Hamilton-Jacobi formulation of first order scalar conservation laws

Christian Claudel

Assistant professor, Civil, Architectural and Environmental Engineering

University of Texas, Austin

Outline

Primitive flow models based on LWR

- The LWR PDE
- Integral formulation and physical interpretation
- Isolines of the Moskowitz function

Solution methods based on Viability theory

- Viability formulation
- Properties of the viability solution
- Semi-analytic formulation (exact solution)

Discussion

- Comparison with VT
- GSOM models

Conclusion

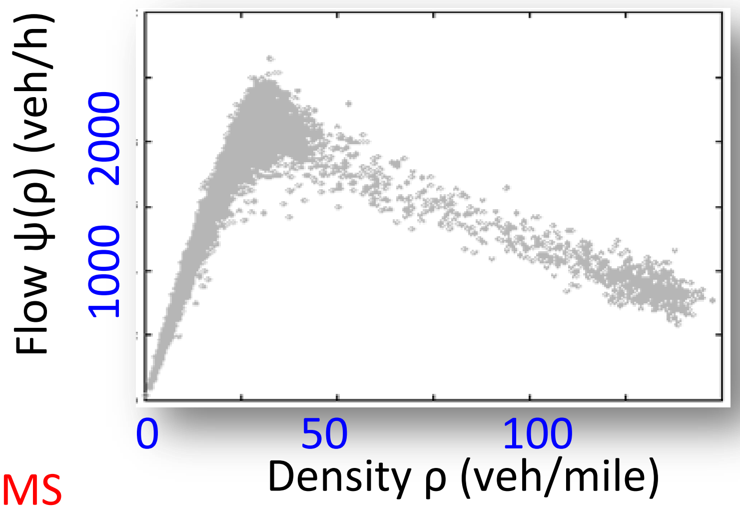
The LWR PDE

- First derived by Lighthill-Whitham (1955), and extended by Richards (1956)
- First order scalar hyperbolic conservation law

$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial \psi(\rho(t,x))}{\partial x} = 0$$

- Based on the conservation of vehicles, and on the existence of a relationship between flow and density: q=ψ(ρ).

ψ(.) is assumed to be concave



[Newell 93], [Daganzo 03,06]

Source: PeMS

Integral formulation

Equivalently, we can define M(t,x) such that:

$$\mathbf{M}(t_2, x_2) - \mathbf{M}(t_1, x_1) = \int_{x_1}^{x_2} -\rho(t_1, x)dx + \int_{t_1}^{t_2} \psi(\rho(t, x_2))dt$$

The function M(t,x) is called *Moskowitz function*. Its spatial derivative is the opposite of the density function; its temporal derivative is the flow function.

M(t,x) satisfies the following Hamilton-Jacobi PDE:

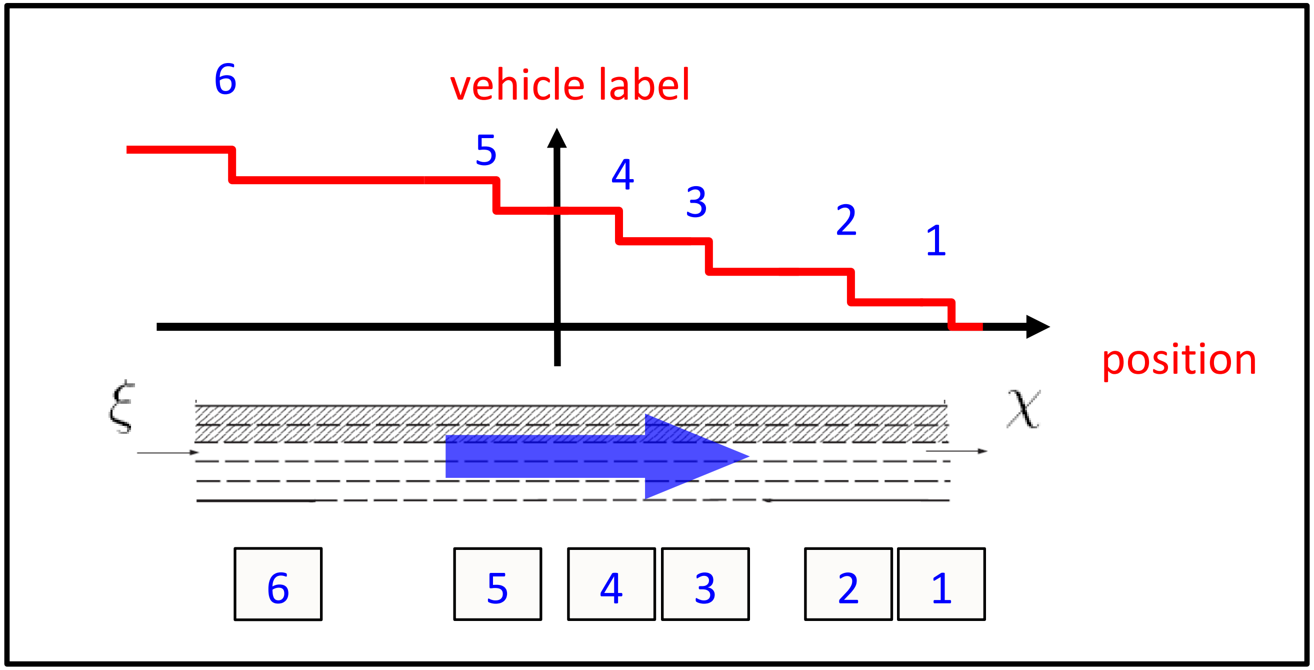
$$\frac{\partial \mathbf{M}(t,x)}{\partial t} - \psi\left(-\frac{\partial \mathbf{M}(t,x)}{\partial x}\right) = 0$$

[Newell 93], [Daganzo 03,06]

Physical interpretation of the Moskowitz function

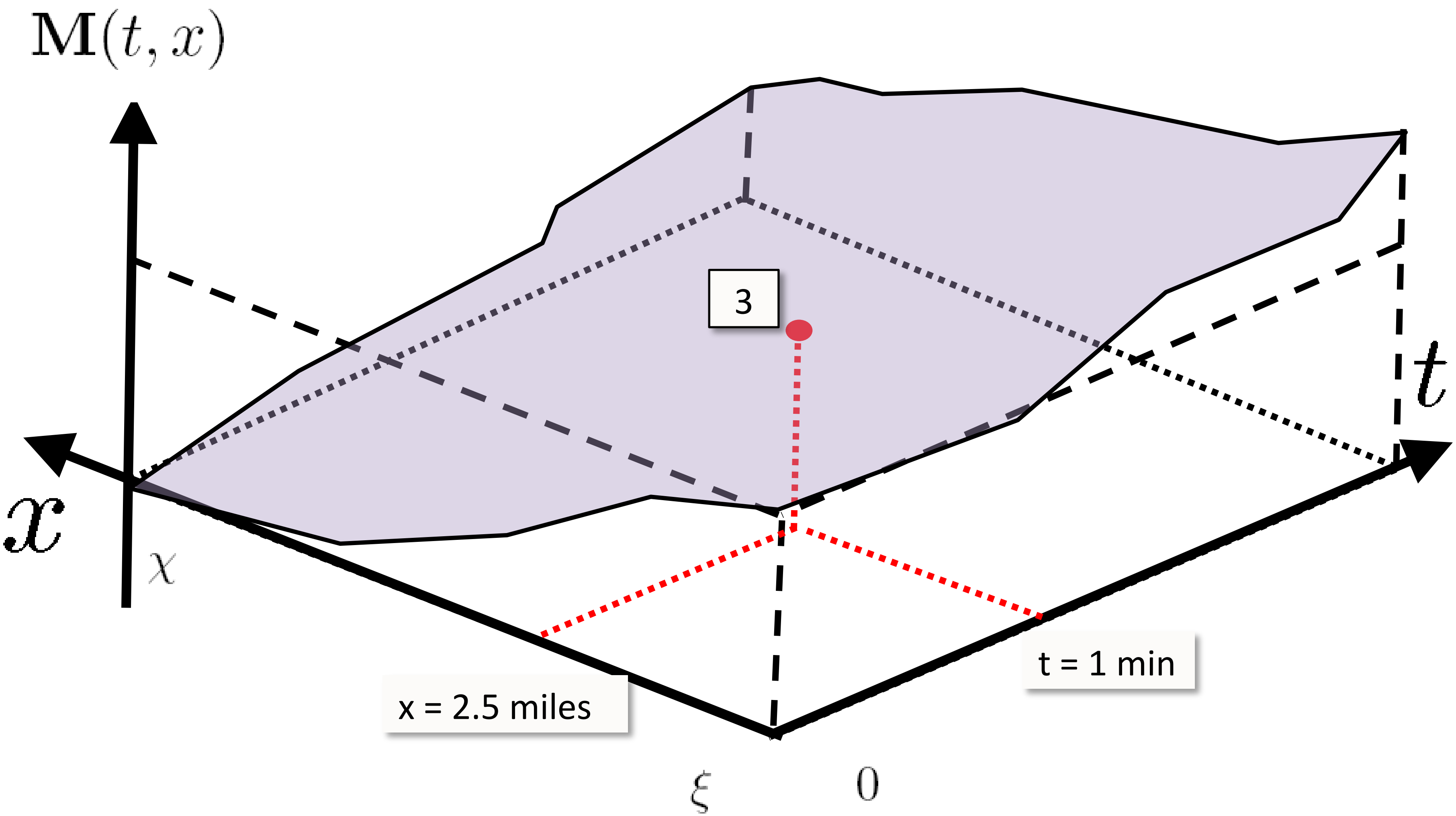
M(t,x) can be interpreted as a vehicle label at location x and time t (assuming that no vehicles pass each other).

M(t,x) is also known as the *cumulative vehicle number* in the traffic flow community.

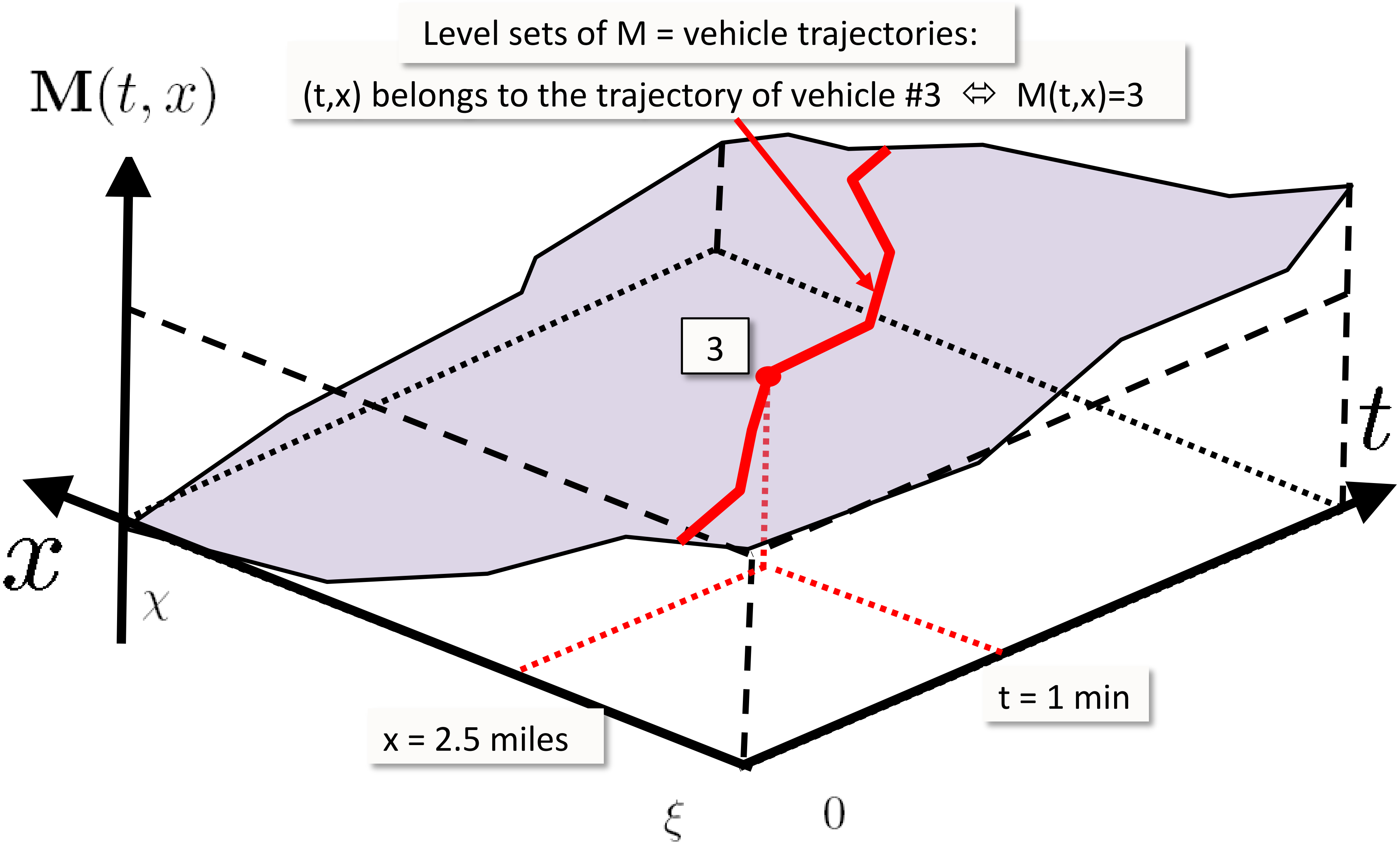


[Newell 93], [Daganzo 03,06]

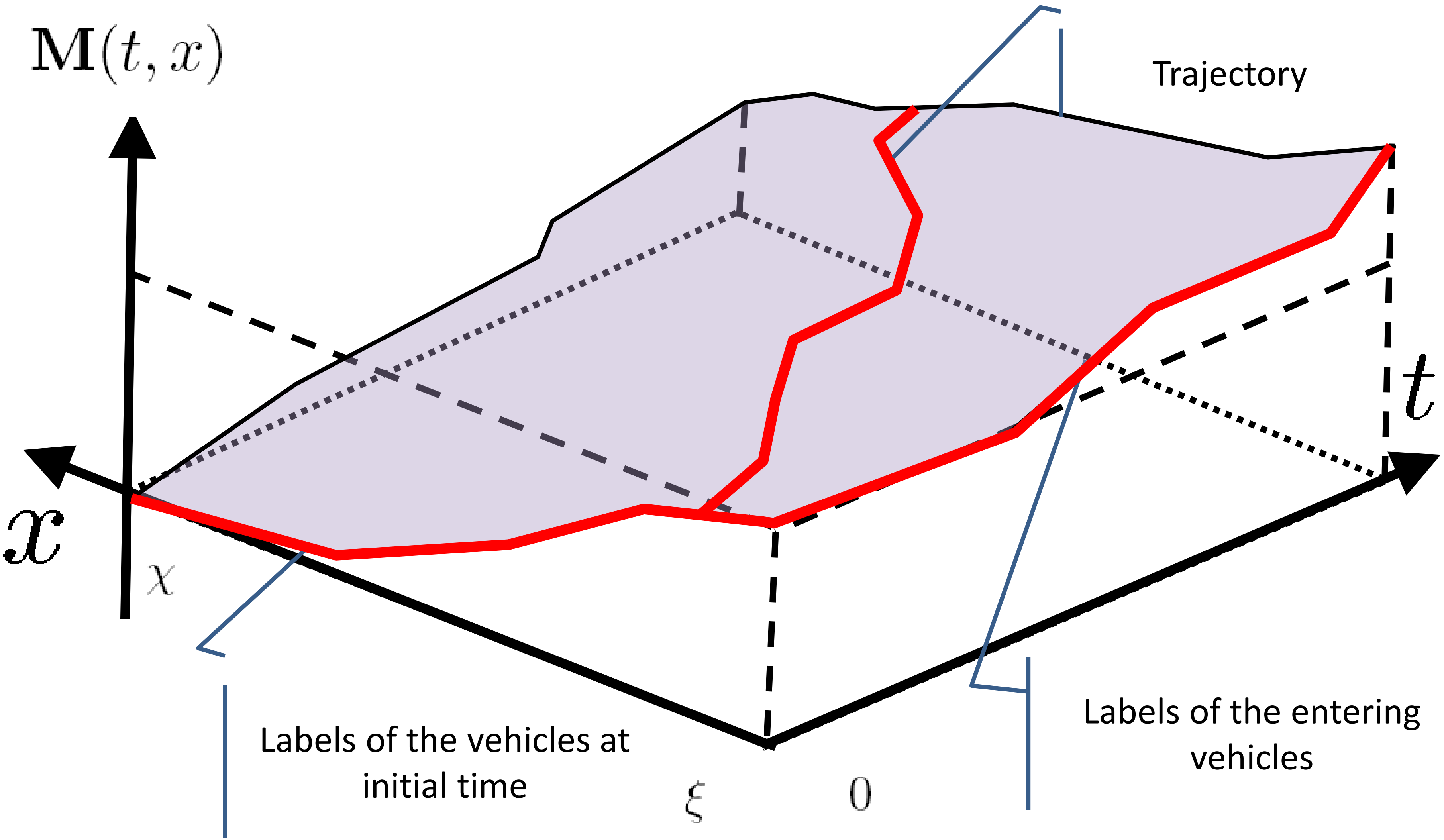
Physical interpretation of the Moskowitz function



Level sets of M



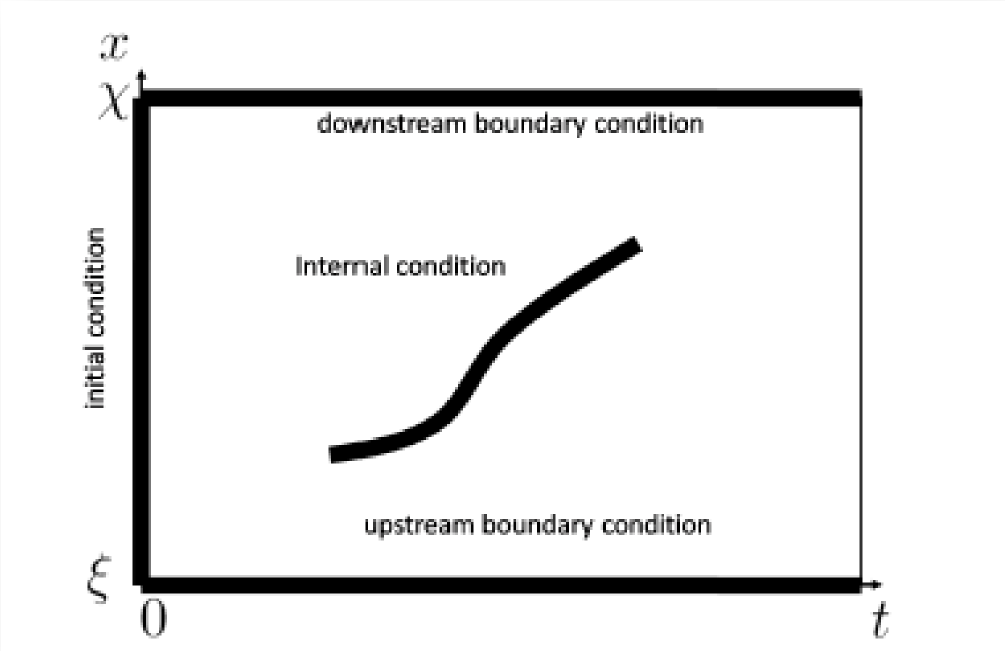
Value conditions: extension of initial/boundary conditions



Value conditions

The Moskowitz formulation is particularly suited to Lagrangian problems: when no passing is allowed, vehicles keep their labels, which enables the definition of internal conditions (trajectory constraints that apply to the solution)

Defining internal conditions for the LWR PDE is possible, but not as easy.



[Newell 93], [Daganzo 03,06]

Outline

TEXAS
The University of Texas at Austin

Primitive flow models based on LWR

- The LWR PDE
- Integral formulation and physical interpretation
- Isolines of the Moskowitz function

Solution methods based on Viability theory

- Viability formulation
- Properties of the viability solution
- Semi-analytic formulation (exact solution)

Discussion

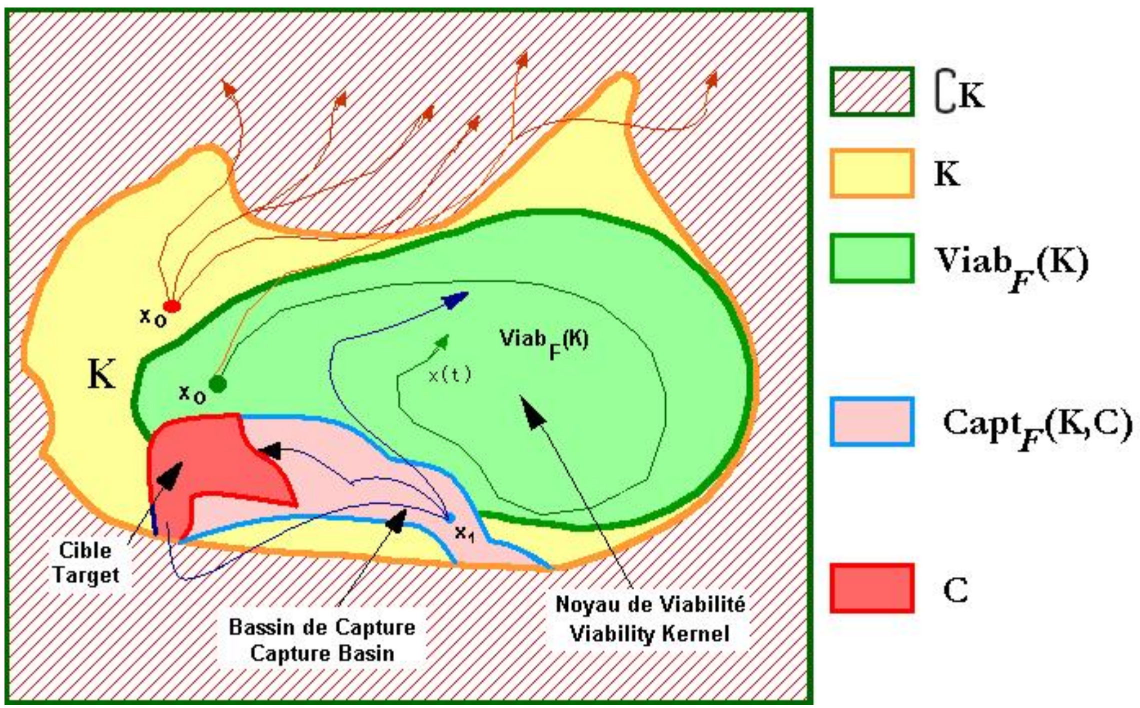
- Comparison with VT
- GSOM models

Conclusion

Solution methods for HJ PDEs

Different solution methods exist for solving HJ PDEs, including finite difference schemes (Lax-Friedrichs…), dynamic programming, level-set methods and semi-analytic schemes.

We focus here on the control framework of Viability theory



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

[Aubin 91] [Aubin Bayen Saint Pierre 2008] [Cardaliaguet, Quincampoix, Saint Pierre 99]

Capture basin formulation

Convex transform of ψ:

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

Characteristic system:

$$F := \begin{cases} \tau'(t) \ = \ -1 \\ x'(t) \ = \ u(t) \\ y'(t) \ = \ -\varphi^*(u(t)) \end{cases} \quad \text{where } u(t) \ \in \ \mathrm{Dom}(\varphi^*)$$

Environment and target sets:

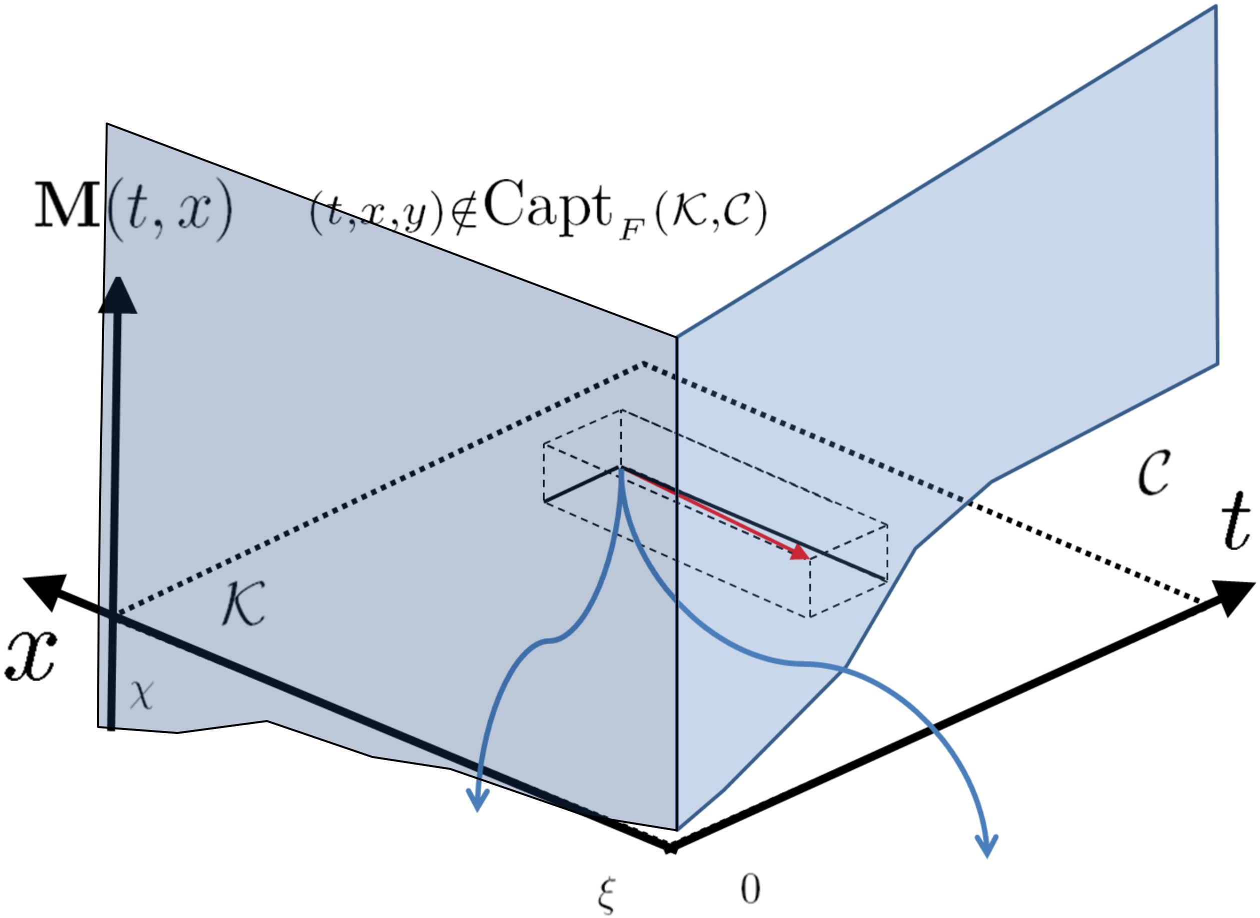
$$\mathcal{K} := \mathbb{R}_+ \times [\xi, \chi] \times \mathbb{R}$$

c represents a value condition (lower-semicontinuous)

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

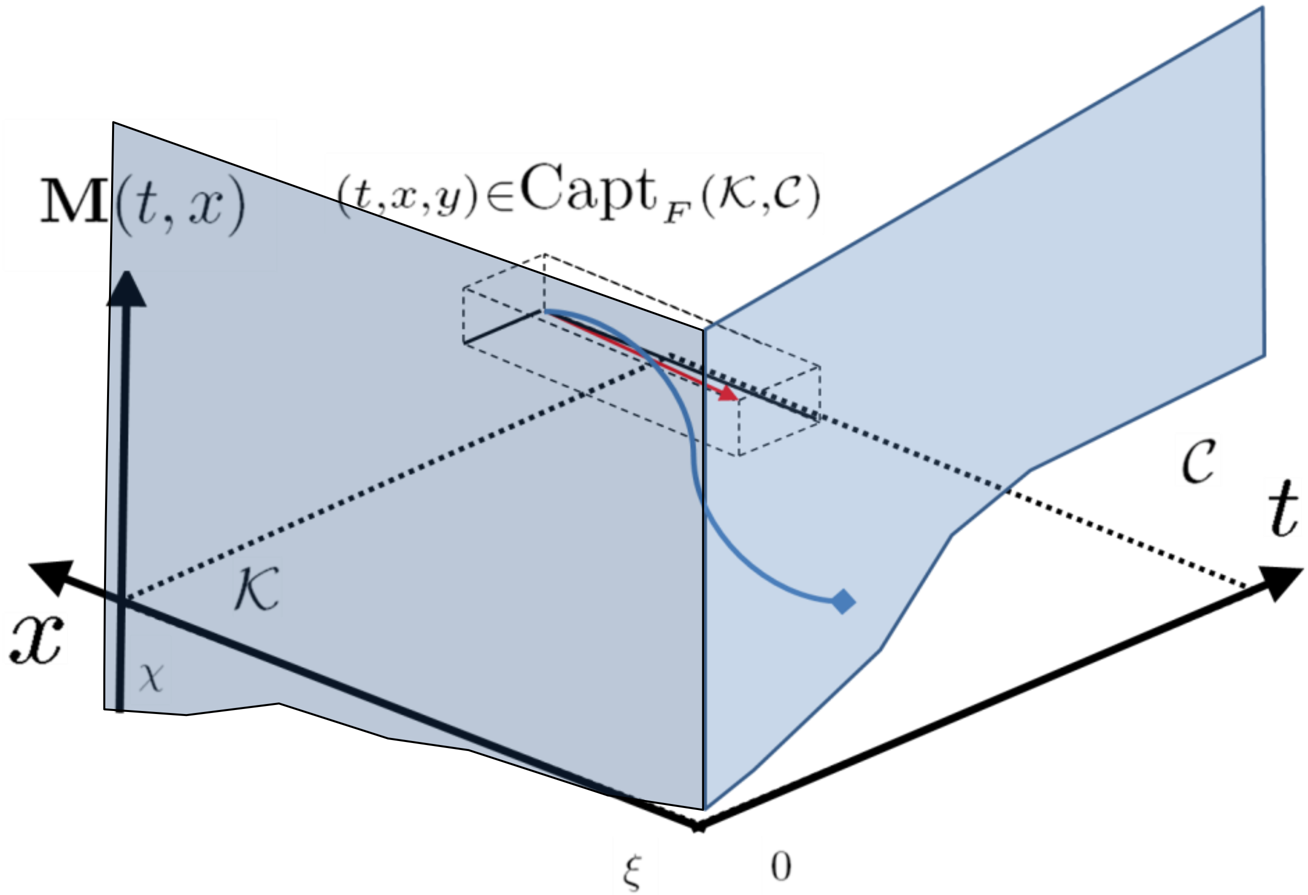
[Aubin, Bayen, Saint Pierre, SIAM 08]

Capture basin formulation



[Aubin, Bayen, Saint Pierre, SIAM 08]

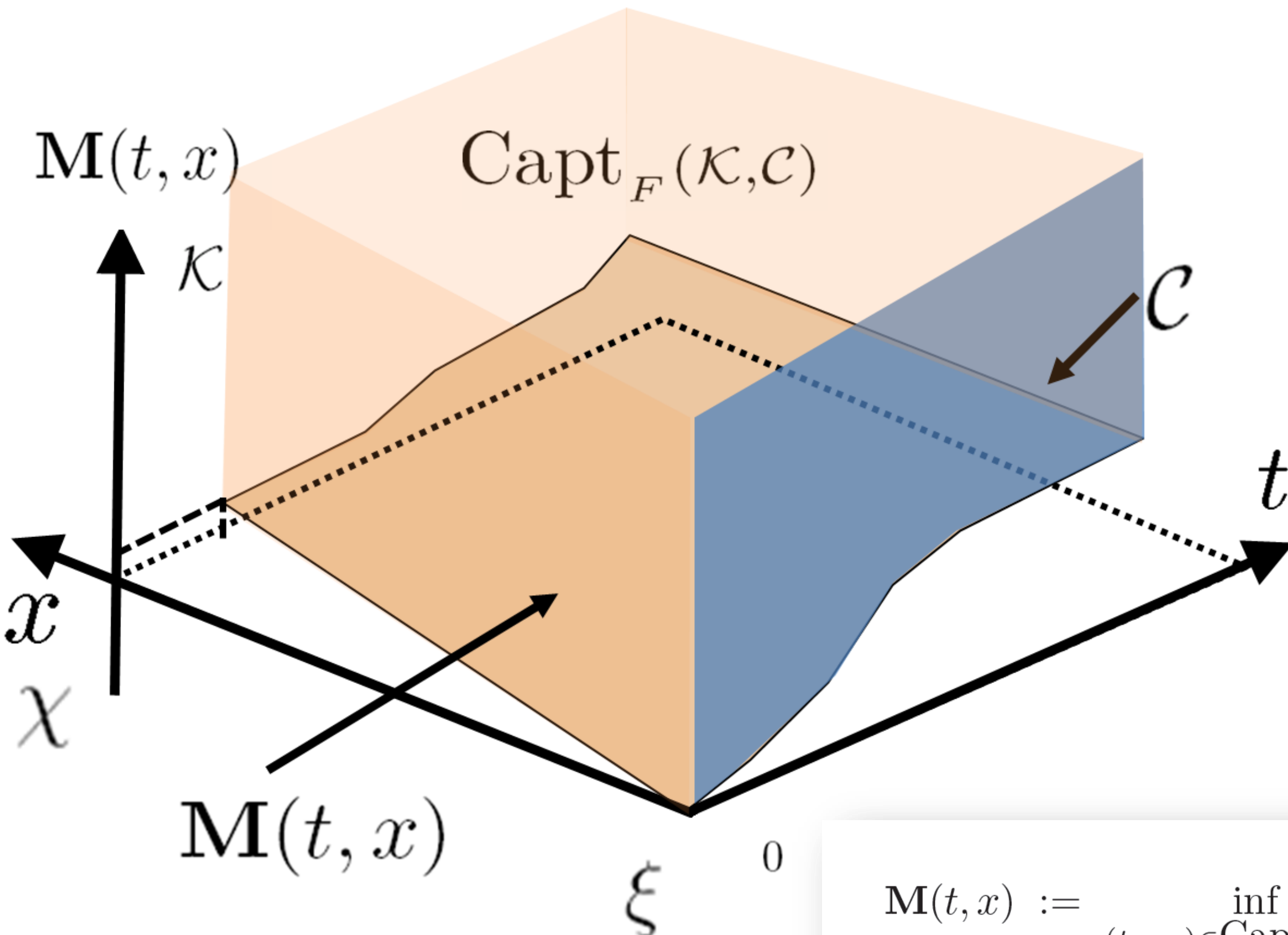
Capture basin formulation



[Aubin, Bayen, Saint Pierre, SIAM 08]

Capture basin formulation

Solution (also known as Viability episolution): lower envelope of the Capture Basin



$$\mathbf{M}(t,x) := \inf_{(t,x,y)\in \mathrm{Capt}_F(\mathcal{K},\mathcal{C})} y$$

[Aubin, Bayen, Saint Pierre, SIAM 08]

Properties: Lax-Hopf formula

There exists an implicit expression of the solution (Lax-Hopf formula)

This formula can be obtained by using

$$\mathbf{M}(t,x) := \inf_{(t,x,y)\in \mathrm{Capt}_F(\mathcal{K},\mathcal{C})} y$$

$$\mathbf{M_c}(t,x) = \inf_{(u(\cdot),T,y)\in R} \left(\mathbf{c}(t-T, x + \int_0^T u(\tau)d\tau) + \int_0^T \varphi^*(u(\tau)) \right)$$

using some properties of the dynamical system: the optimal trajectories of the dynamical system are straight lines (using Jensen's inequality):

$$\mathbf{M_c}(t,x) = \inf_{(u,T)\in \mathrm{Dom}(\varphi^*)\times\mathbb{R}_+} (\mathbf{c}(t-T, x+Tu) + T\varphi^*(u))$$

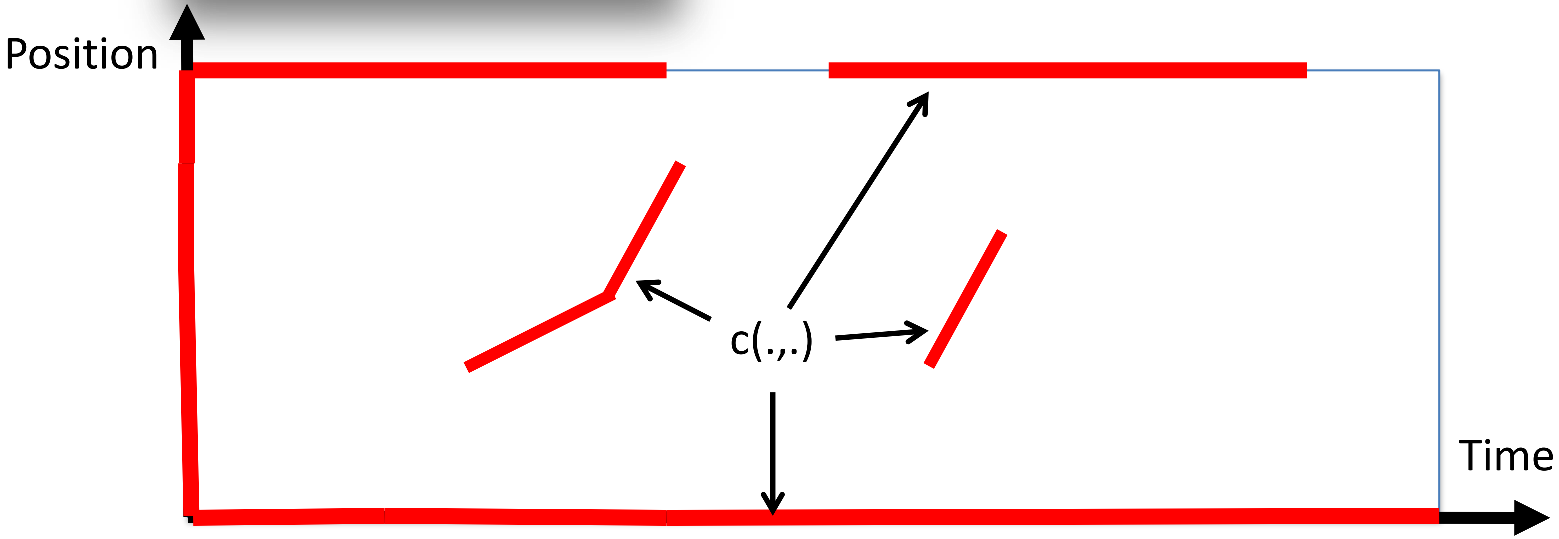
[Daganzo 2005 TR-B] [Daganzo 2006] [Claudel Bayen TAC 2010]

Lax-Hopf formula

For a boundary data function c(.,.), the solution M_c(.,.) is given by:

$$\mathbf{M_c}(t,x) = \inf_{(u,T)\in \mathrm{Dom}(\varphi^*)\times\mathbb{R}_+} \left(\mathbf{c}(t-T, x+Tu) + T\varphi^*(u)\right)$$

where $\varphi^*(u) := \sup_{p\in\mathrm{Dom}(\psi)} [p\cdot u + \psi(p)]$ is the convex transform of ψ

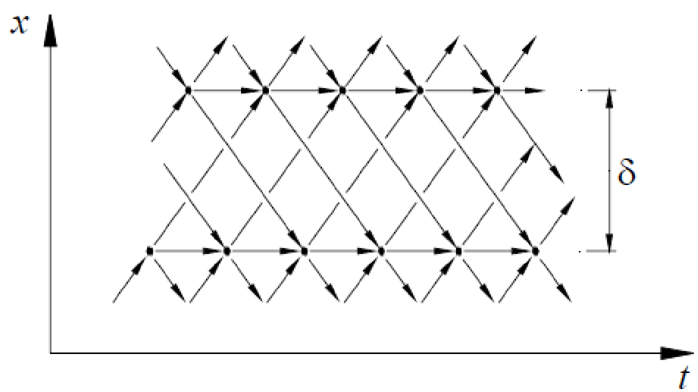


[Lax 1973] [Aubin Bayen Saint Pierre SIAM SICON 2009] [Claudel Bayen IEEE TAC part II 2010]

Variational theory

The Variational Theory is a possible computational method obtained by solving the Lax Hopf formula on a discretized grid, using dynamic programming:

$$\mathbf{M_c}(t,x) = \inf_{(u,T)\in \mathrm{Dom}(\varphi^*)\times\mathbb{R}_+} \left(\mathbf{c}(t-T, x+Tu) + T\varphi^*(u)\right)$$



The Viability Algorithm has a similar structure, but furthermore allows the computation of solutions that have lower state constraints.

[Daganzo 2005, Daganzo 2006]

Inf-morphism

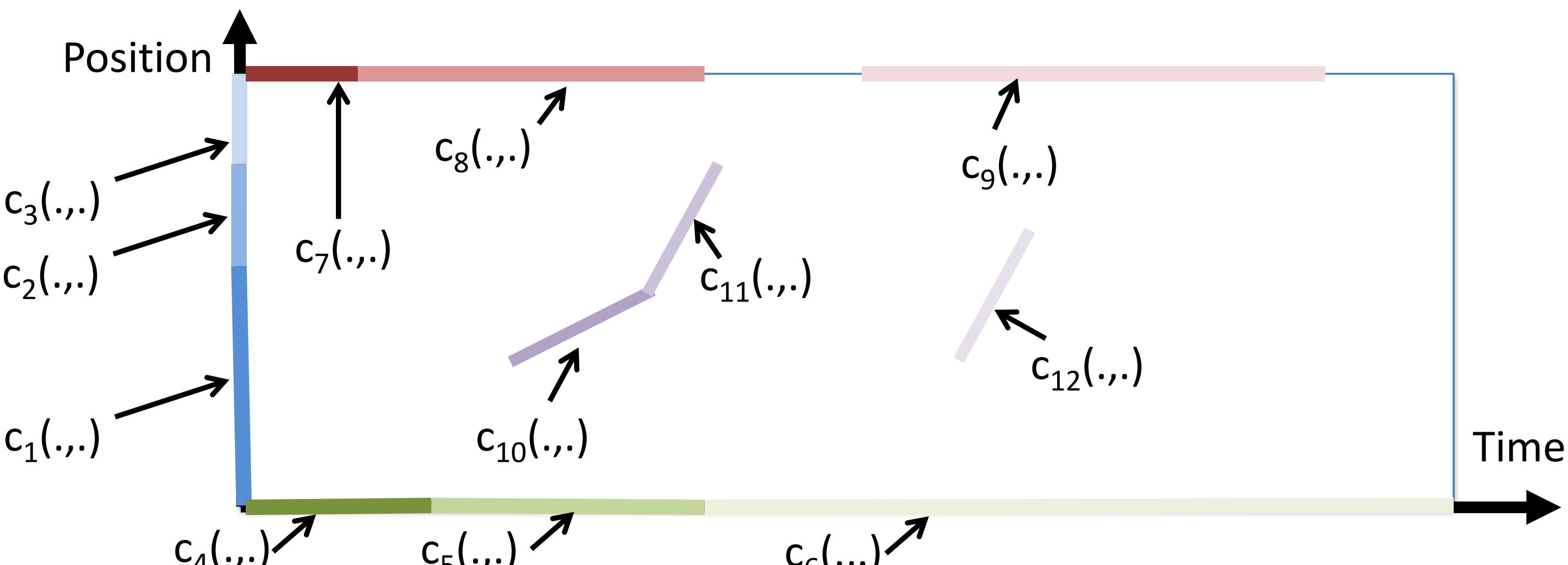
- <u>Inf-morphism property</u>

Let us assume that the boundary data c is the minimum of a finite number of lower semicontinuous functions:

$$\forall (t,x) \in [0,t_{\max}] \times [x, c], \; c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0,t_{\max}] \times [x, c], \; M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$

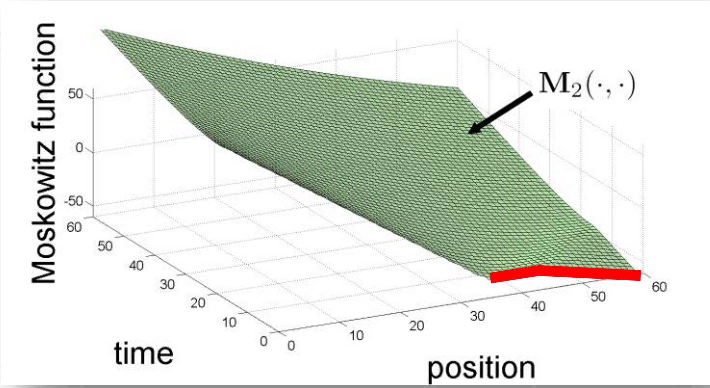


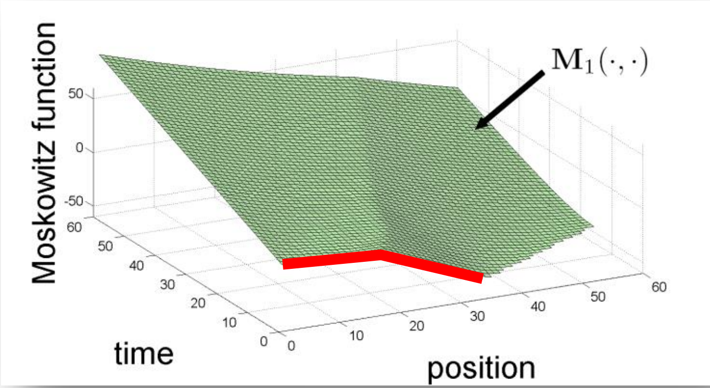
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

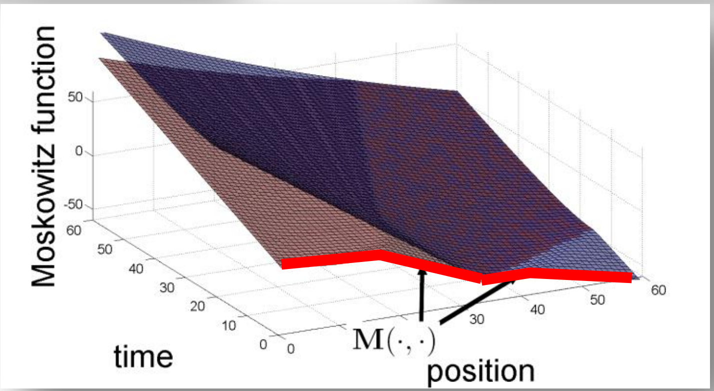
Physical interpretation

- <u>Inf-morphism property:</u>

  The solution associated with a set of boundary conditions is the minimum of the solutions associated with each boundary condition.





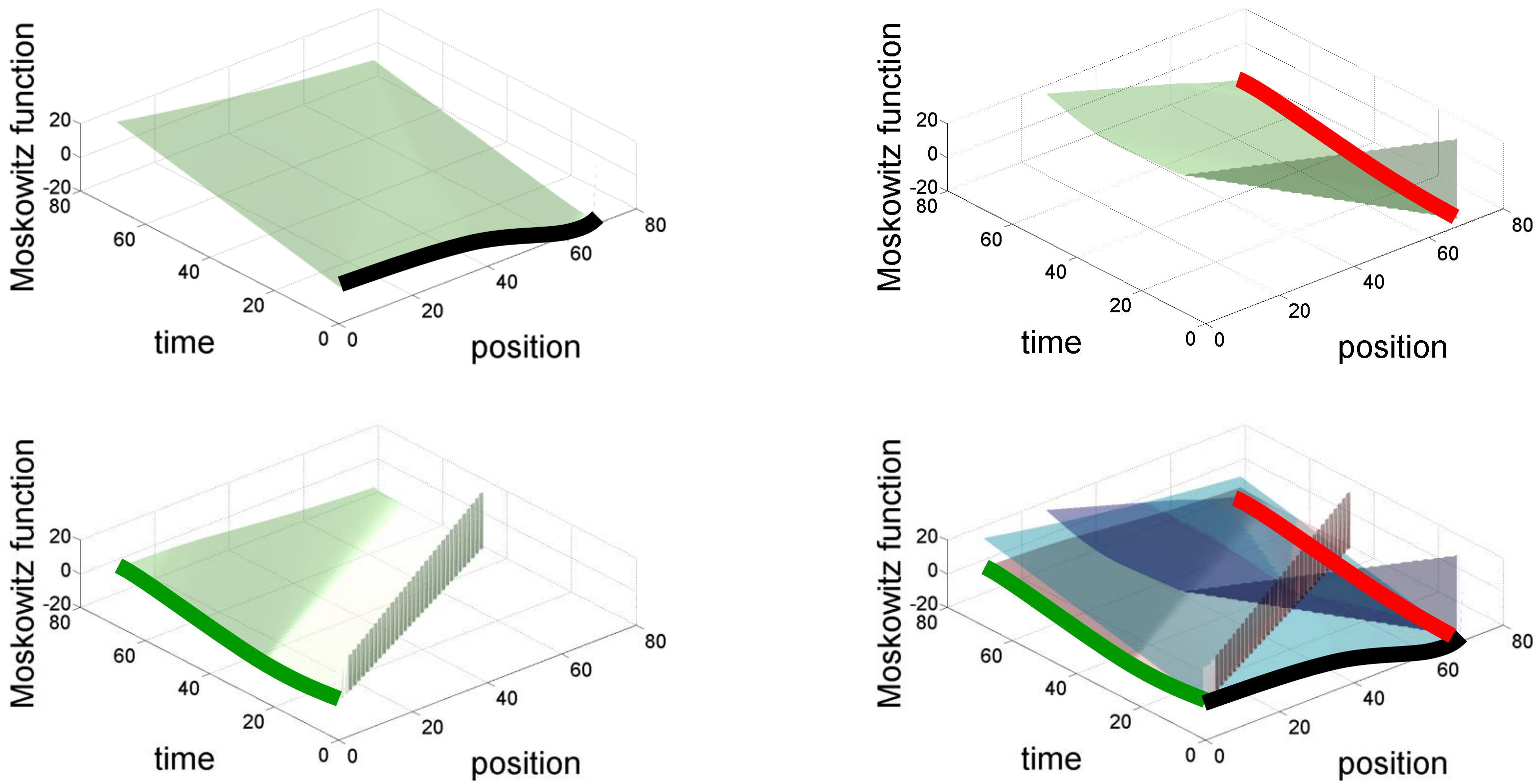


[Newell 93] [Daganzo 06] [Aubin Bayen Saint-Pierre 07] [Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Physical interpretation

- <u>Inf-morphism property:</u>

  The solution associated with a set of boundary conditions is the minimum of the solutions associated with each boundary condition.



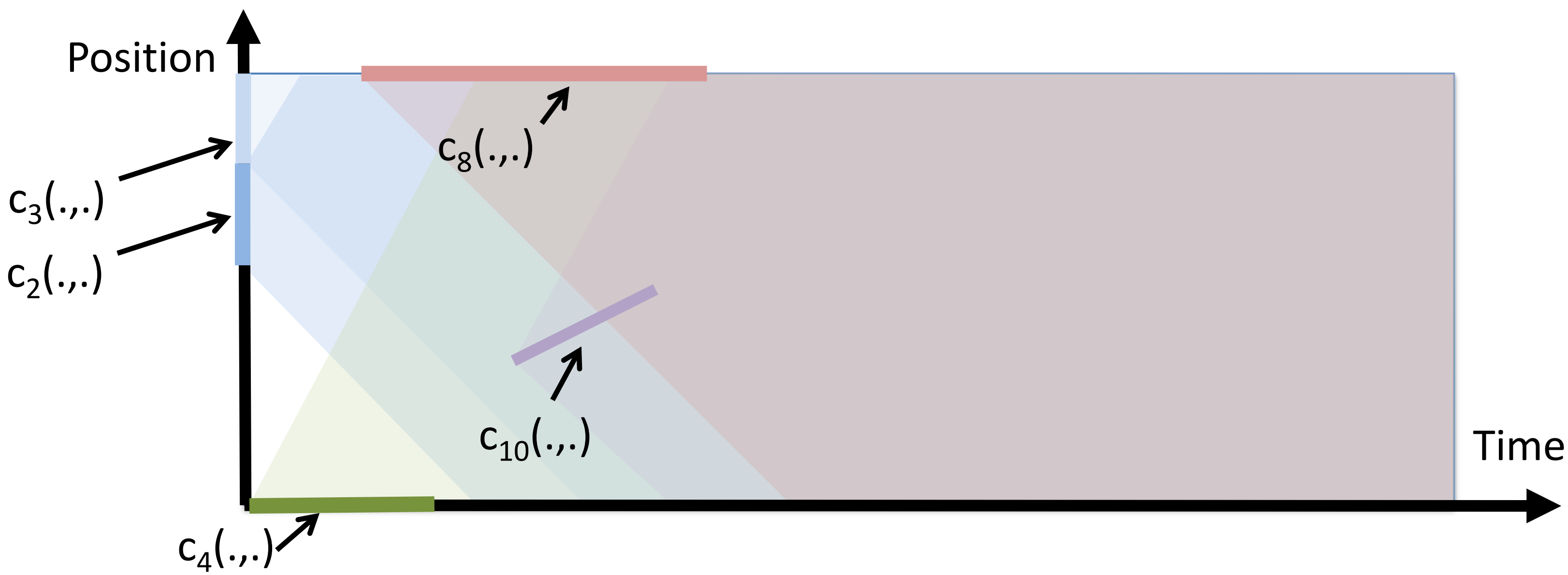
[Newell 93] [Daganzo 06] [Aubin Bayen Saint-Pierre 07]
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Semi-analytic computational methods

- <u>Inf-morphism property</u>

Let us assume that the boundary data c is the minimum of a finite number of lower semicontinuous functions: $\boxed{\forall (t,x) \in [0,t_{\max}] \times [\xi, \chi], c(t,x) := \min_{j \in J} c_j(t,x)}$

The solution associated with the above boundary data function can be decomposed as: $\boxed{\forall (t,x) \in [0,t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)}$



[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Semi-analytic computational methods

- <u>Inf-morphism property</u>

Let us assume that the boundary data c is the minimum of a finite number of lower semicontinuous functions: $\forall \ (t,x) \in [0,t_{\max}] \times [x, c], c(t,x) := \min_{j \in J} c_j(t,x)$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0,t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$

If $c_j(.,.)$ is convex, so is $M_{c_j}(.,.)$

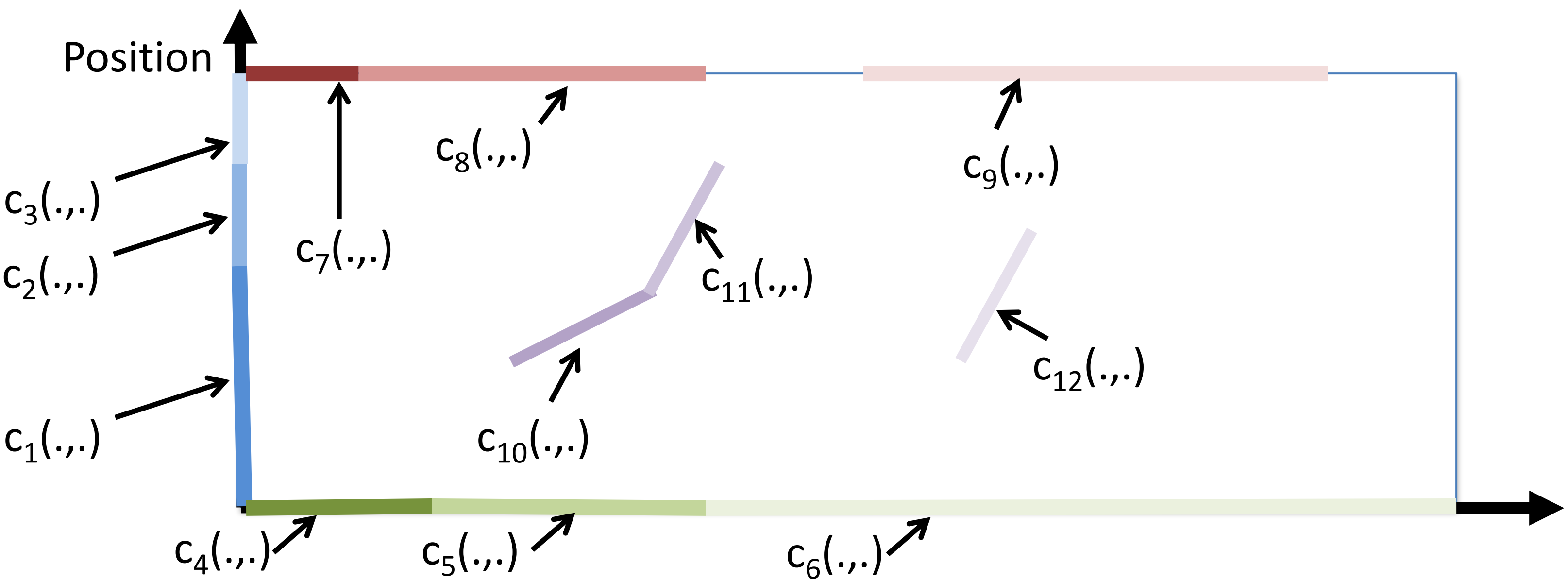
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Semi-analytic computational methods

- <u>Inf-morphism property</u>

The solution associated with the above boundary data function can be decomposed as: $\boxed{\forall\ (t,x) \in [0,t_{\max}] \times [x, c], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)}$

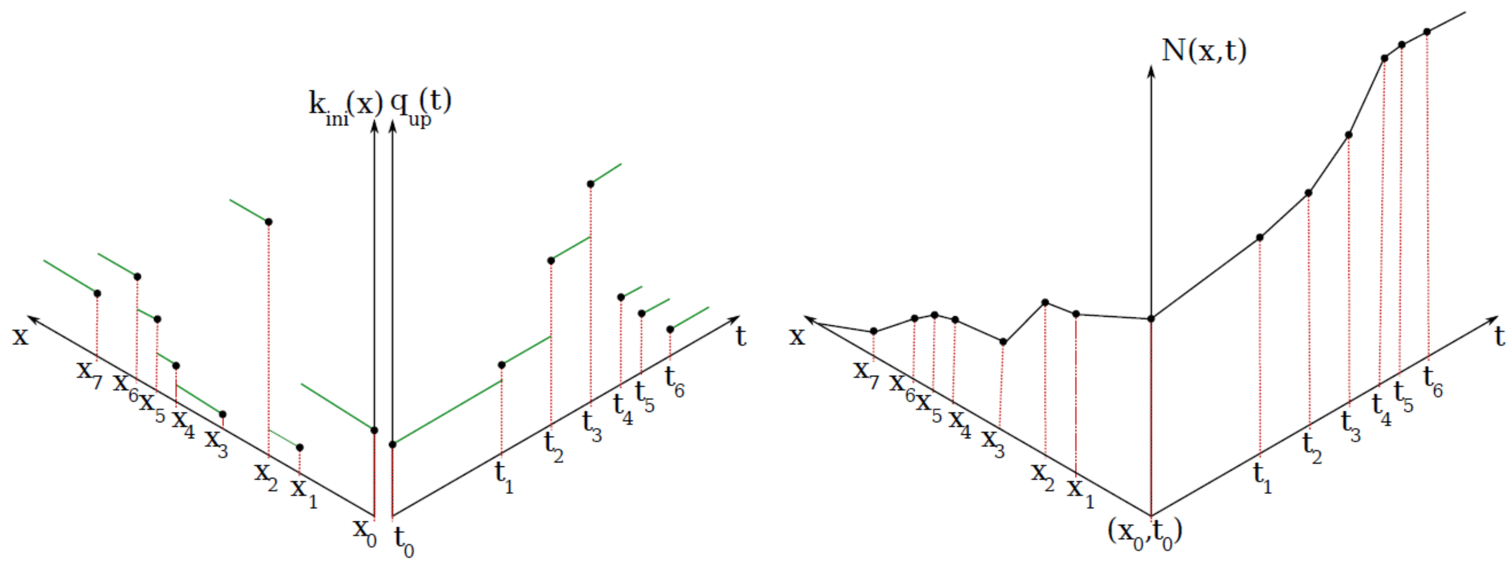
If $c_j(.,.)$ is linear, then $M_{c_j}(.,.)$ can be explicitly computed (solution to a 1D convex optimization problem.



[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Semi-analytic formulation

Most of the time, we consider piecewise constant initial and boundary conditions for the LWR PDE. This translates into piecewise affine initial and boundary conditions for the HJ PDE.



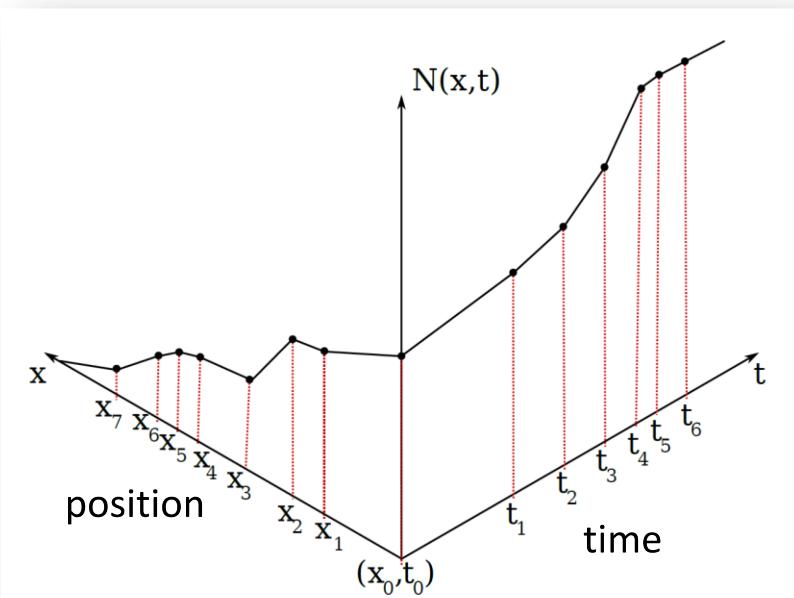
[Daganzo 2005 TR-B] [Daganzo 2006] [Mazare Dehwah Claudel Bayen 2012]

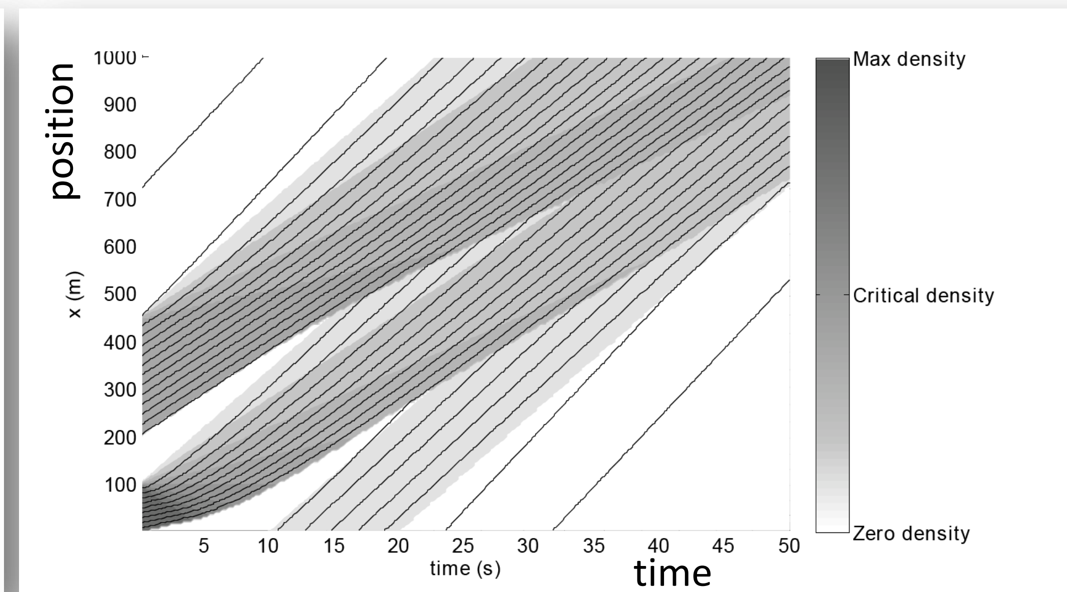
Semi-analytic computational methods

$$\forall (t,x) \in [0,t_{\max}] \times [\xi, \chi], c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0,t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$





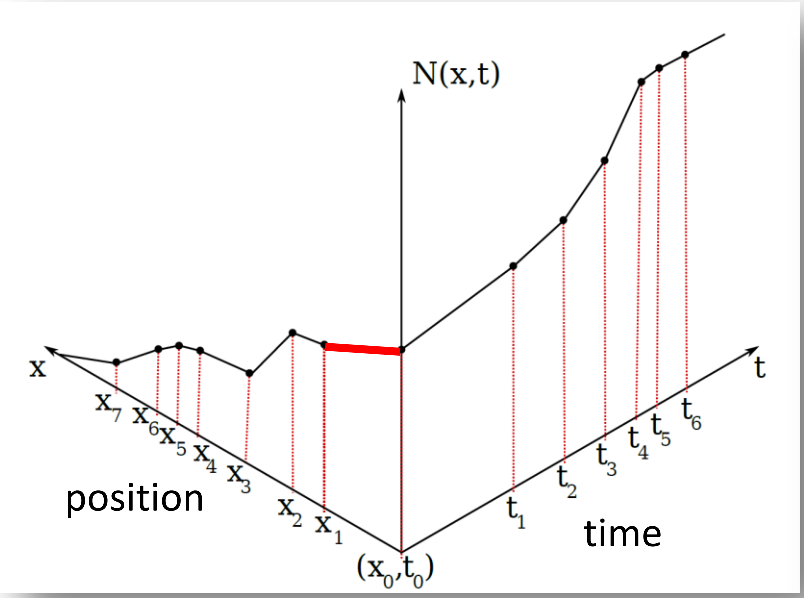
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

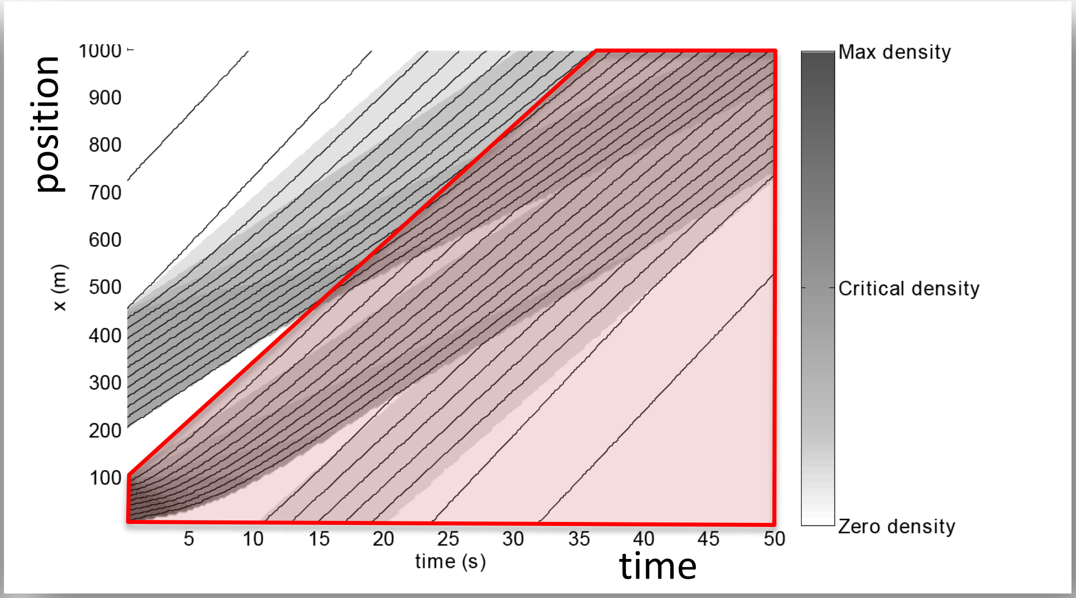
Semi-analytic computational methods

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$





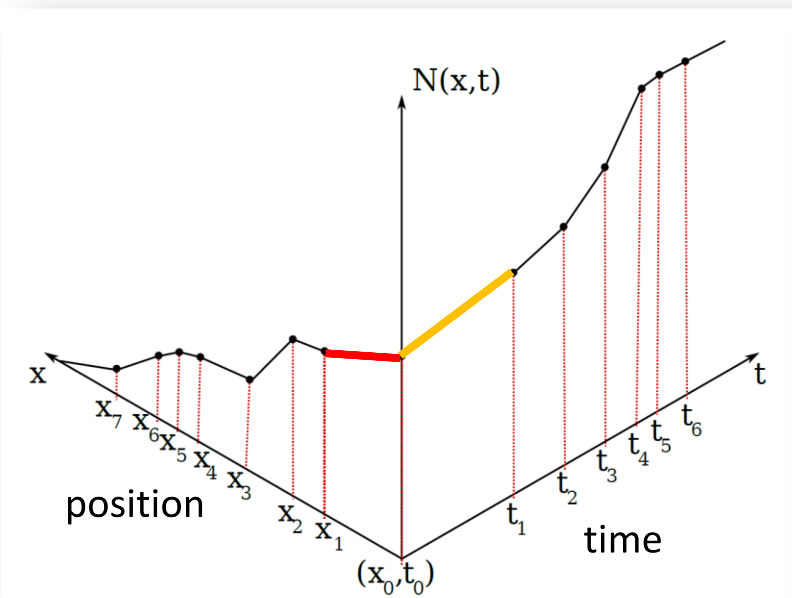
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

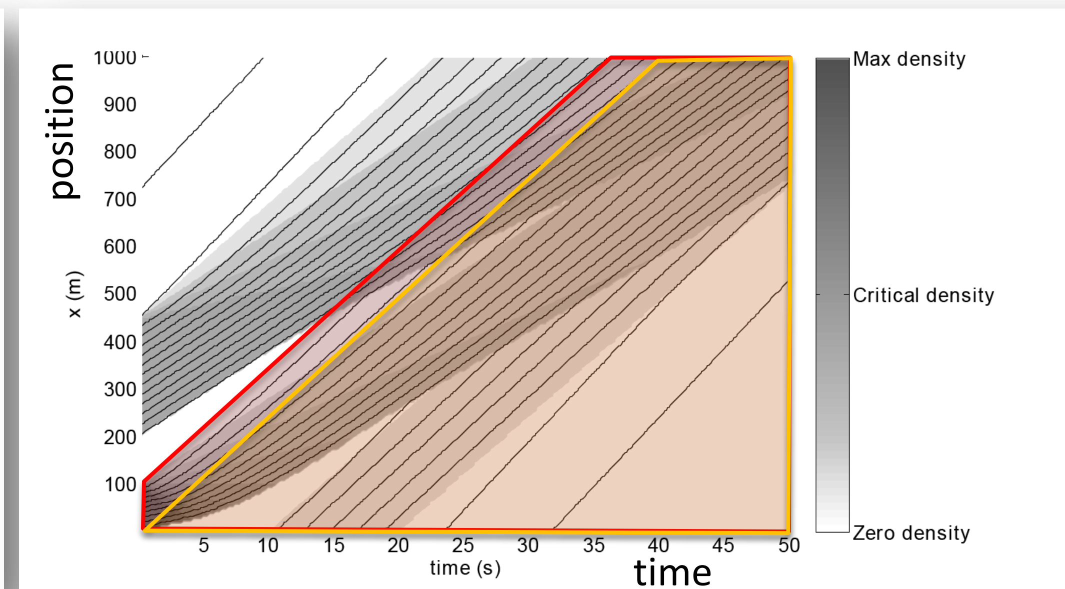
Semi-analytic computational methods

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$





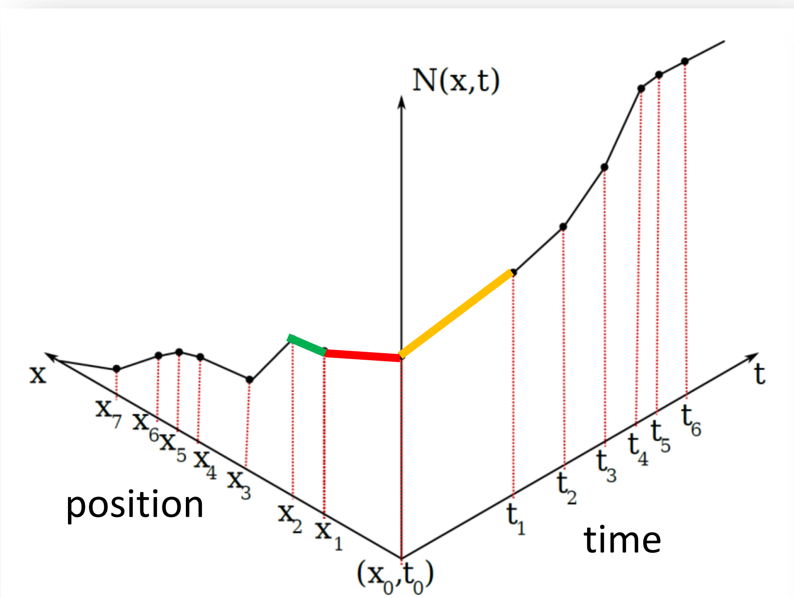
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

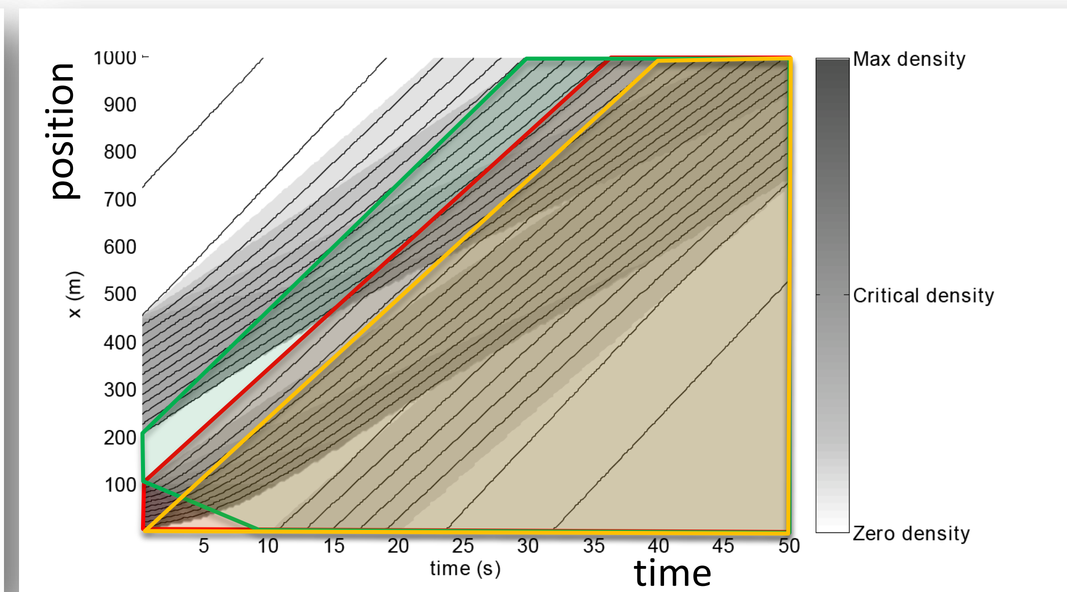
Semi-analytic computational methods

$$\forall\ (t,x) \in [0,t_{\max}] \times [\xi,\chi],\ c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall\ (t,x) \in [0,t_{\max}] \times [\xi,\chi],\ M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$





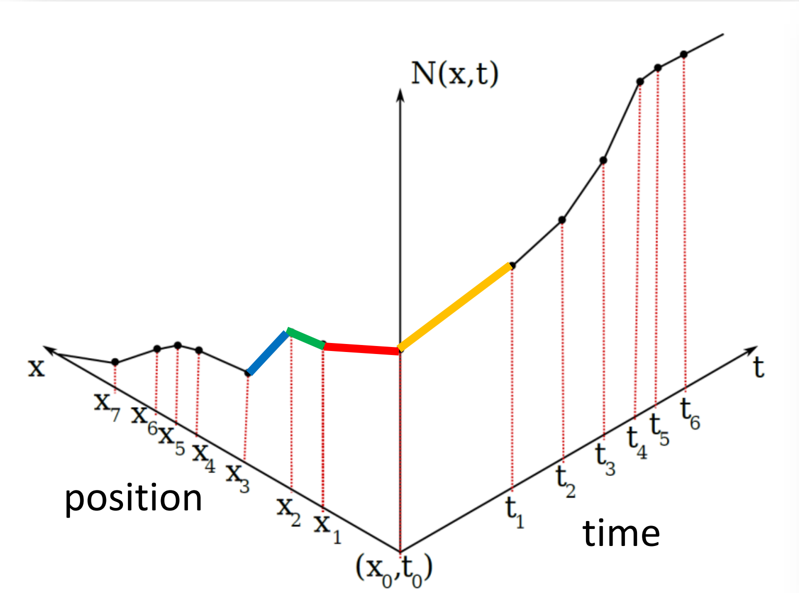
[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

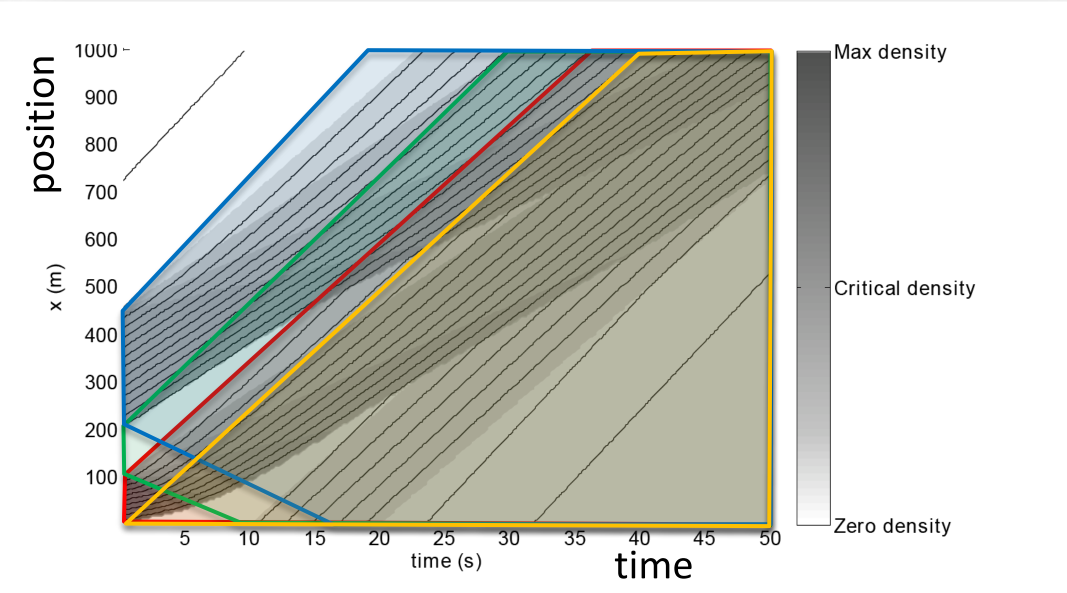
Semi-analytic computational methods

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], c(t,x) := \min_{j \in J} c_j(t,x)$$

The solution associated with the above boundary data function can be decomposed as:

$$\forall (t,x) \in [0, t_{\max}] \times [\xi, \chi], M_C(t,x) = \min_{j \in J} M_{c_j}(t,x)$$





[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Solutions to affine value conditions

The question becomes: can we quickly compute the solution associated with an affine (linear) initial or boundary condition?

To be able to use the inf morphism property in general, we need each affine initial or boundary condition to be defined on a subset of $\mathbb{R}^2$

Non physical/mathematical problem

[Daganzo 2005 TR-B] [Daganzo 2006] [Claudel Bayen TAC 2010]

Solutions to affine value conditions

Affine initial condition:

$$\mathcal{M}_{0,i}(0,x) = \begin{cases} a_i x + b_i & \text{if } x \in [\overline{\alpha}_i, \overline{\alpha}_{i+1}] \\ +\infty & \text{otherwise} \end{cases}$$

Affine upstream boundary condition

$$\gamma_j(t,\xi) = \begin{cases} c_j t + d_j & \text{if } t \in [\overline{\gamma}_j, \overline{\gamma}_{j+1}] \\ +\infty & \text{otherwise} \end{cases}$$

Affine downstream boundary condition

$$\beta_k(t,x) = \begin{cases} e_k t + f_k & \text{if } t \in [\overline{\beta}_k, \overline{\beta}_{k+1}] \\ +\infty & \text{otherwise} \end{cases}$$

[Claudel Bayen TAC 2010]

Solutions to affine value conditions

TEXAS
The University of Texas at Austin

Example of solution: affine initial condition

$$\mathcal{M}_{0,i}(0,x) = \begin{cases} a_i x + b_i & \text{if } x \in [\overline{\alpha}_i, \overline{\alpha}_{i+1}] \\ +\infty & \text{otherwise} \end{cases}$$

Lax-Hopf formula:

$$\mathrm{M}_{\mathcal{M}_{0,i}}(t,x) \;=\; \inf_{u \in \mathrm{Dom}(\varphi^*) \cap \left[\frac{\overline{\alpha}_i - x}{t}, \frac{\overline{\alpha}_{i+1} - x}{t}\right]} \left(a_i(x+tu) + b_i\right.$$
$$\left. + t\varphi^*(u)\right), \;\; \forall (t,x) \in \mathbb{R}_+^* \times X$$

Objective function: convex (in u):

$$\forall u \in \mathrm{Dom}(\varphi^*), \;\; \zeta_{a_i,b_i,t,x}(u) := a_i(x+tu) + b_i + t\varphi^*(u)$$

constraints also convex (intersection of intervals) : 1-D convex optimization.

[Claudel Bayen TAC 2010]

Solutions to affine value conditions

Explicit solution to the convex program requires the use of subgradients, since ϕ* is not necessarily differentiable.

$$
\begin{aligned}
&\forall u \in \mathrm{Dom}(\varphi^*),\\
&\partial_- \zeta_{a_i,b_i,t,x}(u) = \{w \mid \exists v \in \partial_- \varphi^*(u),\ \ w = a_i t + vt\}\\
&\qquad\qquad\qquad := t \cdot (\{a_i\} + \partial_- \varphi^*(u))
\end{aligned}
$$

Thus, posing $u_0(a_i)$ as any element of $-\partial_+ \psi(-a_i)$, we have the following explicit solution:

$$
\mathrm{M}_{\mathcal{M}_{0,i}}(t,x) = \begin{cases}
(i) & t\psi(-a_i) + a_i x + b_i \\
 & \text{if } \ u_0(a_i) \in [\frac{\overline{\alpha}_i - x}{t}, \frac{\overline{\alpha}_{i+1} - x}{t}] \\
(ii) & a_i \overline{\alpha}_i + b_i + t\varphi^*(\frac{\overline{\alpha}_i - x}{t}) \\
 & \text{if } \ u_0(a_i) \leq \frac{\overline{\alpha}_i - x}{t} \\
(iii) & a_i \overline{\alpha}_{i+1} + b_i + t\varphi^*(\frac{\overline{\alpha}_{i+1} - x}{t}) \\
 & \text{if } \ u_0(a_i) \geq \frac{\overline{\alpha}_{i+1} - x}{t}
\end{cases}
$$

[Claudel Bayen TAC 2010]

Semi-analytic computational methods

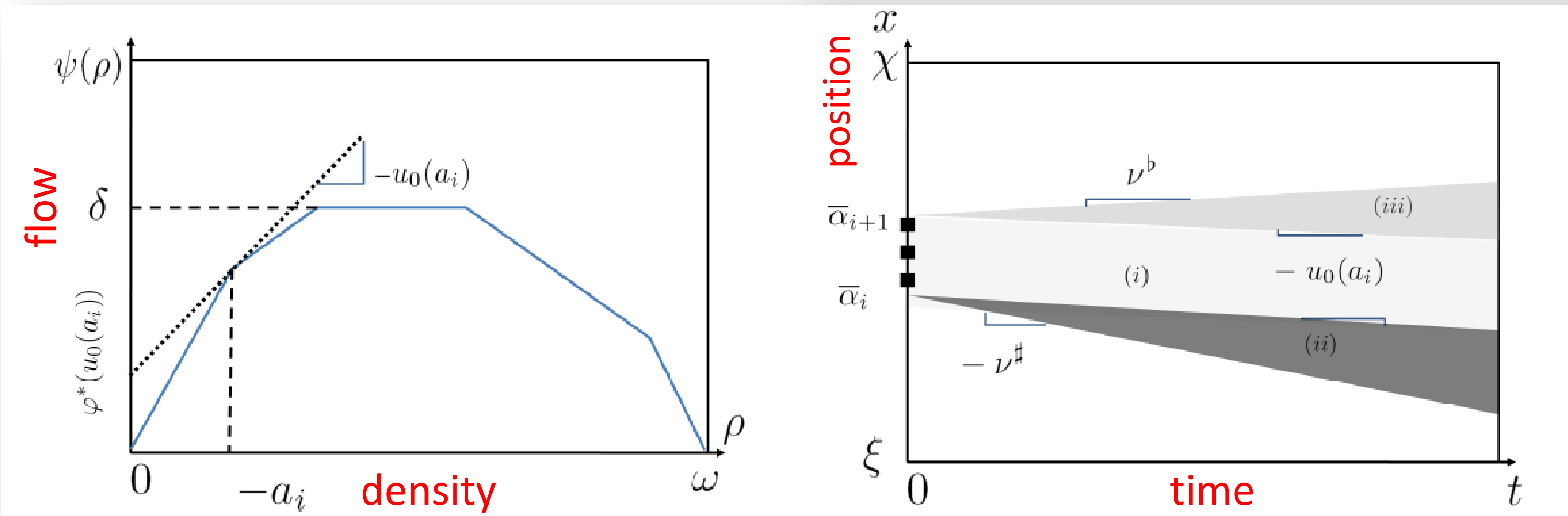
<u>Solution structure:</u>

Data

Model

$$u_0(a_i) \in -\partial_+ \psi(-a_i)$$

$$\mathbf{M}_{\mathcal{M}_0,i}(t,x) = \begin{cases} (i) & t\psi(-a_i) + a_i x + b_i \\ & \text{if } u_0(a_i) \in \left[\frac{\overline{\alpha}_i - x}{t}, \frac{\overline{\alpha}_{i+1} - x}{t}\right] \\ (ii) & a_i\overline{\alpha}_i + b_i + t\varphi^*\left(\frac{\overline{\alpha}_i - x}{t}\right) \\ & \text{if } u_0(a_i) \leq \frac{\overline{\alpha}_i - x}{t} \\ (iii) & a_i\overline{\alpha}_{i+1} + b_i + t\varphi^*\left(\frac{\overline{\alpha}_{i+1} - x}{t}\right) \\ & \text{if } u_0(a_i) \geq \frac{\overline{\alpha}_{i+1} - x}{t} \end{cases}$$

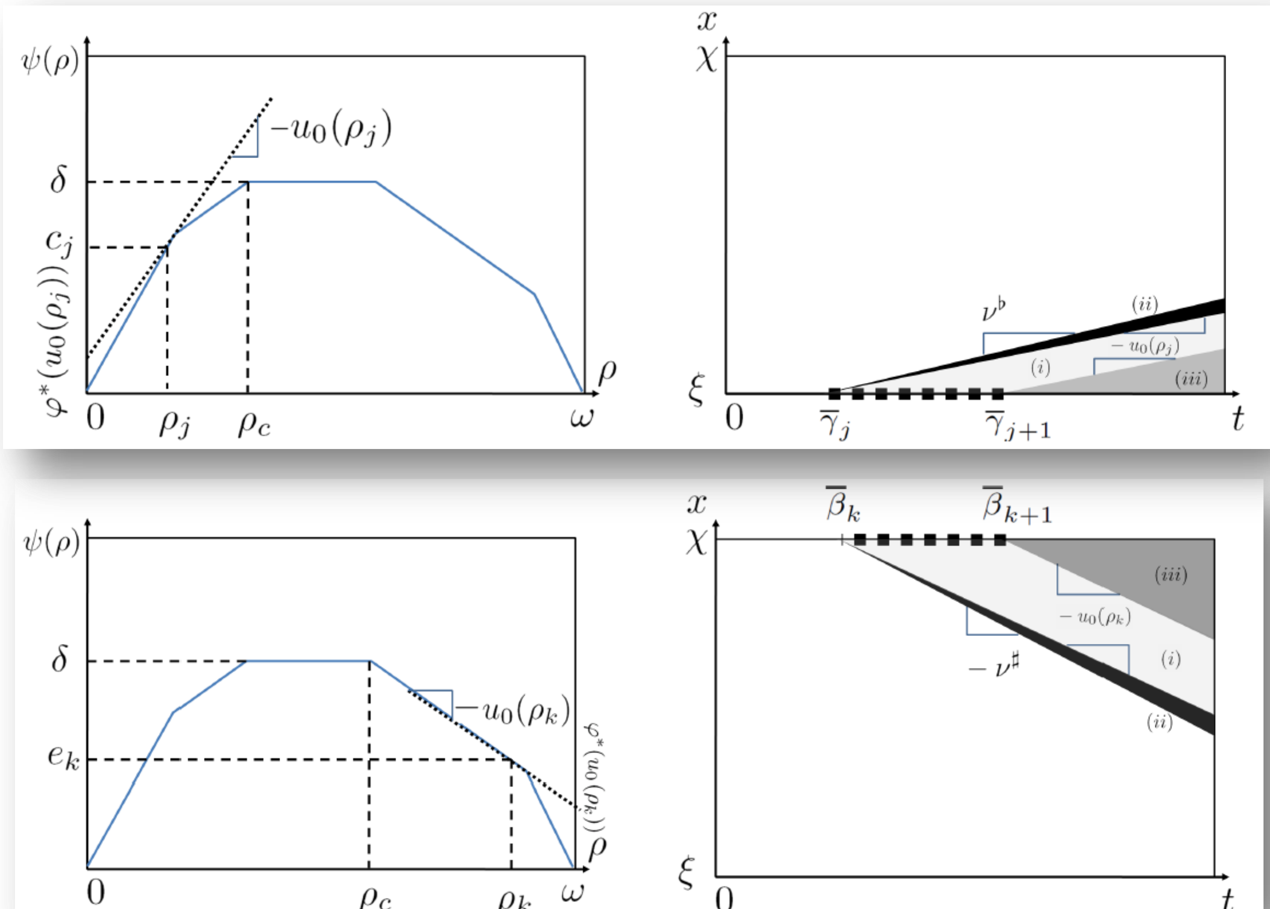


[Claudel Bayen IEEE TAC part II 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Solutions to affine value conditions

:

The cases of upstream and downstream boundary conditions are similar, and can also be solved using 1D convex optimization

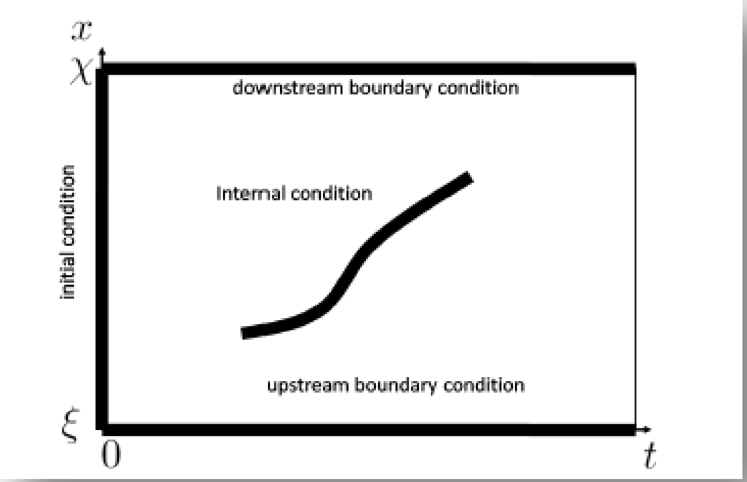


[Claudel Bayen TAC 2010]

Fixed or moving bottlenecks

Affine internal condition definition:

$$\mu_l(t,x) = \begin{cases} g_l(t-\overline{\delta}_l) + h_l & \text{if } x = x_l + v_l(t-\overline{\delta}_l) \\ & \text{and } t \in [\overline{\delta}_l, \overline{\delta}_{l+1}] \\ +\infty & \text{otherwise} \end{cases}$$



[Daganzo 2005 TR-B] [Daganzo 2006] [Claudel Bayen TAC 2010]

Explicit solutions

Solution can be explicitly computed (similarly as before)

Internal conditions apply in the weak sense (as upstream and downstream boundary conditions)

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

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

[Claudel Bayen TAC 2010]

Applications of internal conditions

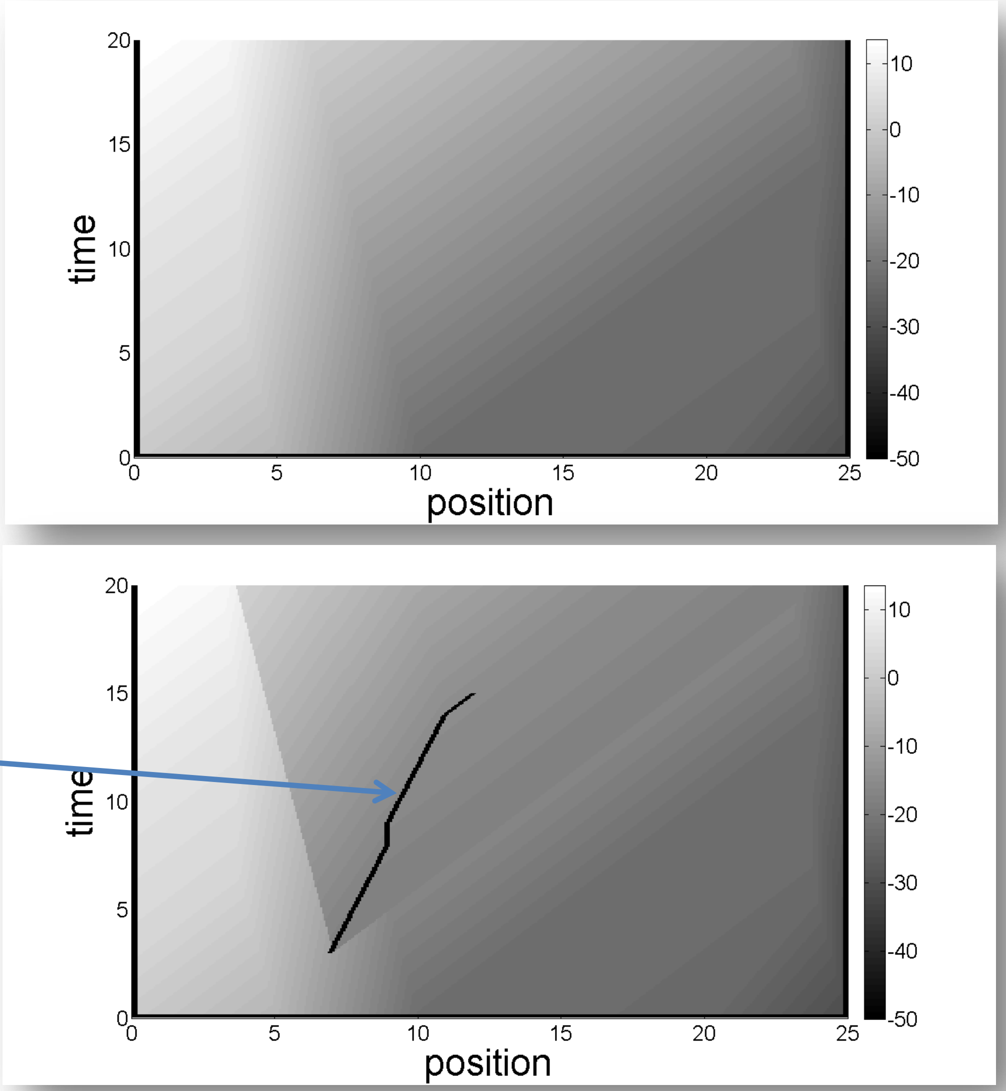
- Modeling the effects of a slow vehicle on traffic.

- Estimation problems with Lagrangian data.

- Inverse modeling problems with Lagrangian data

- Traffic light or accident (fixed bottleneck) modeling

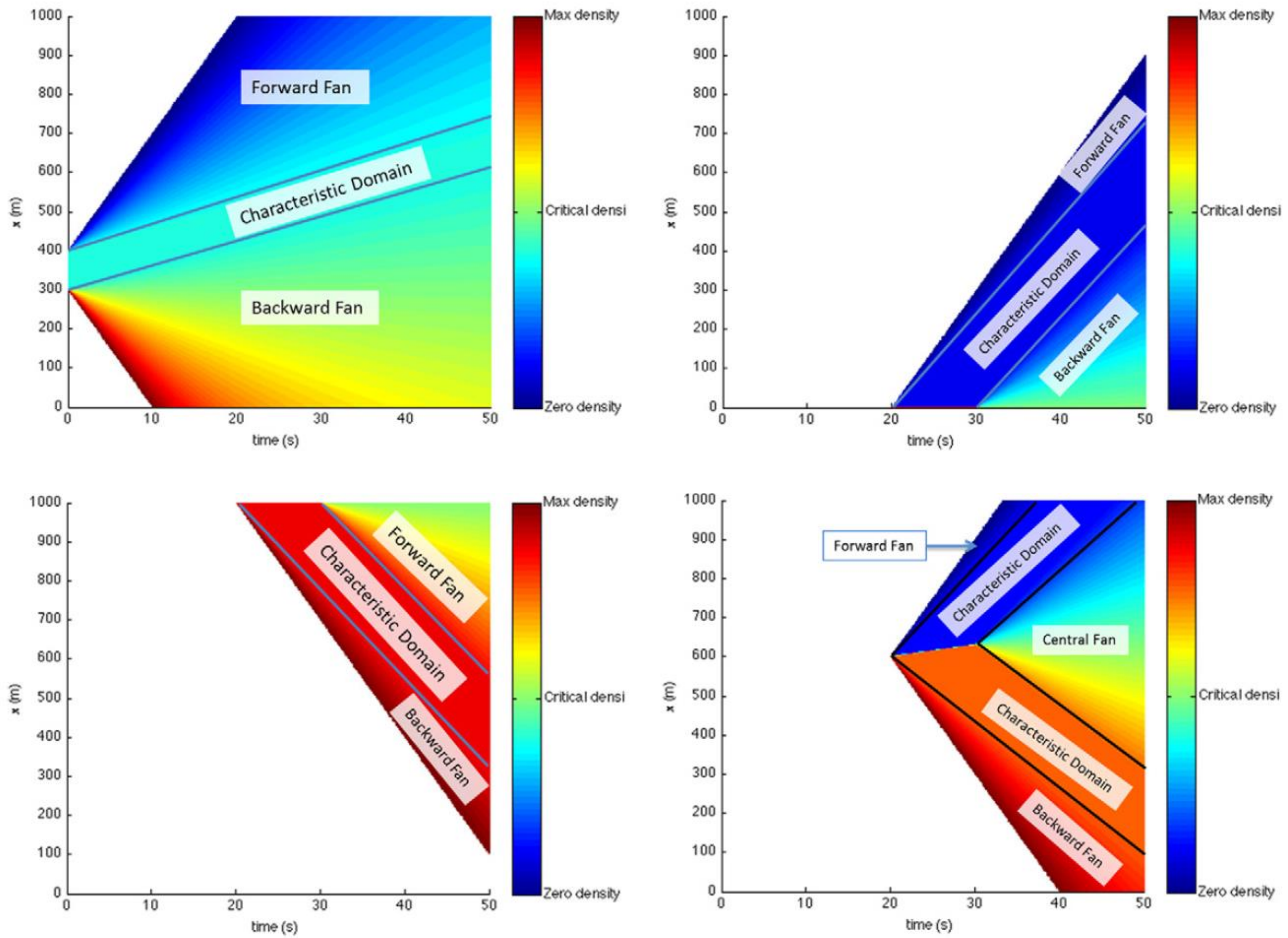
trajectory



[Claudel Bayen TAC 2010]

Solution structure

Solution structure:



[Claudel Bayen TAC 2010]

General solution method for affine value condition functions

Solution method:

LAX-HOPF ALGORITHM FOR THE MOSKOWITZ FUNCTION

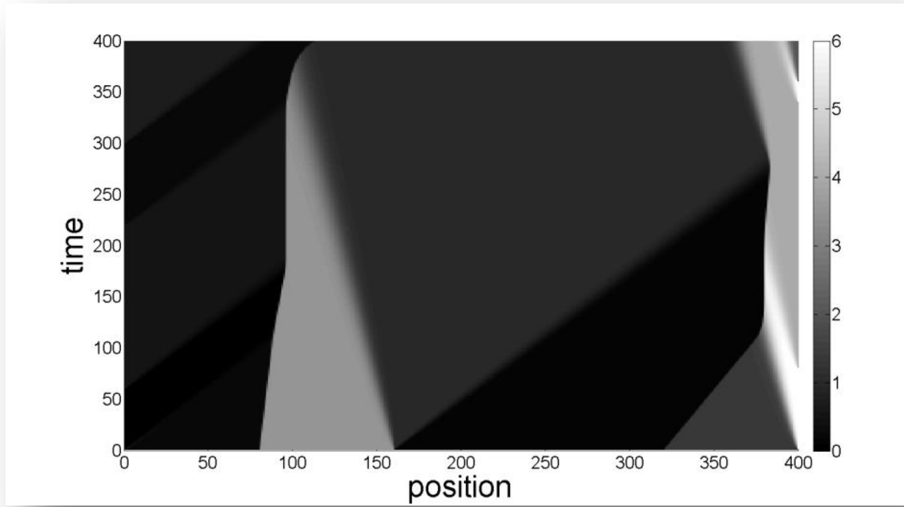
INITIALIZATION
Definition of $\mathrm{M}(M,N) \longleftarrow +\infty \quad \forall (M,N) \in \mathcal{G}$ [Output matrix containing the Moskowitz function]
MAIN LOOP
For $M := 0$ **to** n_t **do** [time iteration]
For $N := 0$ **to** n_x **do** [space iteration]
Definition of $x_N := N\Delta x + \xi$ and $t_M := M\Delta t$ [space and time grid definition]
For $i \in I$ **do** [iteration on the set of initial conditions]
Compute $\mathrm{M}_{\mathbf{M}_{\mathbf{0}_i}}(t_M, x_N)$
If $\mathrm{M}_{\mathbf{M}_{\mathbf{0}_i}}(t_M, x_N) \leq \mathrm{M}(M,N)$ **then** $\mathrm{M}(M,N) = \mathrm{M}_{\mathbf{M}_{\mathbf{0}_i}}(t_M, x_N)$
For $j \in J$ **do** [iteration on the set of left boundary conditions]
Compute $\mathrm{M}_{\gamma_j}(t_M, x_N)$
If $\mathrm{M}_{\gamma_j}(t_M, x_N) \leq \mathrm{M}(M,N)$ **then** $\mathrm{M}(M,N) = \mathrm{M}_{\gamma_j}(t_M, x_N)$
For $k \in K$ **do** [iteration on the set of right boundary conditions]
Compute $\mathrm{M}_{\beta_k}(t_M, x_N)$
If $\mathrm{M}_{\beta_k}(t_M, x_N) \leq \mathrm{M}(M,N)$ **then** $\mathrm{M}(M,N) = \mathrm{M}_{\beta_k}(t_M, x_N)$
For $l \in L$ **do** [iteration on the set of internal conditions]
Compute $\mathrm{M}_{\mu_l}(t_M, x_N)$
If $\mathrm{M}_{\mu_l}(t_M, x_N) \leq \mathrm{M}(M,N)$ **then** $\mathrm{M}(M,N) = \mathrm{M}_{\mu_l}(t_M, x_N)$
RETURN $\mathrm{M}(M,N)$

[Claudel Bayen TAC 2010]

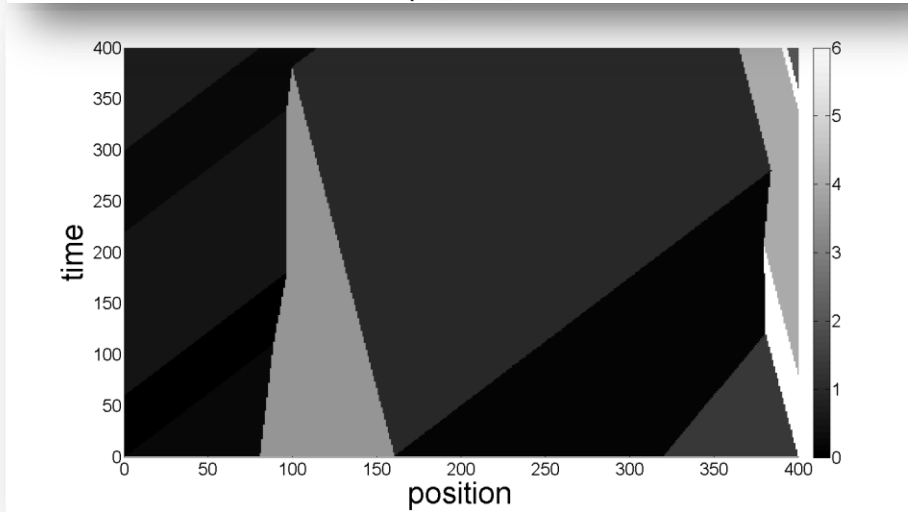
Extension for solutions to the LWR PDE

The solutions to the LWR PDE (density) can also be computed semi analytically:

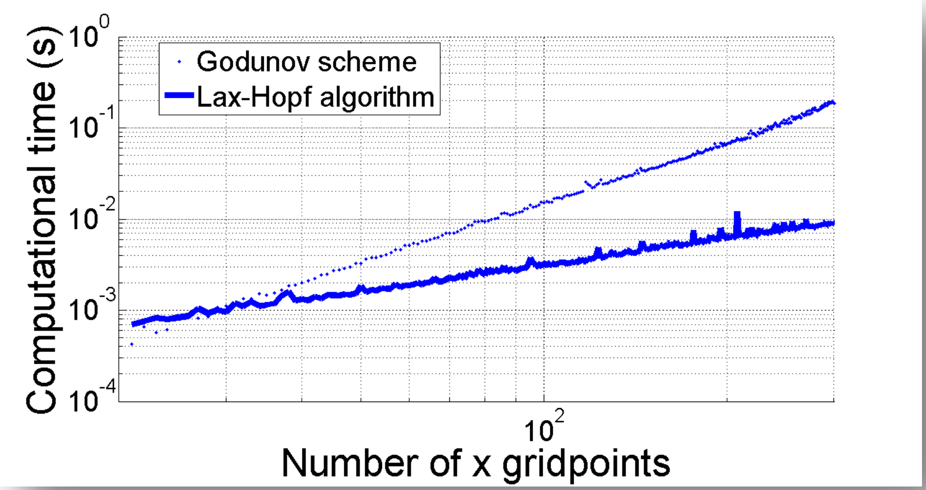
Godunov

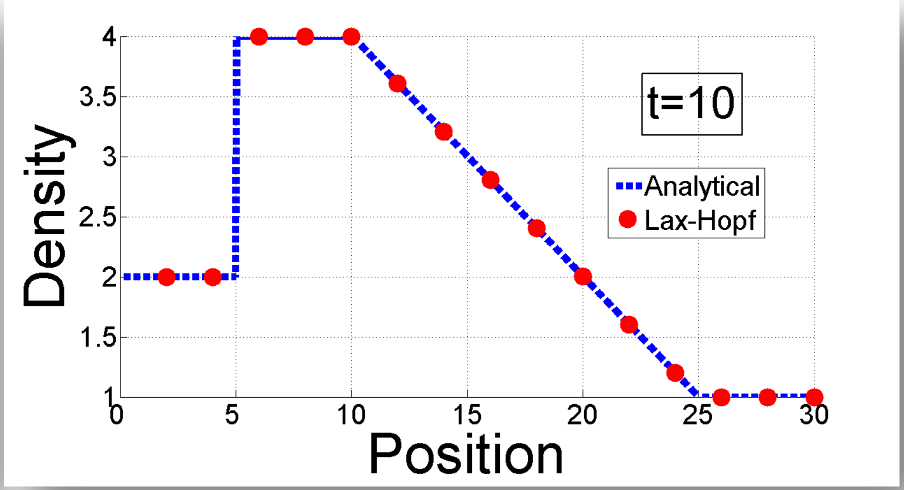


Lax-Hopf



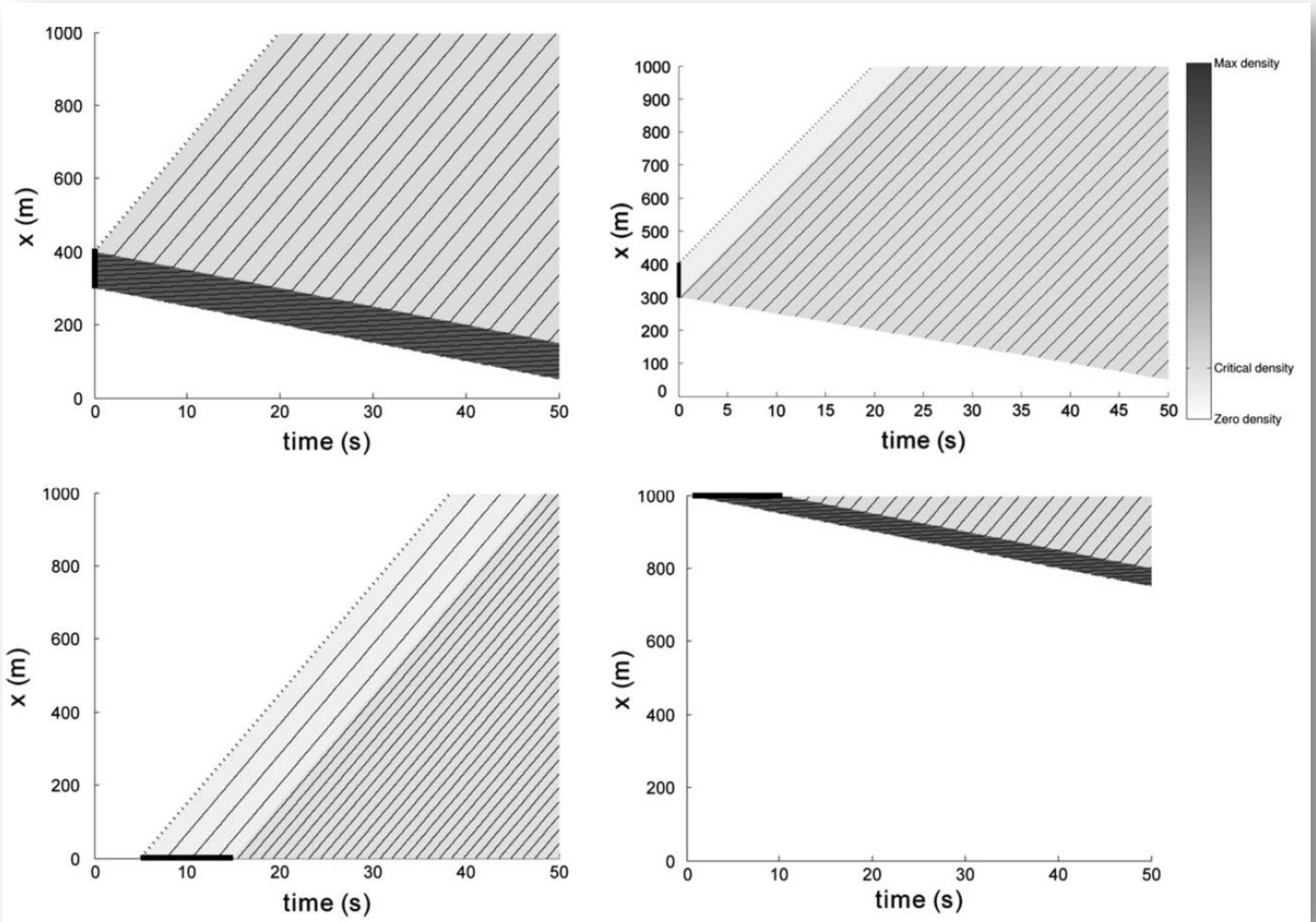
Semi-analytic computational methods





- Pros
  - Very fast (speed depends upon the problem, and number of points used)
  - Exact derivation of both the density and Moskowitz function (for the same cost), for any concave fundamental diagram, on any grid
  - Can integrate probe data
- Cons
  - Slower than the CTM if the density has to be computed everywhere

Faster algorithm for triangular fundamental diagram



Faster algorithm for triangular fundamental diagram

- If the fundamental diagram is triangular, the boundary condition components satisfy an additional inequality constraints

- Reduces the worst case number of operations from $n_i+n_u+n_d$ to n_i+2

- With this additional property, the worst case computational time at a fixed point (t,x) is independent of t

Outline

Primitive flow models based on LWR

- The LWR PDE
- Integral formulation and physical interpretation
- Isolines of the Moskowitz function

Solution methods based on Viability theory

- Viability formulation and properties
- Properties of the viability solution
- Semi-analytic formulation (exact solution)

Discussion

- Comparison with VT
- GSOM models

Conclusion

Semi-analytic computational methods

TEXAS
The University of Texas at Austin

- Many existing computational methods:
- For LWR:
  - Godunov scheme (or equivalently CTM)
  - Wave-front tracking
  - Other finite difference schemes (ENO, WENO)
- For HJ:
  - Lax Friedrichs schemes (or other numerical schemes)
  - Variational method (dynamic programming)
  - Semi-analytic method (for homogeneous problems), which can be used for both HJ and LWR

[Daganzo06] [Claudel Bayen IEEE TAC part I & 2, 2010] [Mazare Dehwah Claudel Bayen, TR-B 2012]

Comparison with Variational Theory

	Lax–Hopf algorithm	Dynamic programming (variational method)
Computational principle	Minimization of closed form partial solutions (grid-free)	Minimization of a cost function over a computational grid
Computational cost	Proportional to the total number of initial and boundary condition blocks (depends upon the complexity of the problem)	Function of the grid (depends upon the complexity of the grid)
Exactness	Exact solution for the Moskowitz and density functions for piecewise affine initial boundary and internal boundary conditions and arbitrary concave fundamental diagrams	Exact for piecewise affine initial and boundary conditions and piecewise affine fundamental diagrams. Not exact for arbitrary concave fundamental diagrams, and no exact computation of the density function
Integration of internal boundary conditions (moving bottlenecks, traffic lights...)	Straightforward (explicit formulas also exist for internal boundary conditions)	Possible, but requires the definition of shortcuts in the domain of the internal boundary condition
Integration of space/time varying fundamental diagrams	Possible, but requires a completely new derivation of explicit formulas for space/time varying diagrams	Straightforward (requires the introduction of a space or time dependent cost)

Pros/cons of common computational methods for LWR

TEXAS
The University of Texas at Austin

	Lax–Hopf algorithm	Variational method	Godunov scheme	Wave-front tracking
Computational principle	Minimization of closed form partial solutions	Minimization of a cost function over a computational grid	First order finite differences scheme	Event-based scheme
Main advantages	Exact for general concave fundamental diagrams. Very fast for computing the solution at one particular point (or on a small domain). Recomputing the solution after adding a fixed or moving bottleneck is very fast	Exact for some classes of fundamental diagrams. Ability to use space or time dependent fundamental diagrams. Ability to use fixed or moving bottlenecks	Easiness of implementation, natural extension to networks. Ability to use space or time dependent fundamental diagrams	Exact for some classes of fundamental diagrams. Ability to use non-concave or non-continuous fundamental diagrams. Performs well for computing the solution on a full space-time domain. Extends naturally to networks

Pros/cons of common computational methods for LWR

	Lax–Hopf algorithm	Variational method	Godunov scheme	Wave-front tracking
Main disadvantages	Inability to use space or time dependent fundamental diagrams. Results are not exact for networks (unless event-based algorithms are used)	Slower than the Lax–Hopf algorithm (requires in the best case the same number of operations as the Lax–Hopf algorithm). No exact derivation of the density function	Not exact. Limited by the CFL condition. Adding moving bottlenecks is difficult	Difficulty to predict the computational time in advance. Can be very slow if the fundamental diagram contains a large number of pieces

GSOM models

- GSOM (generic second order models) are a class of traffic flow models:

  - Complete structure:
    - $\frac{\partial k(t,x)}{\partial t} + \frac{\partial k(t,x)v(t,x)}{\partial x} = 0$ (conservation of vehicles)
    - $\frac{dI}{dt} = \frac{\partial I(t,x)}{\partial t} + v(t,x)\frac{\partial I(t,x)}{\partial x} = \mathrm{f(I)}$ (evolution of the attributes)
    - $v(t,x) = \psi(k(t,x), I(t,x))$ (variable fundamental diagram)

  - If the attributes do not evolve along trajectories, then f(.)=0

[Costeseque, Lebacque 2013-2014]

GSOM models

- Examples of GSOM models:
  - ARZ
  - LWR
  - LWR multi class/ multi lane/ multi commodity
  - LWR bounded acceleration
  - Stochastic LWR
  - 1 phase Colombo
- Can be written as a time dependent HJ PDE

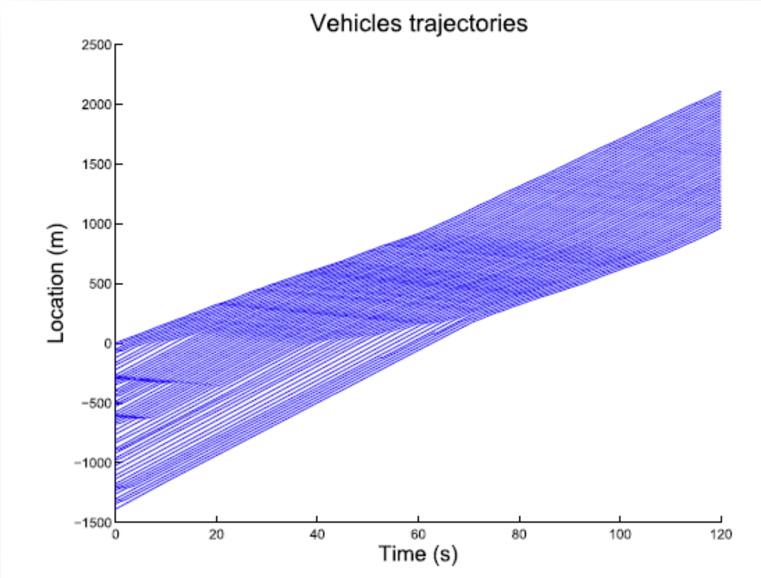
$$\frac{\partial X(t,N)}{\partial t} - \xi\left(-\frac{\partial X(t,N)}{\partial N}, N, t\right) = 0$$

[Costeseque, Lebacque 2013-2014]

GSOM models

- Scheme is exact if the attribute does not evolve over time (case of multi-flow or multi-class models for instance), otherwise method is approximate (due to the integration of I)

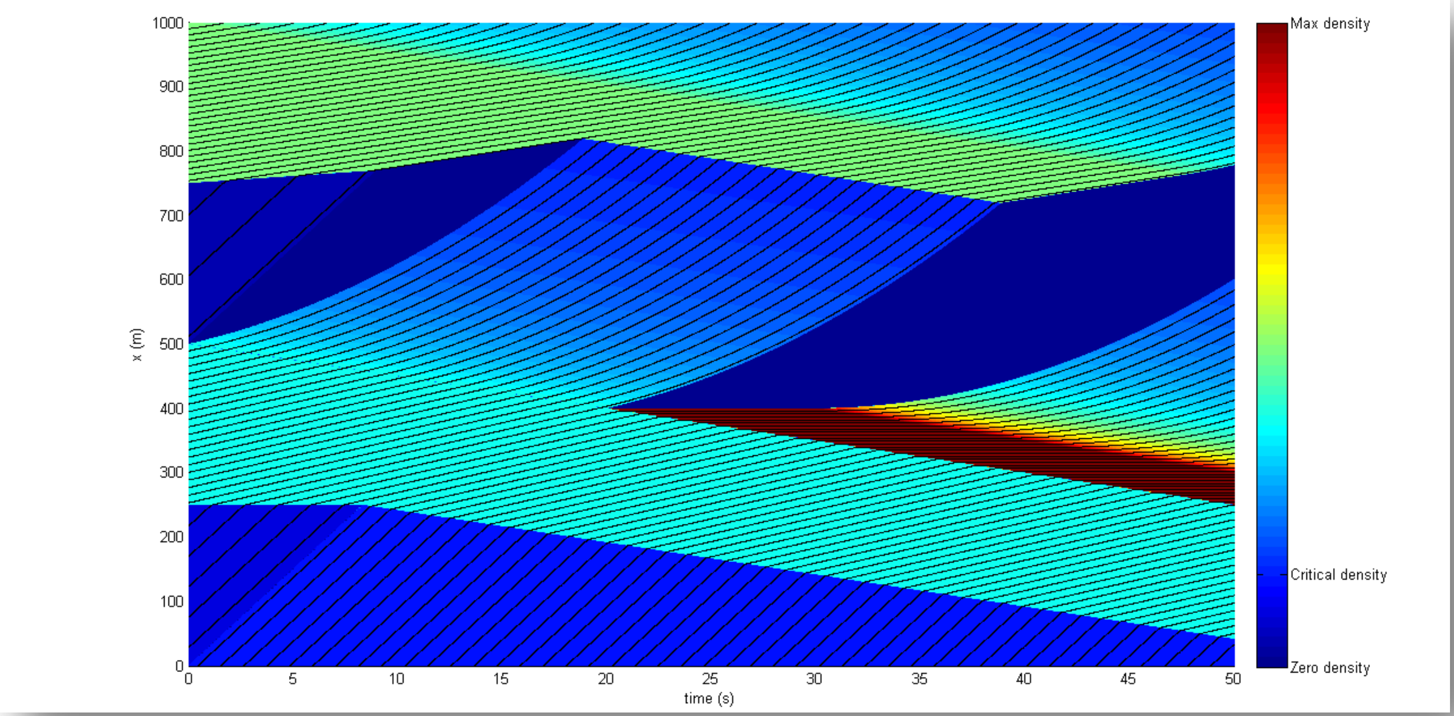
- Example: solution to 1 phase Colombo model



[Costeseque, Lebacque 2013-2014]

GSOM models

- Example: solution to LWR-bounded acceleration model



Outline

The University of Texas at Austin

Primitive flow models based on LWR

- The LWR PDE
- Integral formulation and physical interpretation
- Isolines of the Moskowitz function

Solution methods based on Viability theory

- Viability formulation and properties
- Properties of the viability solution
- Semi-analytic formulation (exact solution)

Discussion

- Comparison with VT
- GSOM models

Conclusion

Conclusion

Fast and exact algorithm for computing solutions to HJ PDEs or to the LWR PDE

Very useful in some situations: computing boundary conditions (network problem), computing the solution at a finite time (when not interested in the intermediate times)

Extensions to most types of second order models (GSOM). Exact solutions, modulo errors in the integration of attributes…