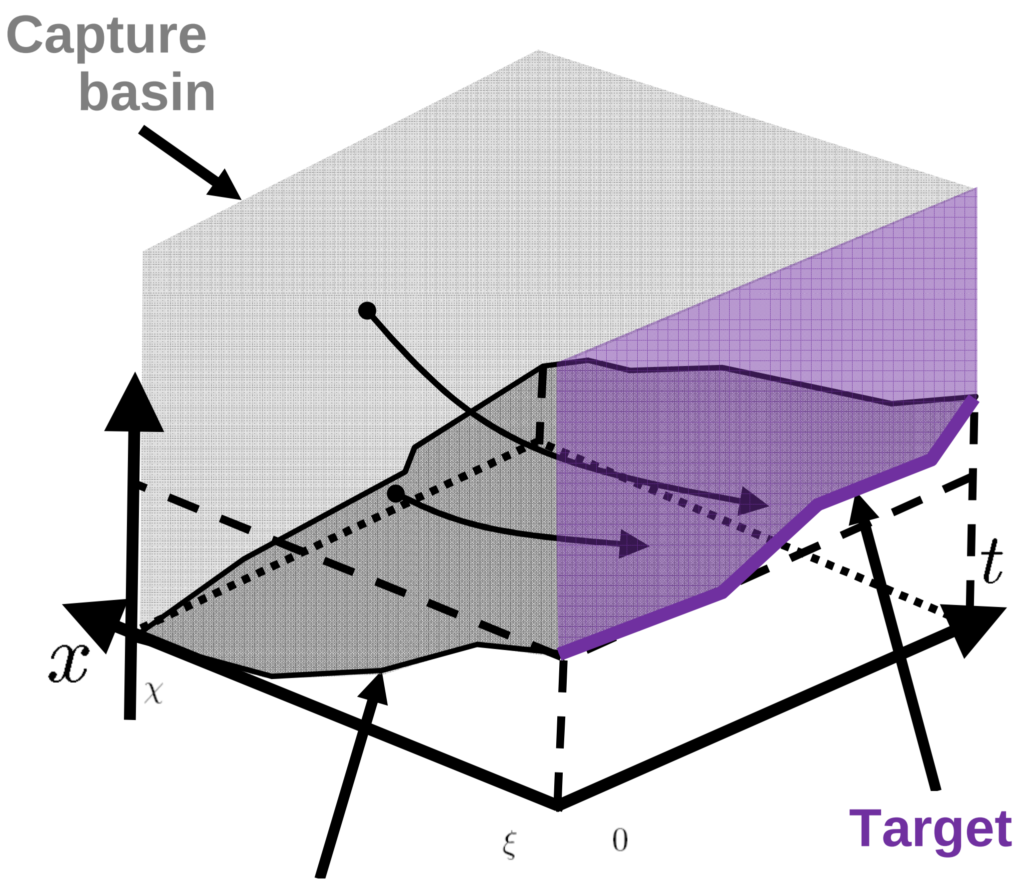
<u>Tangential property:</u>

The lower envelope of the capture basin solves the Hamilton-Jacobi PDE in the Barron-Jensen/Frankowska sense

[Crandall, Evans, Lions 84], [Aubin 91], [Aubin, Bayen Saint-Pierre 08]

STEP 1: Mathematical solution procedure

Control theoretic definition of the solution to the HJ PDE



Solution to the HJ PDE

Data (from sensors)

Boundary conditions

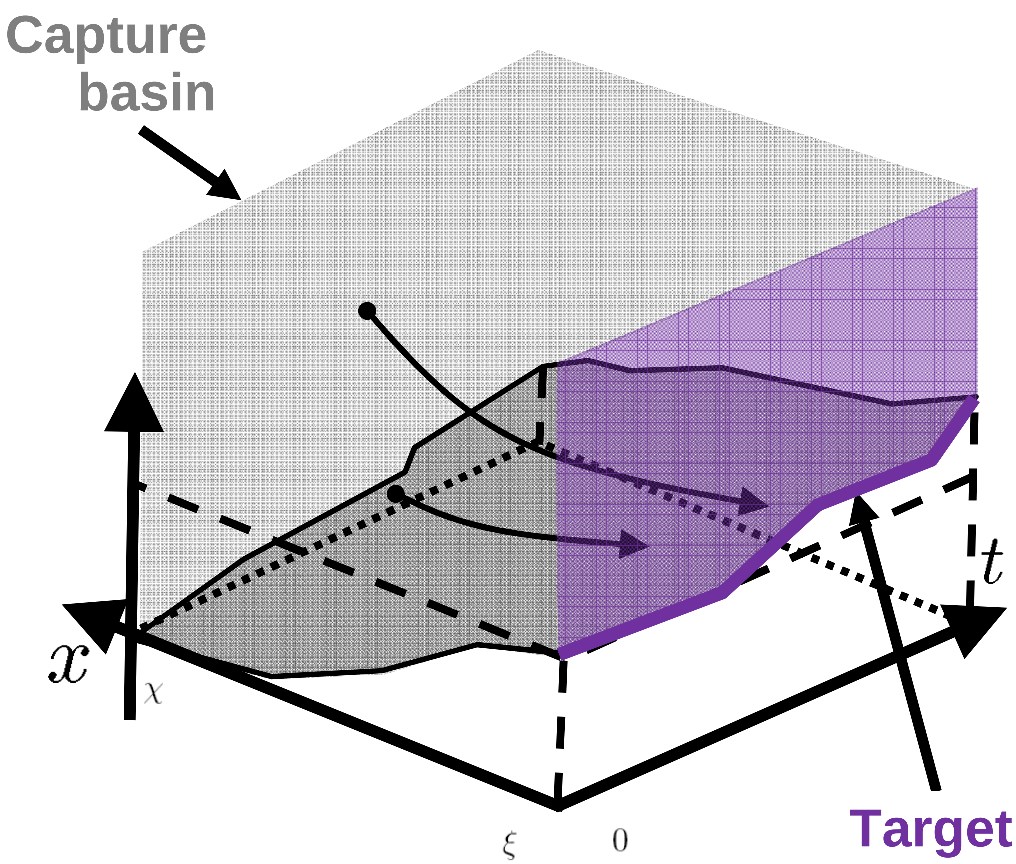
Solve the HJ PDE with these boundary conditions

Solution to the HJ PDE may (or may not) satisfy all the prescribed boundary conditions

[Crandall, Evans, Lions 84], [Aubin 91], [Aubin, Bayen Saint-Pierre 08]

STEP 1: Mathematical solution procedure

Control theoretic definition of the solution to the HJ PDE



A set of boundary conditions is compatible with the model

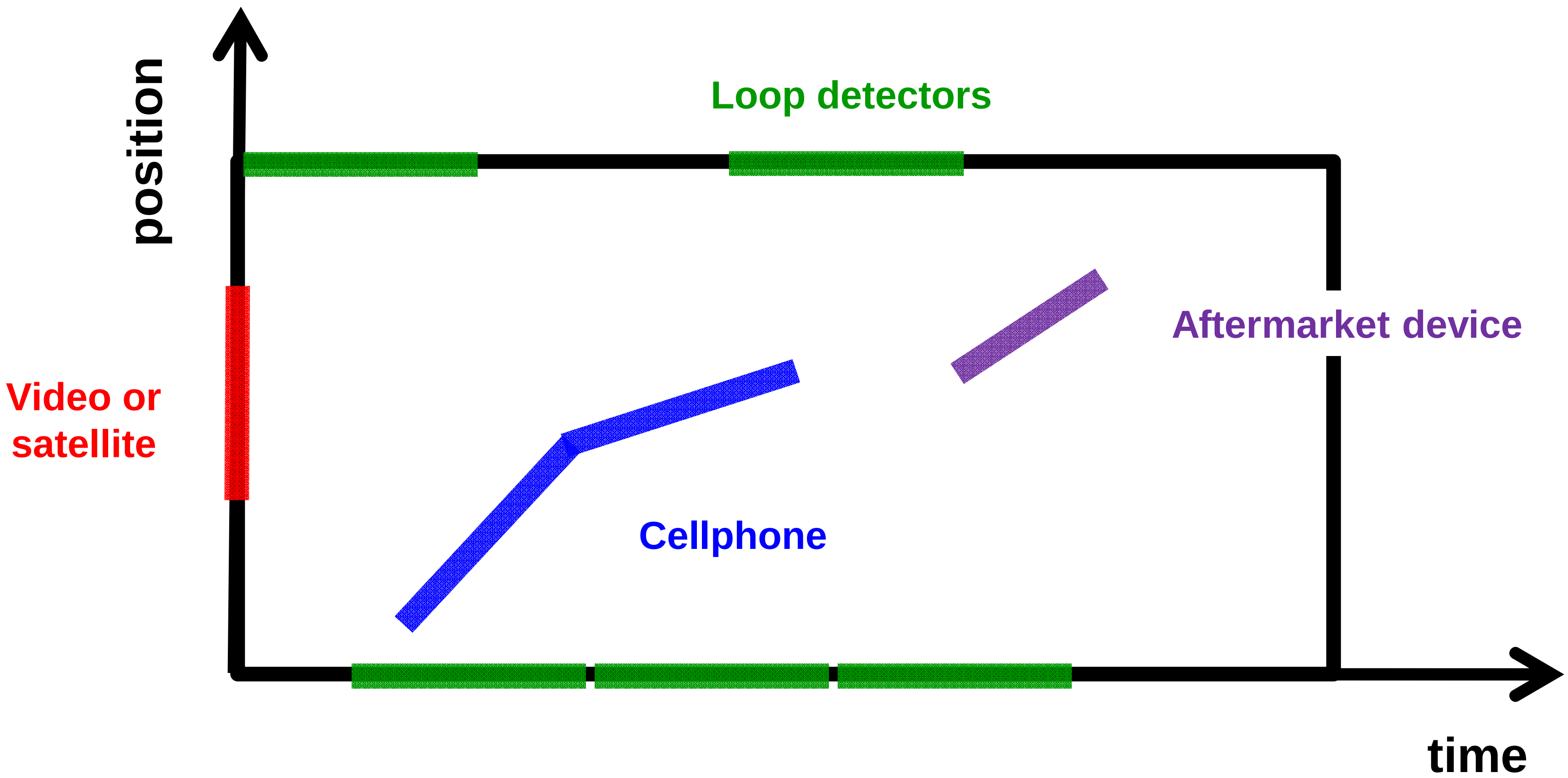
⇕

There exists a solution to the HJ PDE satisfying all the boundary conditions

[Aubin 91], [Aubin, Bayen, Saint-Pierre 08], [Claudel, Bayen, IEEE TAC 09 a]

STEP 2: express the data as PWA functions

The data generated by mobile phones or fixed detectors corresponds to a set of piecewise affine initial, boundary or internal conditions



[Claudel, Bayen, ACC 10 (accepted)], [Claudel, Bayen, SIAM SICON 10 (in review)]

STEP 3: develop a fast semi-analytical scheme

- **Standard schemes:**
  - **Lax-Friedrichs**
  - **Dynamic programming**
  - **Level set methods**

- **For <u>piecewise affine</u> initial, boundary and internal boundary conditions, the solution to the HJ PDE is the minimum of closed-form expression functions**

  Advantages:

  - Exact

  - Faster

<u>Example:</u> solution the HJ PDE associated with an affine internal condition

$$u_p(v_l, g_l) \in -\partial_+ \psi(\rho_p(v_l, g_l)), \quad p \in \{1,2\}$$

$$T_p(t,x,v_l,g_l) := \begin{cases} \frac{x_l + v_l(t-\overline{\delta}_l)-x}{u_p(v_l,g_l)+v_l} & \text{if} \quad u_p(v_l,g_l) \neq -v_l \\ +\infty & \text{if} \quad u_p(v_l,g_l) = -v_l \end{cases}$$

$$\mathrm{M}_{\mu_l}(t,x) = \begin{cases} (i) & \psi(\rho_1(v_l,g_l))(t-\overline{\delta}_l) + (x_l - x)\rho_1(v_l,g_l) + h_l \\ & \text{if} \ \ x_l + v_l(t-\overline{\delta}_l) \le x \\ & \text{and} \ \ T_1(t,x,v_l,g_l) \in \left[t-\overline{\delta}_{l+1}, t-\overline{\delta}_l\right] \\ (ii) & \psi(\rho_2(v_l,g_l))(t-\overline{\delta}_l) + (x_l - x)\rho_2(v_l,g_l) + h_l \\ & \text{if} \ \ x_l + v_l(t-\overline{\delta}_l) \ge x \\ & \text{and} \ \ T_2(t,x,v_l,g_l) \in \left[t-\overline{\delta}_{l+1}, t-\overline{\delta}_l\right] \\ (iii) & h_l + (t-\overline{\delta}_l)\varphi^*\left(\frac{x_l - x}{t-\overline{\delta}_l}\right) \\ & \text{if} \ \ x_l + v_l(t-\overline{\delta}_l) \le x \ \ \text{and} \ \ T_1(t,x,v_l,g_l) \ge t-\overline{\delta}_l \\ & \text{or if} \ \ x_l + v_l(t-\overline{\delta}_l) \ge x \ \ \text{and} \ \ T_2(t,x,v_l,g_l) \ge t-\overline{\delta}_l \\ (iv) & g_l(\overline{\delta}_{l+1} - \overline{\delta}_l) + h_l + (t-\overline{\delta}_{l+1})\varphi^*\left(\frac{x_l + v_l(\overline{\delta}_{l+1}-\overline{\delta}_l)-x}{t-\overline{\delta}_{l+1}}\right) \\ & \text{if} \ \ x_l + v_l(t-\overline{\delta}_l) \le x \ \ \text{and} \ \ T_1(t,x,v_l,g_l) \le t-\overline{\delta}_{l+1} \\ & \text{or if} \ \ x_l + v_l(t-\overline{\delta}_l) \ge x \ \ \text{and} \ \ T_2(t,x,v_l,g_l) \le t-\overline{\delta}_{l+1} \end{cases}$$

[Claudel, Bayen, IEEE TAC 09 b]

STEP 3: develop a fast semi-analytical scheme

Example: solution the HJ PDE associated with an affine internal condition

Intermediate computation → $u_p(v_l, g_l) \in -\partial_+\psi(\rho_p(v_l, g_l)), \quad p \in \{1,2\}$

Model parameter → ψ

$$T_p(t,x,v_l,g_l) := \begin{cases} \frac{x_l + v_l(t-\overline{\delta}_l) - x}{u_p(v_l,g_l)+v_l} & \text{if} \quad u_p(v_l,g_l) \neq -v_l \\ +\infty & \text{if} \quad u_p(v_l,g_l) = -v_l \end{cases}$$

Solution → M_{μ_l}

Coordinates → (t,x)

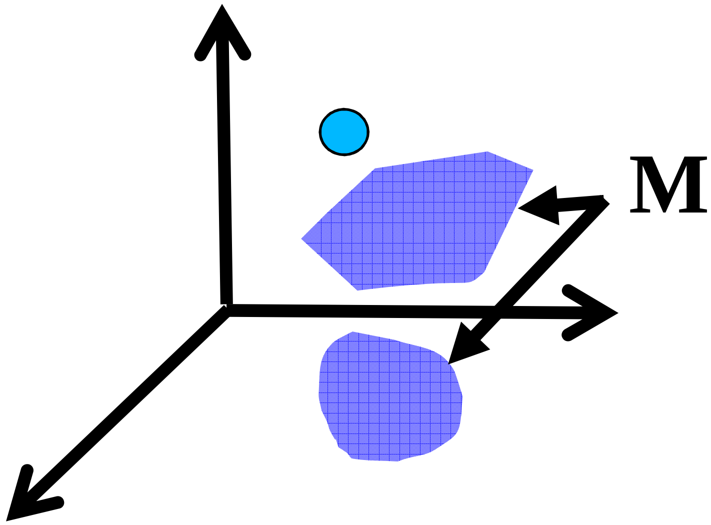
Coefficients of the affine internal condition → $g_l, \overline{\delta}_{l+1}, \overline{\delta}_l, h_l, x_l, v_l$

$$\mathrm{M}_{\mu_l}(t,x) = \begin{cases} (i) & \psi(\rho_1(v_l,g_l))(t-\overline{\delta}_l) + (x_l - x)\rho_1(v_l,g_l) + h_l \\ & \text{if } x_l + v_l(t-\overline{\delta}_l) \le x \\ & \text{and } T_1(t,x,v_l,g_l) \in \left[t-\overline{\delta}_{l+1}, t-\overline{\delta}_l\right] \\ (ii) & \psi(\rho_2(v_l,g_l))(t-\overline{\delta}_l) + (x_l - x)\rho_2(v_l,g_l) + h_l \\ & \text{if } x_l + v_l(t-\overline{\delta}_l) \ge x \\ & \text{and } T_2(t,x,v_l,g_l) \in \left[t-\overline{\delta}_{l+1}, t-\overline{\delta}_l\right] \\ (iii) & h_l + (t-\overline{\delta}_l)\varphi^*\left(\frac{x_l - x}{t-\overline{\delta}_l}\right) \\ & \text{if } x_l + v_l(t-\overline{\delta}_l) \le x \text{ and } T_1(t,x,v_l,g_l) \ge t-\overline{\delta}_l \\ & \text{or if } x_l + v_l(t-\overline{\delta}_l) \ge x \text{ and } T_2(t,x,v_l,g_l) \ge t-\overline{\delta}_l \\ (iv) & g_l(\overline{\delta}_{l+1} - \overline{\delta}_l) + h_l + (t-\overline{\delta}_{l+1})\varphi^*\left(\frac{x_l + v_l(\overline{\delta}_{l+1} - \overline{\delta}_l) - x}{t-\overline{\delta}_{l+1}}\right) \\ & \text{if } x_l + v_l(t-\overline{\delta}_l) \le x \text{ and } T_1(t,x,v_l,g_l) \le t-\overline{\delta}_{l+1} \\ & \text{or if } x_l + v_l(t-\overline{\delta}_l) \ge x \text{ and } T_2(t,x,v_l,g_l) \le t-\overline{\delta}_{l+1} \end{cases}$$

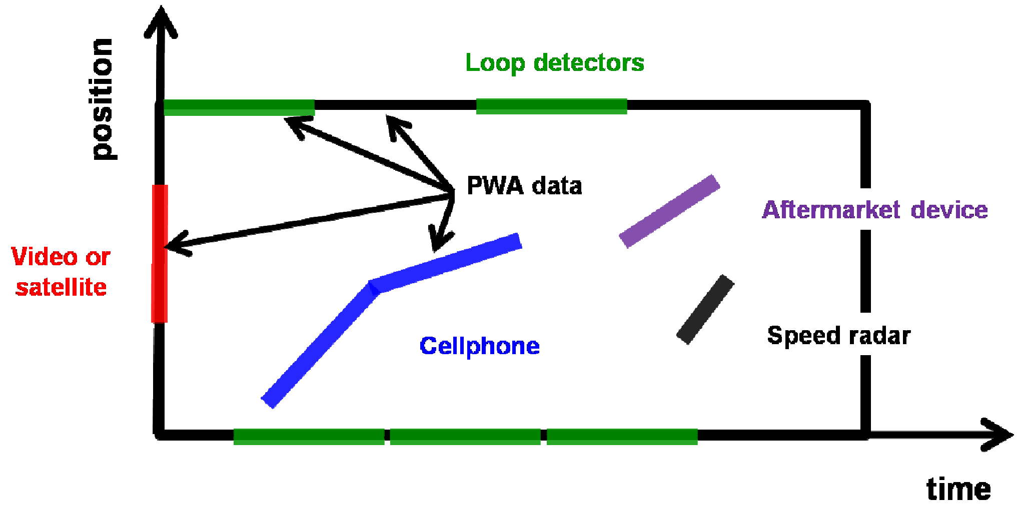
[Claudel, Bayen, IEEE TAC 09 b]

STEP 4: compatibility conditions as convex inequalities

<u>Problem:</u> how can we check quickly if a set of coefficients (of PWA boundary conditions) satisfies the model?



Coefficients of the PWA boundary conditions



[Claudel Bayen, SIAM SICON 10 (in review)]

STEP 4: compatibility conditions as convex inequalities

Depends on the boundary condition coefficients

Checking that there exists a solution to the Hamilton-Jacobi PDE amounts to checking that a set of convex inequalities is satisfied

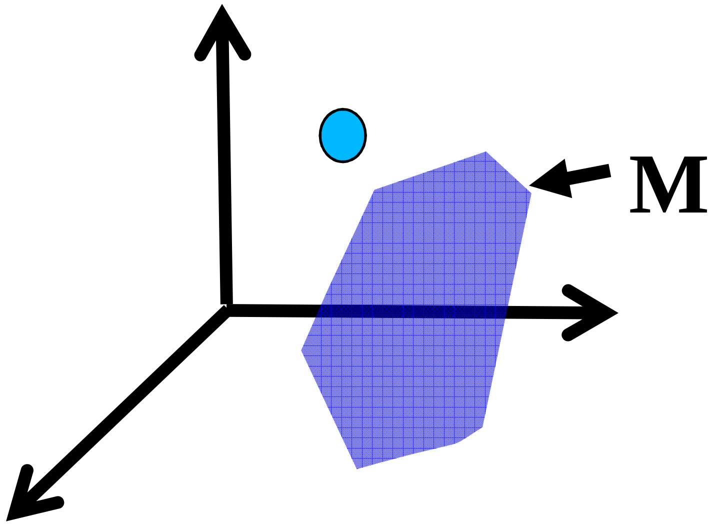
Explicit solutions to the HJ PDE (depend on boundary condition coefficients and model parameter)

$$
\left\{
\begin{array}{ll}
(i) & a_1\overline{\alpha}_1 + b_1 = c_1\overline{\gamma}_1 + d_1 \\
(ii) & a_{i_{\max}}\overline{\alpha}_{i_{\max}+1} + b_{i_{\max}} = e_1\gamma_1 + f_1 \\
(iii) & \mathrm{M}_{\mathcal{M}_{0,i}}(t,\xi) \geq c_j t + d_j \\
 & \forall t \in [\overline{\gamma}_j, \overline{\gamma}_{j+1}], \forall j \in J, \quad \forall i \in I \\
(iv) & \mathrm{M}_{\mathcal{M}_{0,i}}(t,\chi) \geq e_k t + f_k \\
 & \forall t \in [\overline{\beta}_k, \overline{\beta}_{k+1}], \quad \forall k \in K, \quad \forall i \in I \\
(v) & \mathrm{M}_{\gamma_j}(t,\chi) \geq e_k t + f_k \\
 & \forall t \in [\overline{\beta}_k, \overline{\beta}_{k+1}], \quad \forall k \in K, \quad \forall j \in J \\
(vi) & \mathrm{M}_{\beta_k}(t,\xi) \geq c_j t + d_j \\
 & \forall t \in [\overline{\beta}_j, \overline{\beta}_{j+1}], \quad \forall j \in J, \quad \forall \in K \\
(vii) & \mathrm{M}_{\gamma_j}(t, v_{p_l}(t-\overline{\delta}_{pl}) + x_{p_l}) \geq g_{p_l}(t-\overline{\delta}_{pl}) + h_{p_l} \\
 & \forall t \in [\overline{\delta}_{pl}, \overline{\delta}_{pl+1}], \quad \forall l \in L_p, \quad \forall p \in P, \quad \forall j \in J \\
(viii) & \mathrm{M}_{\beta_k}(t, v_{p_l}(t-\overline{\delta}_{pl}) + x_l) \geq g_{p_l}(t-\overline{\delta}_{pl}) + h_{p_l} \\
 & \forall t \in [\overline{\delta}_{pl}, \overline{\delta}_{pl+1}], \quad \forall l \in L_p, \quad \forall p \in P, \quad \forall k \in K \\
(ix) & \mathrm{M}_{\mu_{p_l}}(t,\xi) \geq c_j t + d_j \\
 & \forall t \in [\overline{\gamma}_j, \overline{\gamma}_{j+1}], \quad \forall p \in P, \quad \forall l \in L_p \\
(x) & \mathrm{M}_{\mu_{p_l}}(t,\xi) \geq e_k t + f_k \\
 & \forall t \in [\overline{\beta}_k, \overline{\beta}_{k+1}], \quad \forall p \in P, \quad \forall l \in L_p \\
(xi) & \mathrm{M}_{\mathcal{M}_{0,i}}(t, v_{p_l}(t-\overline{\delta}_{pl}) + x_{p_l}) \geq g_{p_l}(t-\overline{\delta}_{pl}) + h_{p_l} \\
 & \forall t \in [\overline{\delta}_{pl}, \overline{\delta}_{pl+1}], \quad \forall i \in I, \quad \forall l \in L_p, \quad \forall p \in P \\
(xii) & \mathrm{M}_{\mu_{q_m}}(t, v_{p_l}(t-\overline{\delta}_{pl}) + x_{p_l}) \geq g_{p_l}(t-\overline{\delta}_{pl}) + h_{p_l} \\
 & \forall t \in [\overline{\delta}_{pl}, \overline{\delta}_{pl+1}], \quad \forall l \in L_p, \quad \forall p \in P, \quad \forall m \in L_q, \\
 & \forall q \in P \backslash \{p\}
\end{array}
\right.
$$

[Claudel, Bayen, IEEE TAC 09 a,b]

STEP 4: compatibility conditions as convex inequalities

The inequalities defining the compatibility with the model are convex in terms of the parameters of the PWA conditions

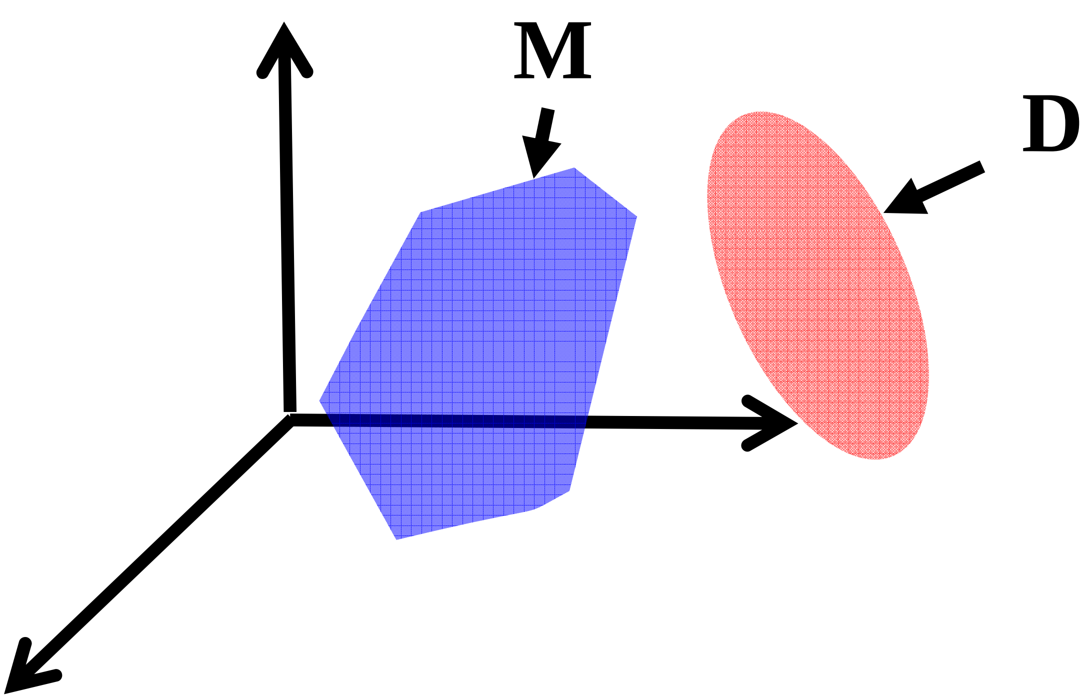


Coefficients of the PWA boundary conditions

[Claudel Bayen, SIAM SICON 10 (in review)]

STEP 5: implementation using convex programming

The set of parameters of the PWA conditions that are compatible with the data is also convex (with usual error models)

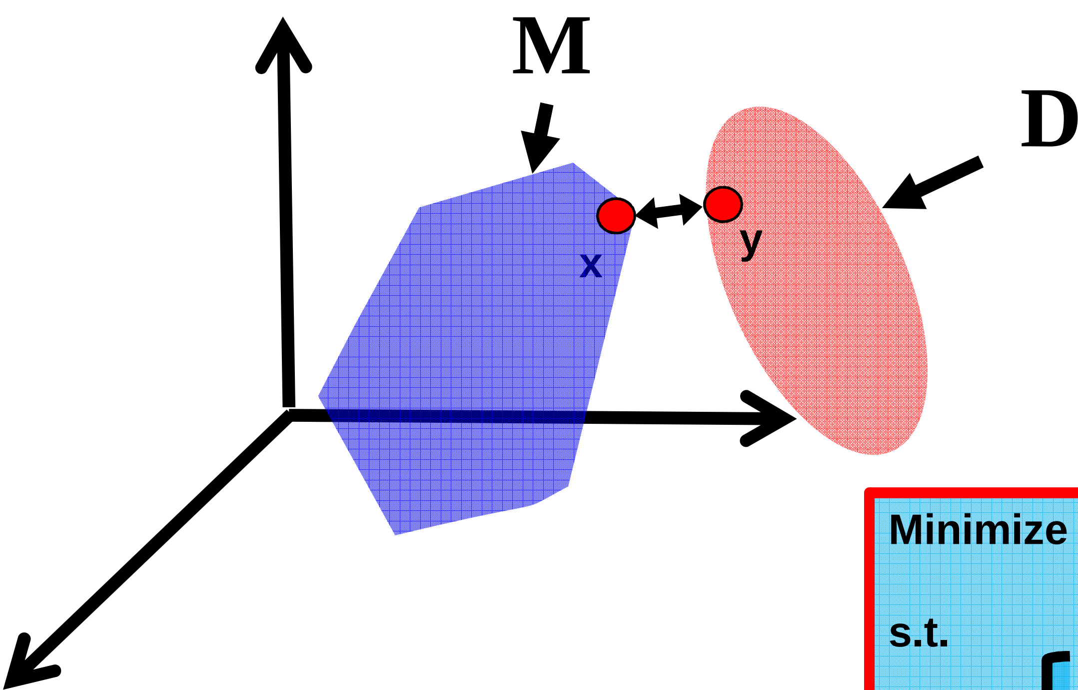


Coefficients of the PWA boundary conditions

[Claudel Bayen, SIAM SICON 10 (in review)]

STEP 5: implementation using convex programming

Data assimilation and reconciliation problems: finding the minimum distance between M and D



Coefficients of the PWA boundary conditions

Minimize ||x-y||

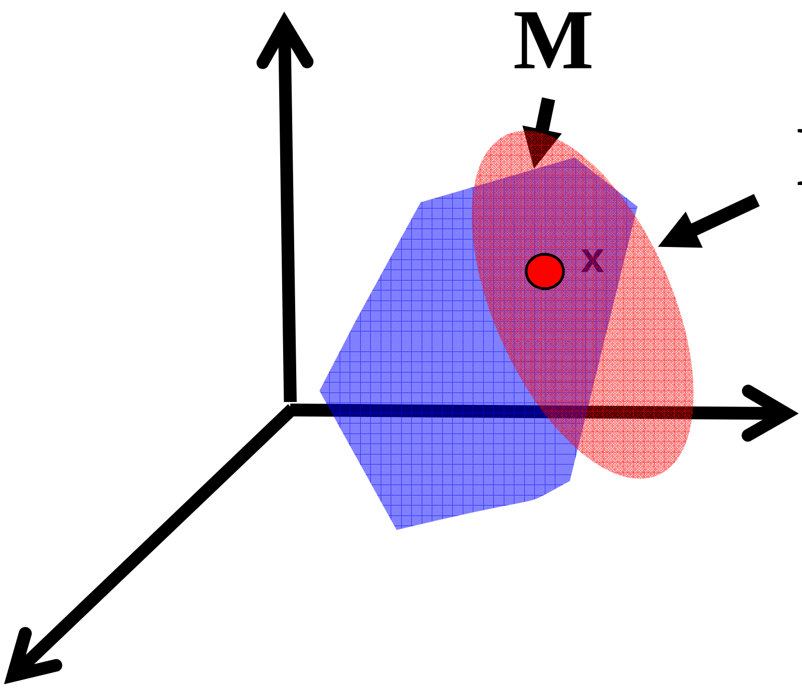
s.t.

- **x satisfies the convex inequality constraints(M)**
- **y satisfies the convex inequality constraints (D)**

[Claudel Bayen, SIAM SICON 10 (in review)]

STEP 5: implementation using convex programming

Bounds on PWA condition parameters (traffic parameters) x_i



Coefficients of the PWA boundary conditions

Minimize (or maximize) x_i

s.t.

{ x satisifies the convex inequality constraints (M) and (D)

[Claudel Bayen, SIAM SICON 10 (in review)]

Outline

Inverse modeling problems

- **Problem description**
- **Introductory example**

Inverse modeling problems involving Hamilton-Jacobi equations

- **Background**
- **Problem definition**
- **Optimization formulations for inverse modeling**
- **Parameter uncertainty**

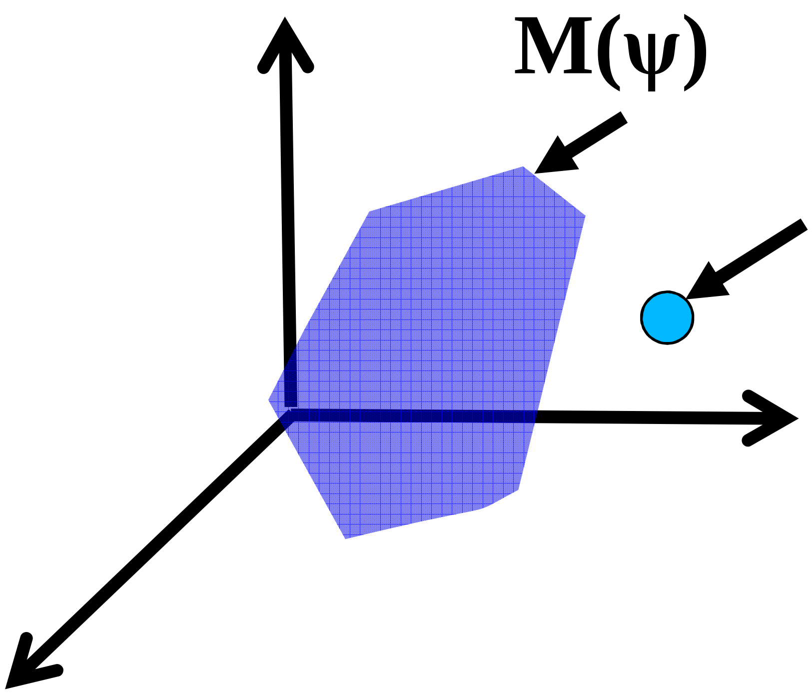
Sensing applications in traffic flow

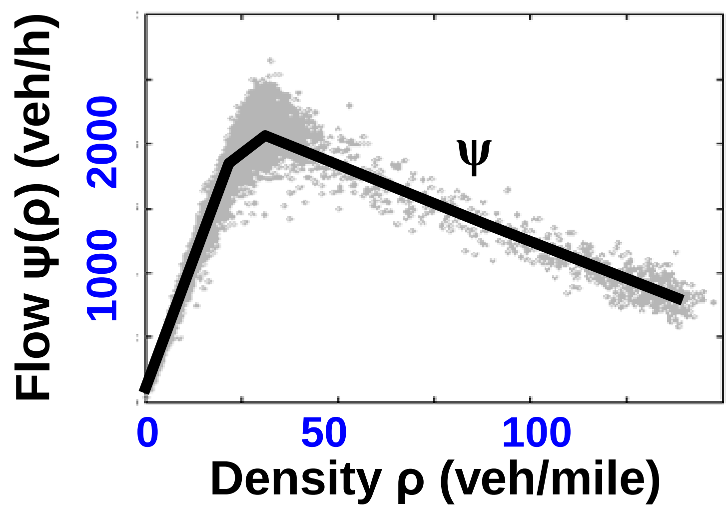
- **Fault detection in the PeMS sensor network**
- **Other applications: racing with Google Maps**

Conclusion

Model parameter uncertainty

The model parameter ψ is usually uncertain. Nothing guarantees that real data solves the model exactly.

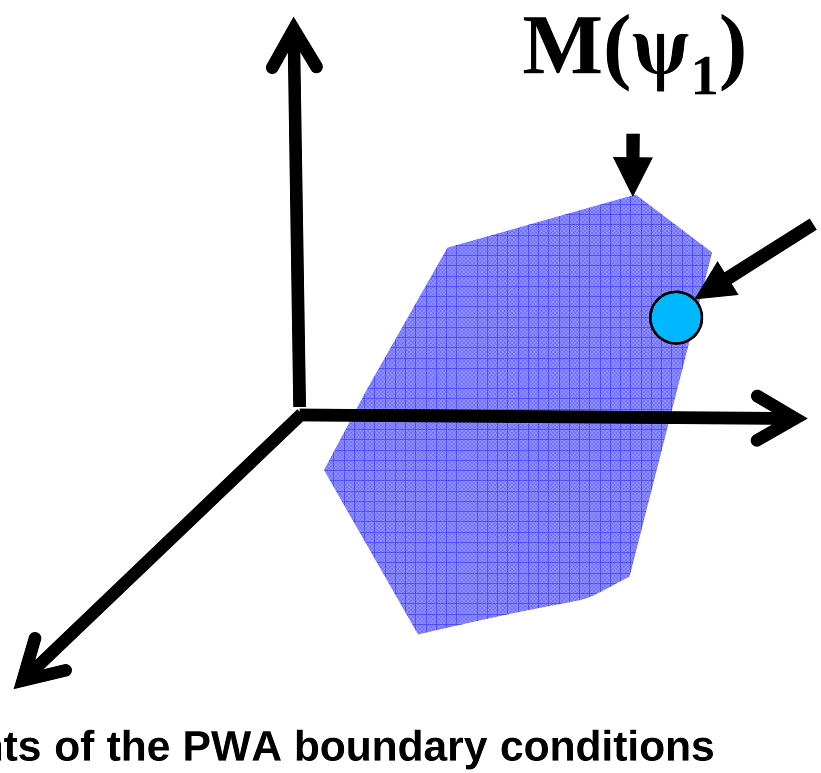


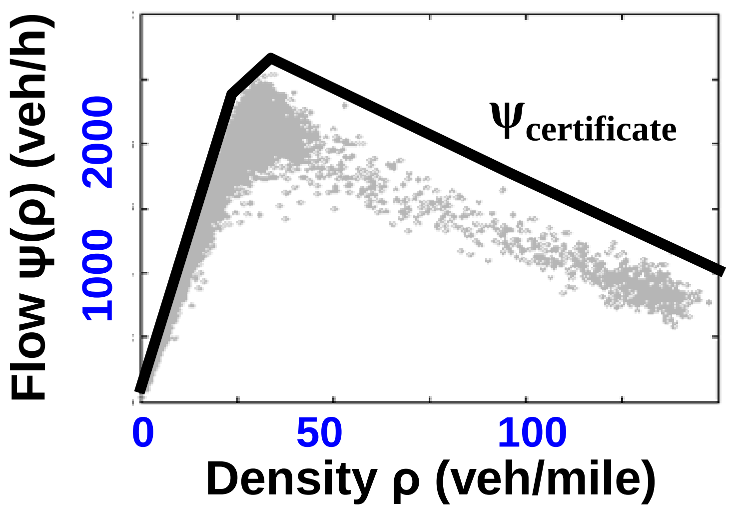


[Claudel, Bayen, Allerton CCC 2009]

Model parameter uncertainty

The real value of the parameters is guaranteed to satisfy the HJ PDE, for a class of parameters ψ

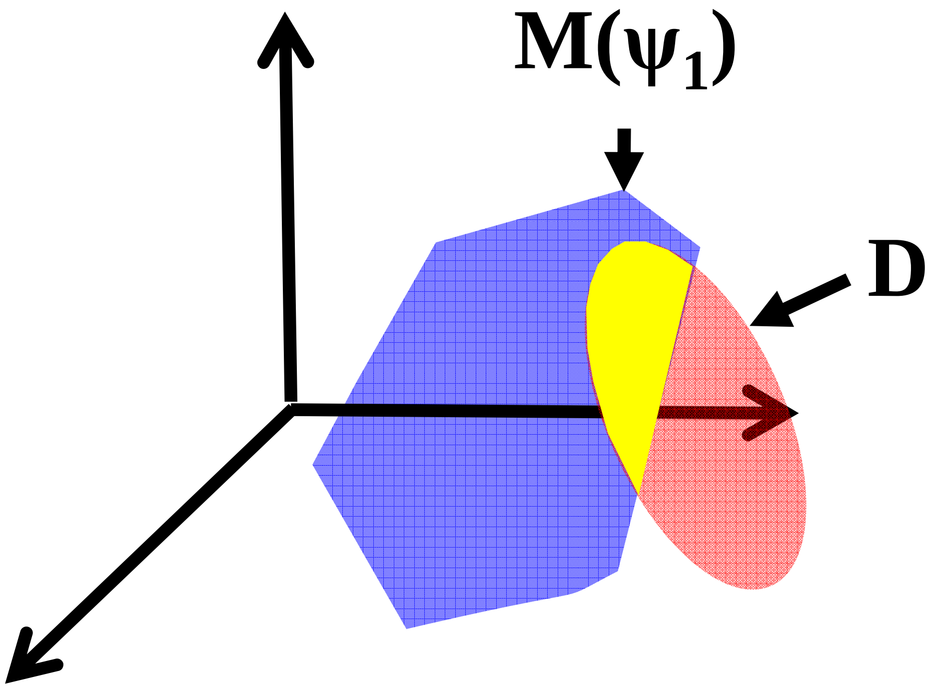




[Claudel, Bayen, Allerton CCC 2009]

Model parameter uncertainty

The real value of the parameters is guaranteed to satisfy the HJ PDE, for a class of parameters ψ



Guaranteed bounds on traffic parameters

Coefficients of the PWA boundary conditions

[Claudel, Bayen, Allerton CCC 2009]

Outline

Inverse modeling problems

- Problem description
- Introductory example

Inverse modeling problems involving Hamilton-Jacobi equations

- Background
- Problem definition
- Optimization formulations for inverse modeling
- Parameter uncertainty

Sensing applications in traffic flow

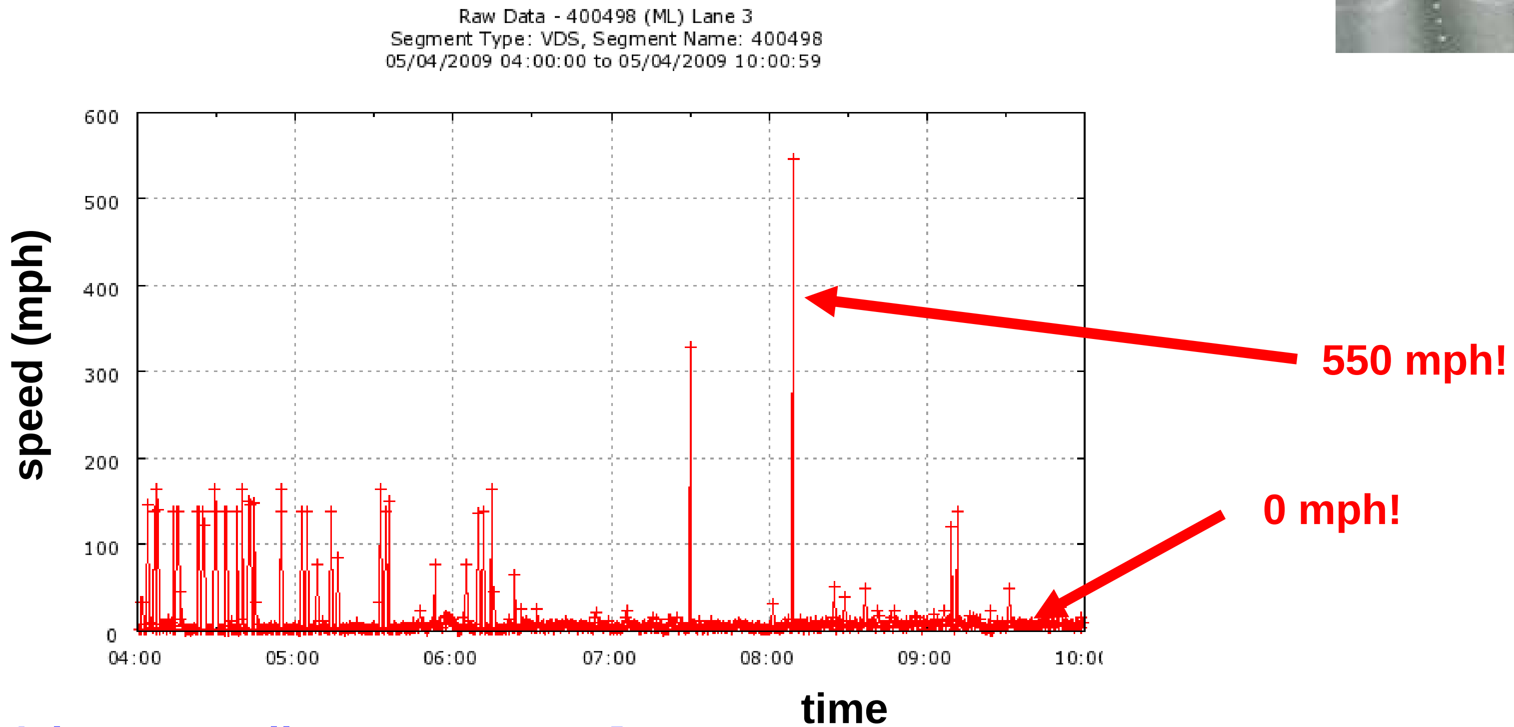
- Fault detection in the PeMS sensor network
- Other applications: racing with Google Maps

Conclusion

Fault detection in sensor infrastructure

PeMS loop detector network:

- 1200 sensors in the San Francisco Bay Area
- poor reliability (70% availability in average)
- detecting sensor failures is a big problem

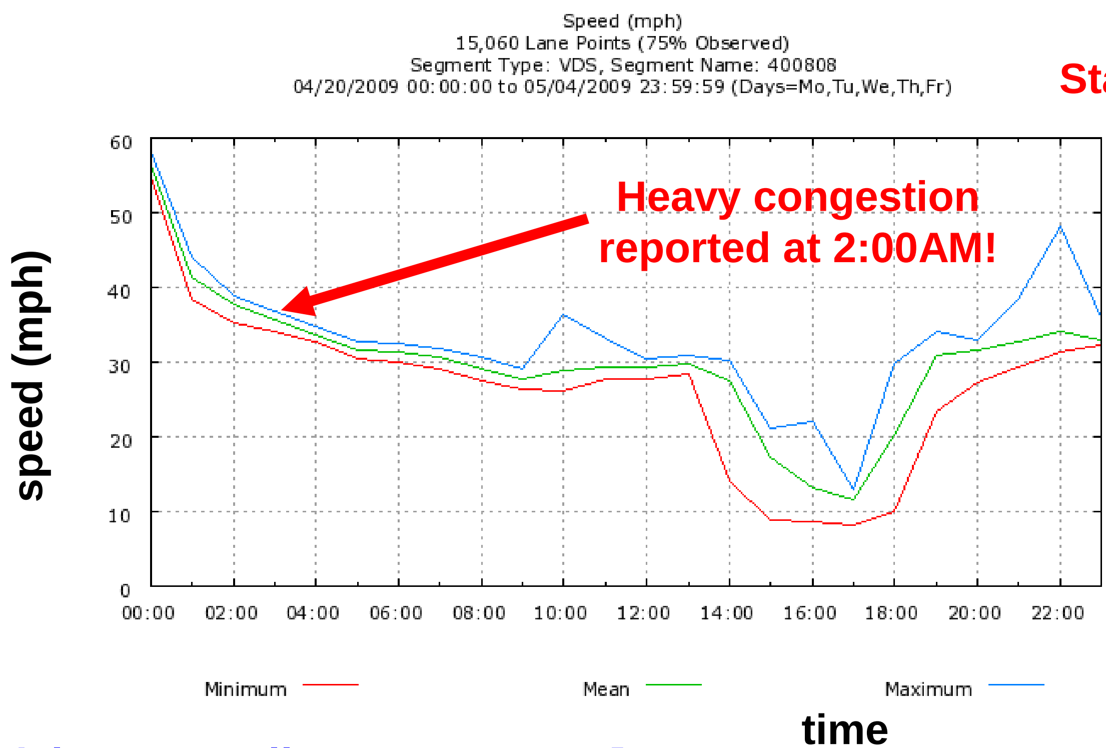


[Claudel, Bayen, Allerton CCC 2009]

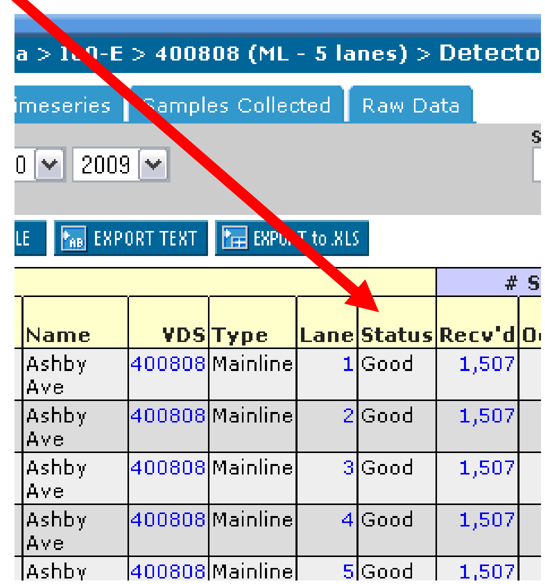
Fault detection in sensor infrastructure

PeMS loop detector network:

- **1200 sensors in the San Francisco Bay Area**
- **poor reliability (70% availability in average)**
- **detecting sensor failures is a big problem**



Status: good!

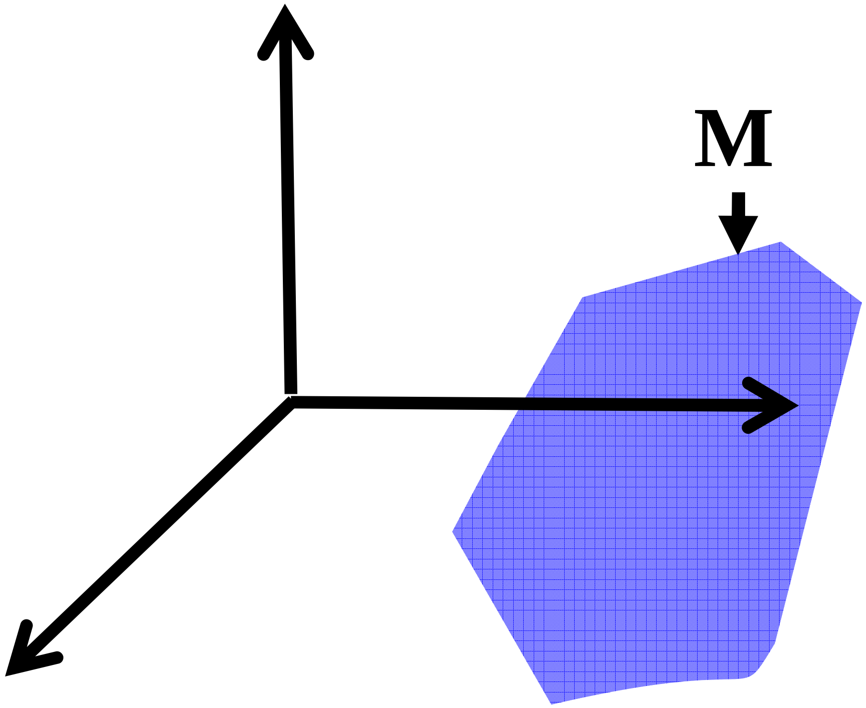


[Claudel, Bayen, Allerton CCC 2009]

Fault detection in sensor infrastructure

Model-based fault detection

Checking (on multiple sensors) that model, data and sensor specifications (maximal error level) are consistent



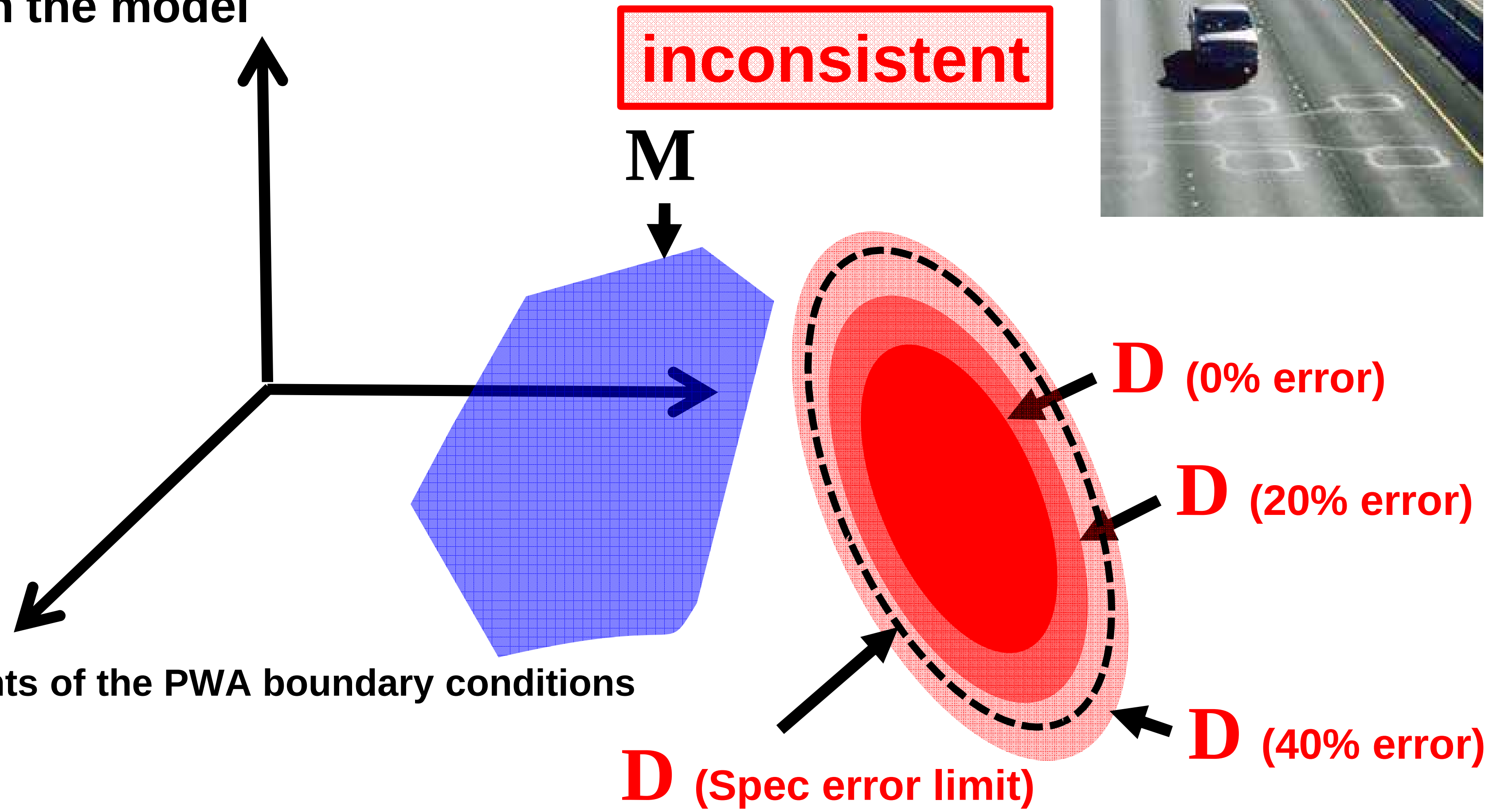
Coefficients of the PWA boundary conditions

[Claudel, Bayen, Allerton CCC 2009]

Fault detection in sensor infrastructure

Minimal error certificates:

What is the minimal possible relative error of both sensors so that the data is consistent with the model



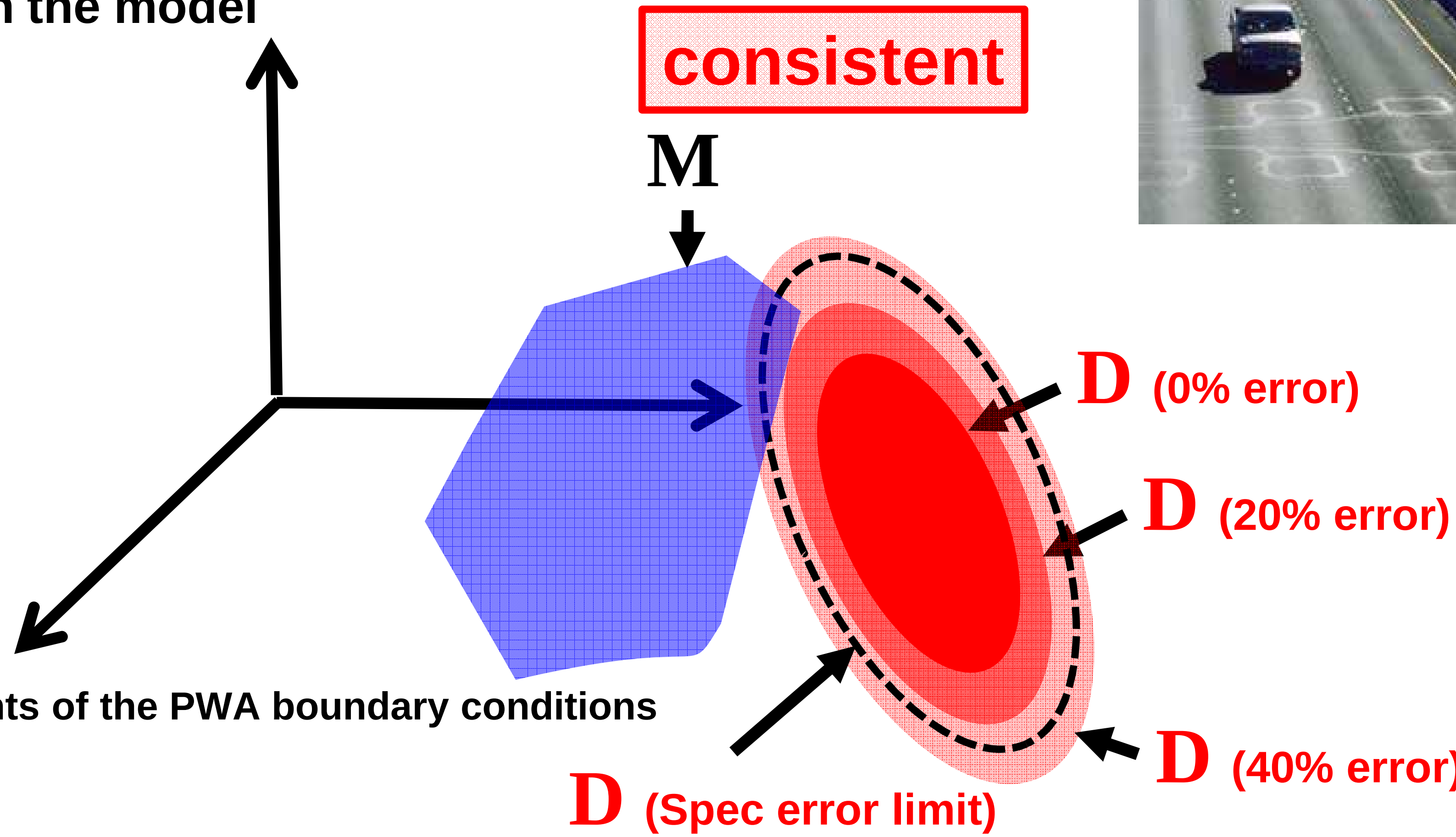
Coefficients of the PWA boundary conditions

[Claudel, Bayen, Allerton CCC 2009]

Fault detection in sensor infrastructure

Minimal error certificates:

What is the minimal possible relative error of both sensors so that the data is consistent with the model

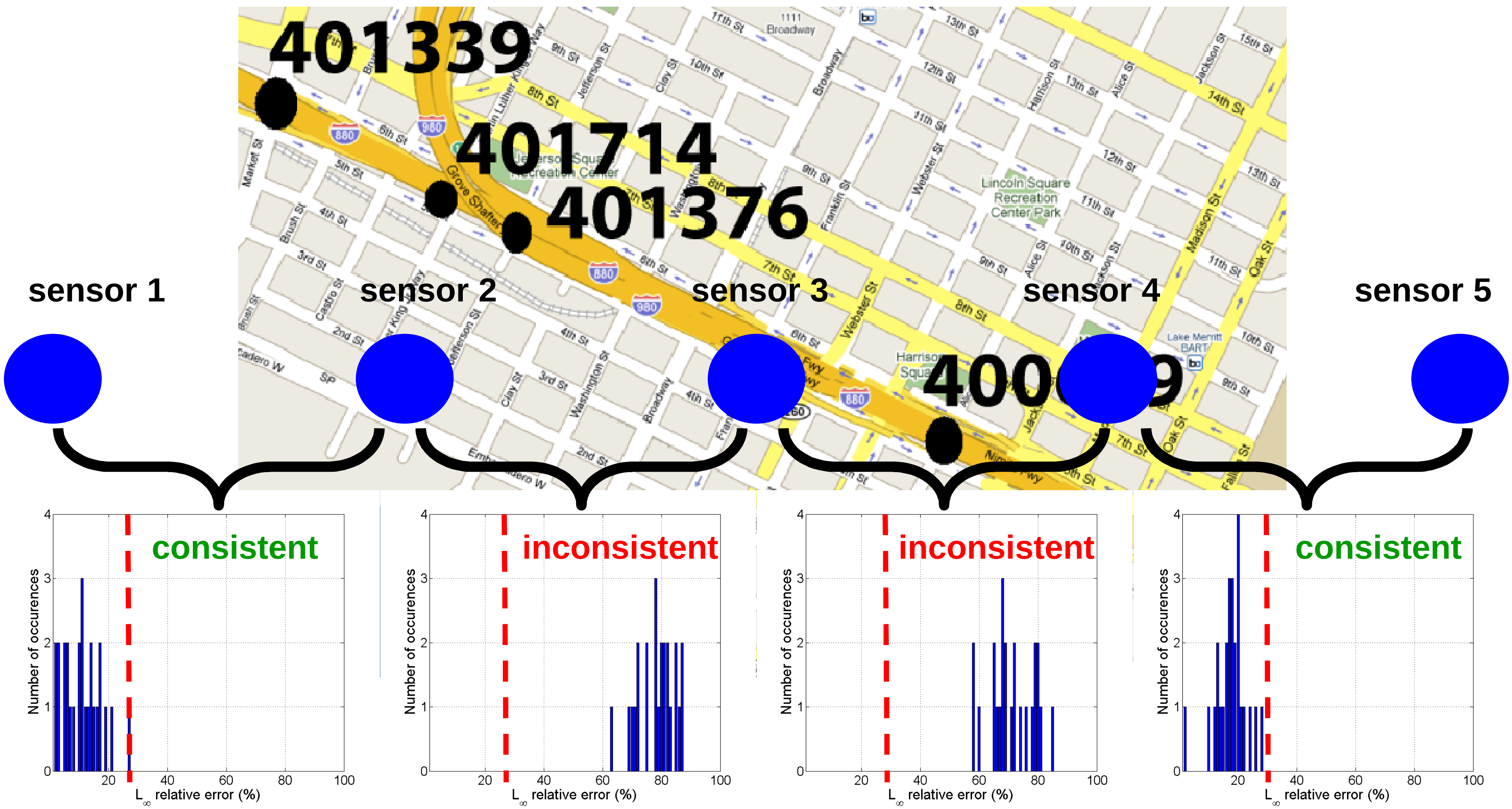


Coefficients of the PWA boundary conditions

[Claudel, Bayen, Allerton CCC 2009]

Fault detection in sensor infrastructure

Example of sensor fault detection (actually sensor misplacement)



Fault detection in sensor infrastructure

Example of sensor fault detection (actually sensor misplacement)

Outline

Inverse modeling problems

- Problem description
- Introductory example

Inverse modeling problems involving Hamilton-Jacobi equations

- Background
- Problem definition
- Optimization formulations for inverse modeling
- Parameter uncertainty

Sensing applications in traffic flow

- Fault detection in the PeMS sensor network
- Other applications: racing with Google Maps

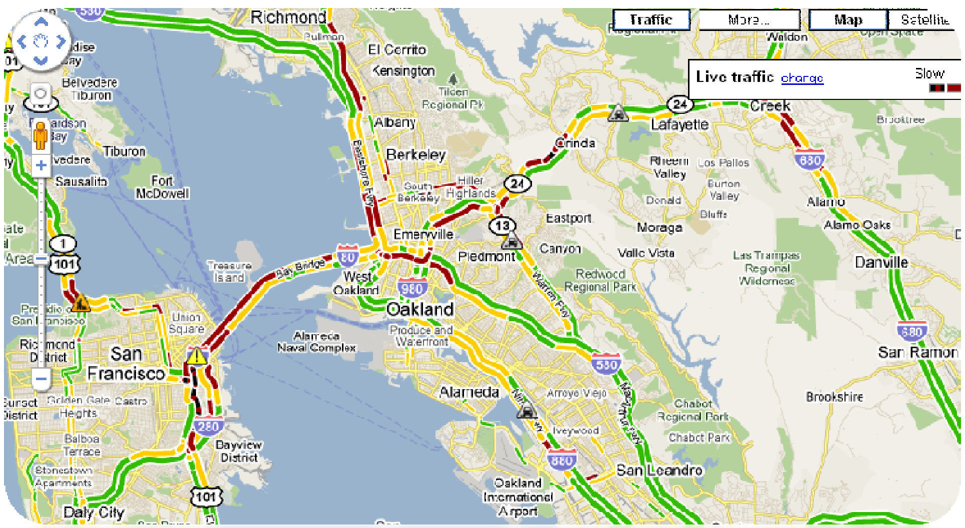
Conclusion

Google Maps vs. HJ PDE

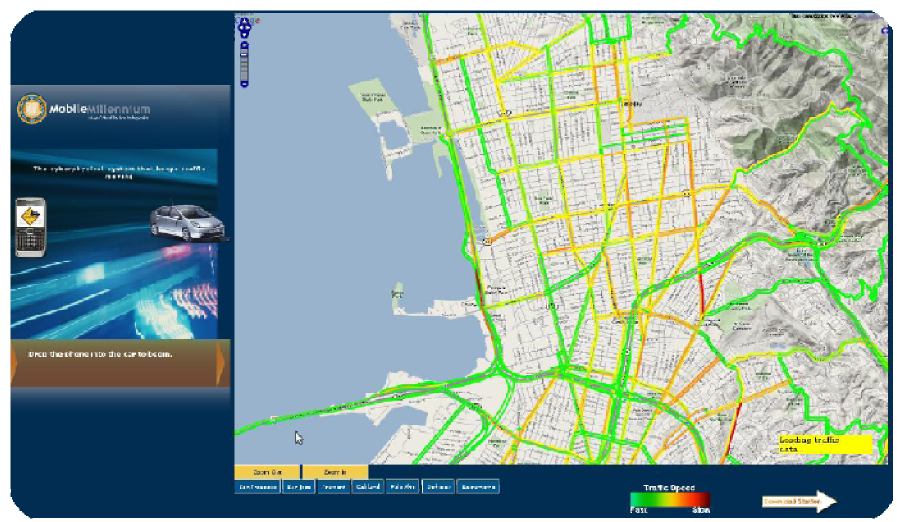
<u>Setup:</u>

February March 20th, 2009
1:30PM (Friday afternoon congestion)

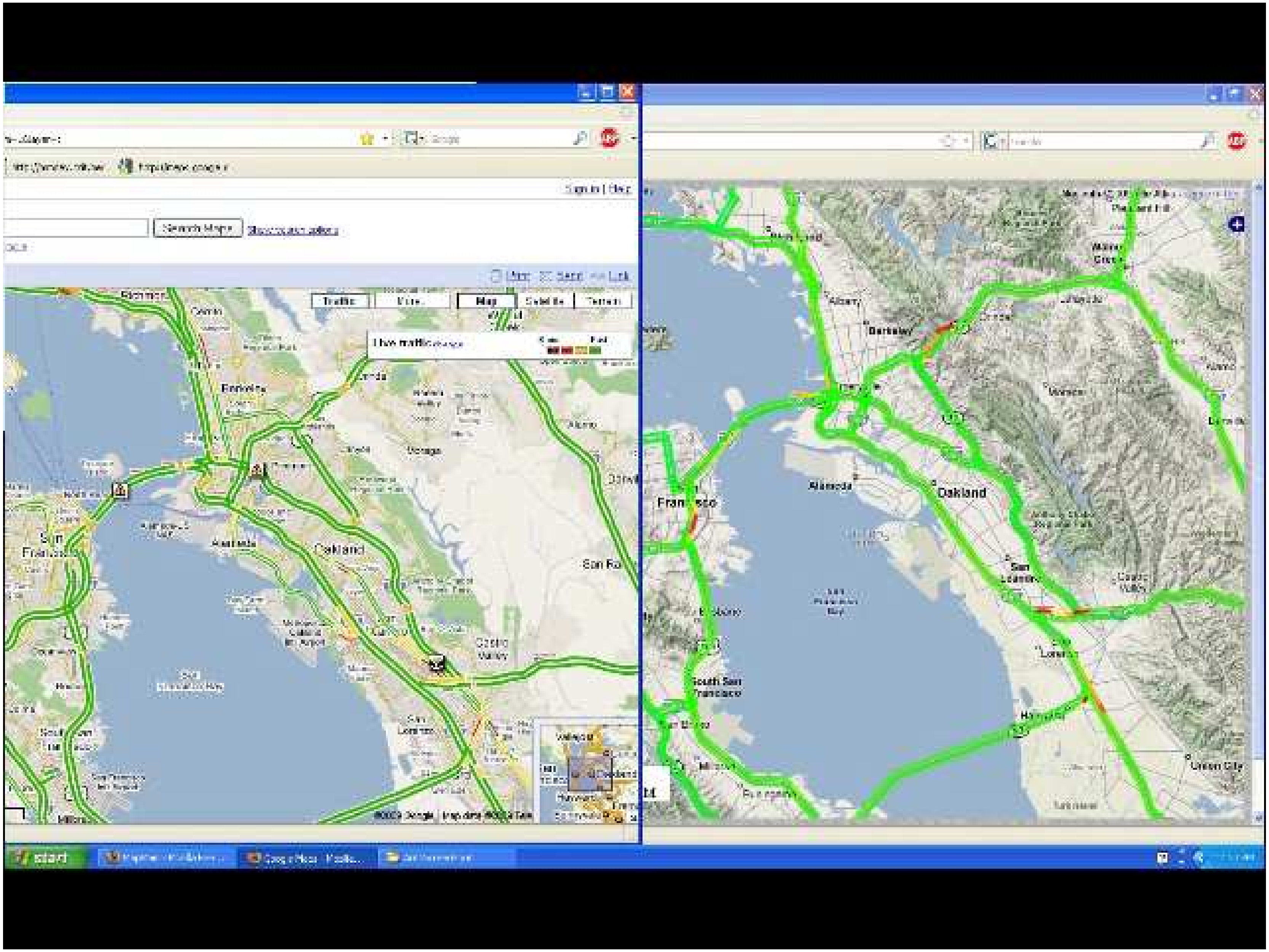
Google Maps



Mobile Millennium



Google Maps vs. HJ PDE



Outline

Inverse modeling problems

- **Problem description**
- **Introductory example**

Inverse modeling problems involving Hamilton-Jacobi equations

- **Background**
- **Problem definition**
- **Optimization formulations for inverse modeling**
- **Parameter uncertainty**

Sensing applications in traffic flow

- **Fault detection in the PeMS sensor network**
- **Other applications: racing with Google Maps**

Conclusion

Conclusion

Transformation of an estimation problem involving a nonsmooth, nonlinear PDE into a convex optimization problem: efficient way of integrating the model constraints into the estimation problem

Can be used for different applications (not showed here), including:

- data assimilation

- data reconciliation

- security analysis

- user privacy analysis