Mean Field Games: Numerical Methods and Applications in Machine Learning

Part 3: Numerical Schemes for MF PDE Systems

Mathieu LAURIÈRE

https://mlauriere.github.io/teaching/MFG-PKU-3.pdf

Peking University
Summer School on Applied Mathematics
July 26 – August 6, 2021

RECAP

Outline

1. Introduction

2. A Finite Difference Scheme

3. A Semi-Lagrangian Scheme

MFG PDE System

Goal: (1) introduce and **(2)** solve a discrete version of the MFG PDE system:

$$
\begin{cases}
0 = -\dfrac{\partial u}{\partial t}(t,x) - \nu \Delta u(t,x) + H(x, m(t,\cdot), \nabla u(t,x)), \\
0 = \dfrac{\partial m}{\partial t}(t,x) - \nu \Delta m(t,x) - \operatorname{div}\left(m(t,\cdot)\partial_p H(\cdot, m(t), \nabla u(t,\cdot))\right)(x), \\
\ u(T,x) = g(x, m(T,\cdot)), \qquad m(0,x) = m_0(x)
\end{cases}
$$

MFG PDE System

Goal: (1) introduce and **(2)** solve a discrete version of the MFG PDE system:

$$\begin{cases} 0 = -\dfrac{\partial u}{\partial t}(t,x) - \nu \Delta u(t,x) + H(x, m(t,\cdot), \nabla u(t,x)), \\ 0 = \dfrac{\partial m}{\partial t}(t,x) - \nu \Delta m(t,x) - \operatorname{div}\left(m(t,\cdot)\partial_p H(\cdot, m(t), \nabla u(t,\cdot))\right)(x), \\ \ u(T,x) = g(x, m(T,\cdot)), \qquad m(0,x) = m_0(x) \end{cases}$$

Desirable properties for (1):

- **Mass** and **positivity** of distribution: $\int_{\mathcal{S}} m(t,x)dx = 1$, $m \geq 0$
- **Convergence** of discrete solution to continuous solution as mesh step $\to 0$

MFG PDE System

Goal: (1) introduce and **(2)** solve a discrete version of the MFG PDE system:

$$\begin{cases} 0 = -\dfrac{\partial u}{\partial t}(t,x) - \nu \Delta u(t,x) + H(x, m(t,\cdot), \nabla u(t,x)), \\ 0 = \dfrac{\partial m}{\partial t}(t,x) - \nu \Delta m(t,x) - \operatorname{div}\left(m(t,\cdot) \partial_p H(\cdot, m(t), \nabla u(t,\cdot)) \right)(x), \\ \;u(T,x) = g(x, m(T,\cdot)), \qquad m(0,x) = m_0(x) \end{cases}$$

Desirable properties for (1):

- **Mass** and **positivity** of distribution: $\int_{\mathcal{S}} m(t,x)dx = 1$, $m \geq 0$
- **Convergence** of discrete solution to continuous solution as mesh step $\to 0$
- The KFP equation is the **adjoint** of the linearized HJB equation
- Link with **optimality condition** of a discrete problem

$\Rightarrow$ Needs a careful discretization

MFG PDE System

Goal: (1) introduce and **(2)** solve a discrete version of the MFG PDE system:

$$\begin{cases} 0 = -\dfrac{\partial u}{\partial t}(t,x) - \nu \Delta u(t,x) + H(x, m(t,\cdot), \nabla u(t,x)), \\ 0 = \dfrac{\partial m}{\partial t}(t,x) - \nu \Delta m(t,x) - \mathrm{div}\left(m(t,\cdot) \partial_p H(\cdot, m(t), \nabla u(t,\cdot)) \right)(x), \\ \quad u(T,x) = g(x, m(T,\cdot)), \qquad m(0,x) = m_0(x) \end{cases}$$

Desirable properties for (1):

- **Mass** and **positivity** of distribution: $\int_{\mathcal{S}} m(t,x)dx = 1$, $m \geq 0$
- **Convergence** of discrete solution to continuous solution as mesh step $\to 0$
- The KFP equation is the **adjoint** of the linearized HJB equation
- Link with **optimality condition** of a discrete problem

$\Rightarrow$ Needs a careful discretization

For (2): Once we have a discrete system, how can we compute its solution?

Outline

1. Introduction

2. A Finite Difference Scheme
- FD Scheme
- Algorithms

3. A Semi-Lagrangian Scheme

Outline

1. Introduction

2. A Finite Difference Scheme
- FD Scheme
- Algorithms

3. A Semi-Lagrangian Scheme

Discretization

Semi-implicit finite difference scheme introduced by [Achdou, Capuzzo-Dolcetta][1]
Discretization:

- For simplicity we consider the domain $\mathbb{T}$ = one-dimensional (unit) torus.
- Let $\nu = \sigma^2/2$.
- We consider N_h and N_T steps respectively in space and time.
- Let $h = 1/N_h$ and $\Delta t = T/N_T$. Let $\mathbb{T}_h$ = discretized torus.
- We approximate $m_0(x_i)$ by ρ_i^0 such that $h\sum_i \rho_i^0 = 1$.

[1] Achdou, Y., & Capuzzo-Dolcetta, I. (2010). Mean field games: numerical methods. *SIAM Journal on Numerical Analysis*, 48(3), 1136-1162.

Discretization

Semi-implicit finite difference scheme introduced by [Achdou, Capuzzo-Dolcetta][1]
Discretization:

- For simplicity we consider the domain $\mathbb{T} =$ one-dimensional (unit) torus.
- Let $\nu = \sigma^2/2$.
- We consider N_h and N_T steps respectively in space and time.
- Let $h = 1/N_h$ and $\Delta t = T/N_T$. Let $\mathbb{T}_h =$ discretized torus.
- We approximate $m_0(x_i)$ by ρ_i^0 such that $h \sum_i \rho_i^0 = 1$.

Then we introduce the following **discrete operators** : for $\varphi \in \mathbb{R}^{N_T+1}$ and $\psi \in \mathbb{R}^{N_h}$

- **time derivative** : $$(D_t\varphi)^n := \frac{\varphi^{n+1} - \varphi^n}{\Delta t}, \qquad 0 \leq n \leq N_T - 1$$
- **Laplacian** : $$(\Delta_h \psi)_i := -\frac{1}{h^2}\left(2\psi_i - \psi_{i+1} - \psi_{i-1}\right), \qquad 0 \leq i \leq N_h$$
- **partial derivative** : $$(D_h\psi)_i := \frac{\psi_{i+1} - \psi_i}{h}, \qquad 0 \leq i \leq N_h$$
- **gradient** : $$[\nabla_h \psi]_i := \left((D_h\psi)_i, (D_h\psi)_{i-1}\right), \qquad 0 \leq i \leq N_h$$

[1] Achdou, Y., & Capuzzo-Dolcetta, I. (2010). Mean field games: numerical methods. *SIAM Journal on Numerical Analysis*, 48(3), 1136-1162.

Discrete Hamiltonian

For simplicity, we assume that the drift b and the costs f and g are of the form

$$b(x, m, v) = v, \qquad f(x, m, v) = L(x, v) + \mathtt{f}_0(x, m), \qquad g(x, m) = \mathtt{g}_0(x, m).$$

where $x \in \mathbb{R}^d, v \in \mathbb{R}^d, m \in \mathbb{R}_+$. Then

$$H(x, m, p) = \max_v \{-L(x, v) - \langle v, p\rangle\} - \mathtt{f}_0(x, m) = H_0(x, p) - \mathtt{f}_0(x, m)$$

where H_0 is the convex conjugate (also denoted L^*) of L with respect to v:

$$H_0(x, p) = L^*(x, p) = \sup_v \{\langle v, p\rangle - L(x, v)\}$$

Discrete Hamiltonian

For simplicity, we assume that the drift b and the costs f and g are of the form

$$b(x, m, v) = v, \qquad f(x, m, v) = L(x, v) + \mathtt{f}_0(x, m), \qquad g(x, m) = \mathtt{g}_0(x, m).$$

where $x \in \mathbb{R}^d$, $v \in \mathbb{R}^d$, $m \in \mathbb{R}_+$. Then

$$H(x, m, p) = \max_v \{-L(x, v) - \langle v, p \rangle\} - \mathtt{f}_0(x, m) = H_0(x, p) - \mathtt{f}_0(x, m)$$

where H_0 is the convex conjugate (also denoted L^*) of L with respect to v:

$$H_0(x, p) = L^*(x, p) = \sup_v \{\langle v, p \rangle - L(x, v)\}$$

Discrete Hamiltonian: $(x, p_1, p_2) \mapsto \tilde{H}_0(x, p_1, p_2)$ satisfying:

- Monotonicity: decreasing w.r.t. p_1 and increasing w.r.t. p_2
- Consistency with H_0: for every x, p, $\tilde{H}_0(x, p, p) = H_0(x, p)$
- Differentiability: for every x, $(p_1, p_2) \mapsto \tilde{H}_0(x, p_1, p_2)$ is $\mathcal{C}^1$
- Convexity: for every x, $(p_1, p_2) \mapsto \tilde{H}_0(x, p_1, p_2)$ is convex

Example: if $H_0(x, p) = |p|^2$, a possible choice is $\tilde{H}_0(x, p_1, p_2) = ({p_1}^-)^2 + ({p_2}^+)^2$

Discrete HJB

Discrete solution: We replace $u, m : [0,T] \times \mathbb{T} \to \mathbb{R}$ by vectors

$$U, M \in \mathbb{R}^{(N_T+1) \times N_h}$$

Discrete HJB

Discrete solution: We replace $u, m : [0,T] \times \mathbb{T} \to \mathbb{R}$ by vectors

$$U, M \in \mathbb{R}^{(N_T+1)\times N_h}$$

The **HJB equation**

$$\begin{cases} \partial_t u(t,x) + \nu \Delta u(t,x) + H_0(x, \nabla u(t,x)) = \mathtt{f}_0(x, m(t,x)) \\ u(T,x) = \mathtt{g}_0(x, m(T,x)) \end{cases}$$

is discretized as:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, M_i^{n+1}) \\ U_i^{N_T} = \mathtt{g}_0(x_i, M_i^{N_T}) \end{cases}$$

Discrete KFP

The **KFP equation**

$$\partial_t m(t,x) - \nu\Delta m(t,x) + \operatorname{div}\Big(m(t,x)\partial_q H(x, m(t), \nabla u(t,x))\Big) = 0, \qquad m(0,x) = m_0(x)$$

is discretized as

$$(D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^n, M^{n+1}) = 0, \qquad M_i^0 = \rho_i^0$$

Discrete KFP

The **KFP equation**

$$\partial_t m(t,x) - \nu \Delta m(t,x) + \operatorname{div}\Big(m(t,x) \partial_q H(x, m(t), \nabla u(t,x)) \Big) = 0, \qquad m(0,x) = m_0(x)$$

is discretized as

$$(D_t M_i)^n - \nu (\Delta_h M^{n+1})_i - \mathcal{T}_i(U^n, M^{n+1}) = 0, \qquad M_i^0 = \rho_i^0$$

Here we use the **discrete transport operator** $\approx -\operatorname{div}(\dots)$

$$\mathcal{T}_i(U,M) := \frac{1}{h} \begin{pmatrix} M_i \partial_{p_1} \tilde{H}_0(x_i, [\nabla_h U]_i) - M_{i-1} \partial_{p_1} \tilde{H}_0(x_{i-1}, [\nabla_h U]_{i-1}) \\ + M_{i+1} \partial_{p_2} \tilde{H}_0(x_{i+1}, [\nabla_h U]_{i+1}) - M_i \partial_{p_2} \tilde{H}_0(x_i, [\nabla_h U]_i) \end{pmatrix}$$

Discrete KFP

The **KFP equation**

$$\partial_t m(t,x) - \nu \Delta m(t,x) + \operatorname{div}\Big(m(t,x) \partial_q H(x, m(t), \nabla u(t,x)) \Big) = 0, \qquad m(0,x) = m_0(x)$$

is discretized as

$$(D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^n, M^{n+1}) = 0, \qquad M_i^0 = \rho_i^0$$

Here we use the **discrete transport operator** $\approx -\operatorname{div}(\dots)$

$$\mathcal{T}_i(U,M) := \frac{1}{h} \left(\begin{array}{l} M_i \partial_{p_1} \tilde{H}_0(x_i, [\nabla_h U]_i) - M_{i-1} \partial_{p_1} \tilde{H}_0(x_{i-1}, [\nabla_h U]_{i-1}) \\ + M_{i+1} \partial_{p_2} \tilde{H}_0(x_{i+1}, [\nabla_h U]_{i+1}) - M_i \partial_{p_2} \tilde{H}_0(x_i, [\nabla_h U]_i) \end{array} \right)$$

Intuition: weak formulation & integration by parts

$$\int_{\mathbb{T}} \operatorname{div}\left(m \partial_p H_0(x, \nabla u)\right) w = - \int_{\mathbb{T}} m \partial_p H_0(x, \nabla u) \cdot \nabla w$$

is discretized as

$$-h \sum_i \mathcal{T}_i(U,M) W_i = h \sum_i M_i \nabla_q \tilde{H}_0(x_i, [\nabla_h U]_i) \cdot [\nabla_h W]_i$$

Discrete System – Properties

Discrete forward-backward system:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, M_i^{n+1}), & \forall n \leq N_T - 1 \\ (D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^n, M^{n+1}) = 0, & \forall n \leq N_T - 1 \\ M_i^0 = \rho_i^0, \qquad U_i^{N_T} = \mathtt{g}_0(x_i, M_i^{N_T}), \qquad i = 0, \ldots, N_h & \end{cases}$$

[2] Achdou, Y., & Capuzzo-Dolcetta, I. (2010). Mean field games: numerical methods. *SIAM Journal on Numerical Analysis*, 48(3), 1136-1162.

[3] Achdou, Y., Camilli, F., & Capuzzo-Dolcetta, I. (2012). Mean field games: numerical methods for the planning problem. *SIAM Journal on Control and Optimization*, 50(1), 77-109.

[4] Achdou, Y., & Porretta, A. (2016). Convergence of a finite difference scheme to weak solutions of the system of partial differential equations arising in mean field games. *SIAM Journal on Numerical Analysis*, 54(1), 161-186.

Discrete System – Properties

Discrete forward-backward system:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, M_i^{n+1}), & \forall n \le N_T - 1 \\ (D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^n, M^{n+1}) = 0, & \forall n \le N_T - 1 \\ M_i^0 = \rho_i^0, \qquad U_i^{N_T} = \mathtt{g}_0(x_i, M_i^{N_T}), \qquad i = 0, \dots, N_h \end{cases}$$

This scheme enjoys many nice properties, among which:

- It yields a monotone scheme for the KFP equation: **mass** and **positivity** are preserved
- **Convergence** to classical solution if monotonicity [Achdou, & Camilli, Capuzzo-Dolcetta][2,3]
- Can sometimes be used to show existence of a **weak** solution [Achdou, Porretta][4]
- The discrete KFP operator is the **adjoint** of the linearized Bellman operator
- **Existence** and **uniqueness** result for the discrete system
- It corresponds to the **optimality condition** of a discrete optimization problem (details later)

[2] Achdou, Y., & Capuzzo-Dolcetta, I. (2010). Mean field games: numerical methods. *SIAM Journal on Numerical Analysis*, 48(3), 1136-1162.

[3] Achdou, Y., Camilli, F., & Capuzzo-Dolcetta, I. (2012). Mean field games: numerical methods for the planning problem. *SIAM Journal on Control and Optimization*, 50(1), 77-109.

[4] Achdou, Y., & Porretta, A. (2016). Convergence of a finite difference scheme to weak solutions of the system of partial differential equations arising in mean field games. *SIAM Journal on Numerical Analysis*, 54(1), 161-186.

Outline

1. Introduction

2. A Finite Difference Scheme
- FD Scheme
- Algorithms

3. A Semi-Lagrangian Scheme

Algo 1: Fixed Point Iterations

Input: Initial guess $(\tilde{M}, \tilde{U})$; damping $\delta(\cdot)$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{M}, \hat{U})$ solving the finite difference system

Initialize $M^{(0)} = \tilde{M}^{(0)} = \tilde{M}, U^{(0)} = \tilde{U}$

for $\mathtt{k} = 0, 1, 2, \ldots, \mathtt{K} - 1$ **do**

Let $U^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, \tilde{M}_i^{(\mathtt{k}),n+1}), \quad n \leq N_T - 1 \\ U_i^{N_T} = \mathtt{g}_0(x_i, \tilde{M}_i^{(\mathtt{k}),N_T}) \end{cases}$$

Let $M^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} (D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^{(\mathtt{k}+1),n}, M^{n+1}) = 0, \quad n \leq N_T - 1 \\ M_i^0 = \rho_i^0 \end{cases}$$

Let $\tilde{M}^{(\mathtt{k}+1)} = \delta(\mathtt{k})\tilde{M}^{(\mathtt{k})} + (1 - \delta(\mathtt{k}))M^{(\mathtt{k}+1)}$

return $(M^{(\mathtt{K})}, U^{(\mathtt{K})})$

Algo 1: Fixed Point Iterations

Input: Initial guess $(\tilde{M}, \tilde{U})$; damping $\delta(\cdot)$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{M}, \hat{U})$ solving the finite difference system

Initialize $M^{(0)} = \tilde{M}^{(0)} = \tilde{M}, U^{(0)} = \tilde{U}$

for $\mathtt{k} = 0, 1, 2, \dots, \mathtt{K} - 1$ **do**

  Let $U^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, \tilde{M}_i^{(\mathtt{k}), n+1}), & n \leq N_T - 1 \\ U_i^{N_T} = \mathtt{g}_0(x_i, \tilde{M}_i^{(\mathtt{k}), N_T}) \end{cases}$$

  Let $M^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} (D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^{(\mathtt{k}+1), n}, M^{n+1}) = 0, & n \leq N_T - 1 \\ M_i^0 = \rho_i^0 \end{cases}$$

  Let $\tilde{M}^{(\mathtt{k}+1)} = \delta(\mathtt{k}) \tilde{M}^{(\mathtt{k})} + (1 - \delta(\mathtt{k})) M^{(\mathtt{k}+1)}$

return $(M^{(\mathtt{K})}, U^{(\mathtt{K})})$

Remark: the HJB equation is **non-linear**

- **Idea 1:** replace $\tilde{H}_0(x_i, [D_h U^n]_i)$ by $\tilde{H}_0(x_i, [D_h U^{(\mathtt{k}), n}]_i)$

Algo 1: Fixed Point Iterations

Input: Initial guess $(\tilde{M}, \tilde{U})$; damping $\delta(\cdot)$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{M}, \hat{U})$ solving the finite difference system

Initialize $M^{(0)} = \tilde{M}^{(0)} = \tilde{M}, U^{(0)} = \tilde{U}$

for $\mathtt{k} = 0, 1, 2, \ldots, \mathtt{K} - 1$ **do**

Let $U^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} -(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) = \mathtt{f}_0(x_i, \tilde{M}_i^{(\mathtt{k}), n+1}), \quad n \le N_T - 1 \\ U_i^{N_T} = \mathtt{g}_0(x_i, \tilde{M}_i^{(\mathtt{k}), N_T}) \end{cases}$$

Let $M^{(\mathtt{k}+1)}$ be the solution to:

$$\begin{cases} (D_t M_i)^n - \nu(\Delta_h M^{n+1})_i - \mathcal{T}_i(U^{(\mathtt{k}+1), n}, M^{n+1}) = 0, \quad n \le N_T - 1 \\ M_i^0 = \rho_i^0 \end{cases}$$

Let $\tilde{M}^{(\mathtt{k}+1)} = \delta(\mathtt{k})\tilde{M}^{(\mathtt{k})} + (1 - \delta(\mathtt{k}))M^{(\mathtt{k}+1)}$

return $(M^{(\mathtt{K})}, U^{(\mathtt{K})})$

Remark: the HJB equation is **non-linear**

- **Idea 1:** replace $\tilde{H}_0(x_i, [D_h U^n]_i)$ by $\tilde{H}_0(x_i, [D_h U^{(\mathtt{k}), n}]_i)$
- **Idea 2:** use non linear solver to find a zero of $\mathbb{R}^{N_h \times (N_T+1)} \ni U \mapsto \varphi(U) \in \mathbb{R}^{N_h \times N_T}$,

$$\varphi(U) = \Big(-(D_t U_i)^n - \nu(\Delta_h U^n)_i + \tilde{H}_0(x_i, [D_h U^n]_i) - \mathtt{f}_0(x_i, \tilde{M}_i^{(\mathtt{k}), n+1}) \Big)_{i=0,\ldots,N_h-1}^{n=0,\ldots,N_T-1}$$

Algo 2: Newton's Method for FD System

Idea: Directly look for a zero of $\varphi = (\varphi_{\mathcal{U}}, \varphi_{\mathcal{M}})^\top$ with $\varphi_{\mathcal{U}}$ and $\varphi_{\mathcal{M}}$ s.t.

$$\begin{cases} \varphi_{\mathcal{U}}(U, M) = 0 & \Leftrightarrow \; (U, M) \text{ solves discrete HJB equation} \\ \varphi_{\mathcal{M}}(U, M) = 0 & \Leftrightarrow \; (U, M) \text{ solves discrete KFP equation} \end{cases}$$

- Let $X^{(k)} = (U^{(k)}, M^{(k)})^\top$
- Iterate: $X^{(k+1)} = X^{(k)} - J_\varphi(X^{(k)})^{-1} \varphi(X^{(k)})$

Algo 2: Newton's Method for FD System

Idea: Directly look for a zero of $\varphi = (\varphi_{\mathcal{U}}, \varphi_{\mathcal{M}})^\top$ with $\varphi_{\mathcal{U}}$ and $\varphi_{\mathcal{M}}$ s.t.

$$\begin{cases} \varphi_{\mathcal{U}}(U,M) = 0 & \Leftrightarrow \ (U,M) \text{ solves discrete HJB equation} \\ \varphi_{\mathcal{M}}(U,M) = 0 & \Leftrightarrow \ (U,M) \text{ solves discrete KFP equation} \end{cases}$$

- Let $X^{(k)} = (U^{(k)}, M^{(k)})^\top$
- Iterate: $X^{(k+1)} = X^{(k)} - J_\varphi(X^{(k)})^{-1}\varphi(X^{(k)})$
- Or rather: $J_\varphi(X^{(k)})Y = -\varphi(X^{(k)})$, then $X^{(k+1)} = Y + X^{(k)}$

Algo 2: Newton's Method for FD System

Idea: Directly look for a zero of $\varphi = (\varphi_{\mathcal{U}}, \varphi_{\mathcal{M}})^\top$ with $\varphi_{\mathcal{U}}$ and $\varphi_{\mathcal{M}}$ s.t.

$$\begin{cases} \varphi_{\mathcal{U}}(U, M) = 0 & \Leftrightarrow \; (U, M) \text{ solves discrete HJB equation} \\ \varphi_{\mathcal{M}}(U, M) = 0 & \Leftrightarrow \; (U, M) \text{ solves discrete KFP equation} \end{cases}$$

- Let $X^{(k)} = (U^{(k)}, M^{(k)})^\top$
- Iterate: $X^{(k+1)} = X^{(k)} - J_\varphi(X^{(k)})^{-1}\varphi(X^{(k)})$
- Or rather: $J_\varphi(X^{(k)})Y = -\varphi(X^{(k)})$, then $X^{(k+1)} = Y + X^{(k)}$

Key step: Solve a linear system of the form

$$\begin{pmatrix} A_{\mathcal{U},\mathcal{U}} & A_{\mathcal{U},\mathcal{M}} \\ A_{\mathcal{M},\mathcal{U}} & A_{\mathcal{M},\mathcal{M}} \end{pmatrix} \begin{pmatrix} U \\ M \end{pmatrix} = \begin{pmatrix} G_{\mathcal{U}} \\ G_{\mathcal{M}} \end{pmatrix}$$

where $A_{\mathcal{U},\mathcal{M}}(U, M) = \nabla_U \varphi_{\mathcal{M}}(U, M), \quad A_{\mathcal{U},\mathcal{U}}(U, M) = \nabla_U \varphi_{\mathcal{U}}(U, M), \quad \dots$

Newton Method – Implementation

Linear system to be solved: $\begin{pmatrix} A_{\mathcal{U},\mathcal{U}} & A_{\mathcal{U},\mathcal{M}} \\ A_{\mathcal{M},\mathcal{U}} & A_{\mathcal{M},\mathcal{M}} \end{pmatrix} \begin{pmatrix} U \\ M \end{pmatrix} = \begin{pmatrix} G_{\mathcal{U}} \\ G_{\mathcal{M}} \end{pmatrix}$

Structure: $A_{\mathcal{U},\mathcal{M}}, A_{\mathcal{M},\mathcal{U}}$ are block-diagonal, $A_{\mathcal{U},\mathcal{U}} = A_{\mathcal{M},\mathcal{M}}^{\top}$, and

$$A_{\mathcal{U},\mathcal{U}} = \begin{pmatrix} D_1 & 0 & \cdots & \cdots & 0 \\ -\frac{1}{\Delta t}\mathrm{Id}_{N_h} & D_2 & \ddots & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & -\frac{1}{\Delta t}\mathrm{Id}_{N_h} & D_{N_T} \end{pmatrix}$$

where D_n corresponds to the discrete operator

$$Z = (Z_{i,j})_{i,j} \mapsto \left(\frac{1}{\Delta t} Z_{i,j} - \nu(\Delta_h Z)_{i,j} + [\nabla_h Z]_{i,j} \cdot \nabla_p \tilde{H}_0(x_{i,j}, [\nabla_h U^{(k),n}]_{i,j}) \right)_{i,j}$$

[5] Achdou, Y. (2013). Finite difference methods for mean field games. In Hamilton-Jacobi equations: approximations, numerical analysis and applications (pp. 1-47). Springer, Berlin, Heidelberg.

Newton Method – Implementation

Linear system to be solved: $\begin{pmatrix} A_{\mathcal{U},\mathcal{U}} & A_{\mathcal{U},\mathcal{M}} \\ A_{\mathcal{M},\mathcal{U}} & A_{\mathcal{M},\mathcal{M}} \end{pmatrix} \begin{pmatrix} U \\ M \end{pmatrix} = \begin{pmatrix} G_{\mathcal{U}} \\ G_{\mathcal{M}} \end{pmatrix}$

Structure: $A_{\mathcal{U},\mathcal{M}}, A_{\mathcal{M},\mathcal{U}}$ are block-diagonal, $A_{\mathcal{U},\mathcal{U}} = A_{\mathcal{M},\mathcal{M}}^\top$, and

$$A_{\mathcal{U},\mathcal{U}} = \begin{pmatrix} D_1 & 0 & \cdots & \cdots & 0 \\ -\frac{1}{\Delta t}\mathrm{Id}_{N_h} & D_2 & \ddots & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & 0 & -\frac{1}{\Delta t}\mathrm{Id}_{N_h} & D_{N_T} \end{pmatrix}$$

where D_n corresponds to the discrete operator

$$Z = (Z_{i,j})_{i,j} \mapsto \left(\frac{1}{\Delta t} Z_{i,j} - \nu(\Delta_h Z)_{i,j} + [\nabla_h Z]_{i,j} \cdot \nabla_p \tilde{H}_0(x_{i,j}, [\nabla_h U^{(k),n}]_{i,j}) \right)_{i,j}$$

Rem. Initial guess $(U^{(0)}, M^{(0)})$ is important for Newton's method

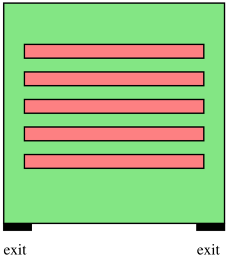
- Idea 1: initialize with the ergodic solution
- Idea 2: continuation method w.r.t. ν (converges more easily with a large viscosity)

See [Achdou'13][5] for more details.

[5] Achdou, Y. (2013). Finite difference methods for mean field games. In Hamilton-Jacobi equations: approximations, numerical analysis and applications (pp. 1-47). Springer, Berlin, Heidelberg.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

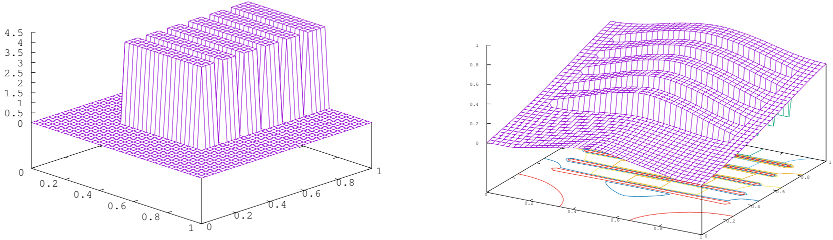


Geometry of the room

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

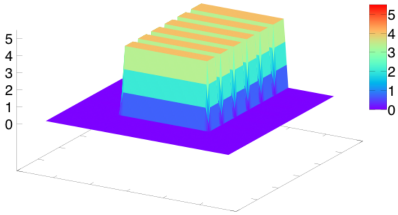


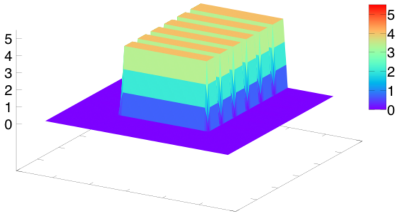
Initial density (left) and final cost (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]



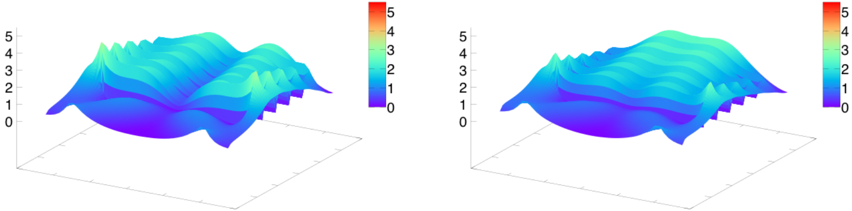


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

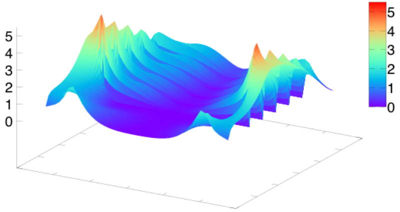


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

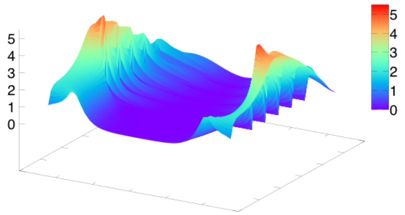


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

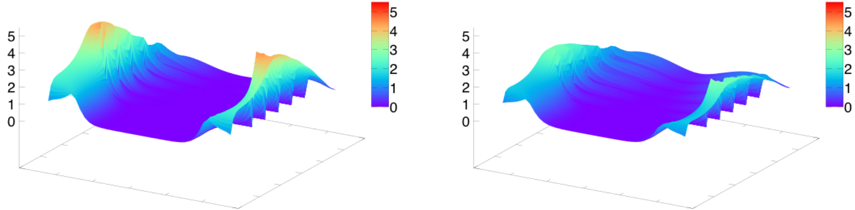


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

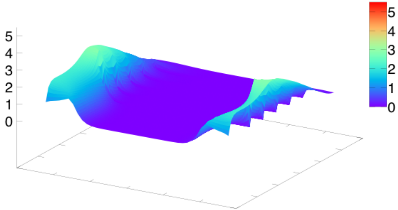


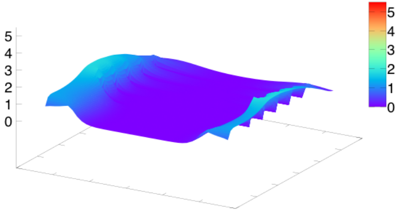
Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]



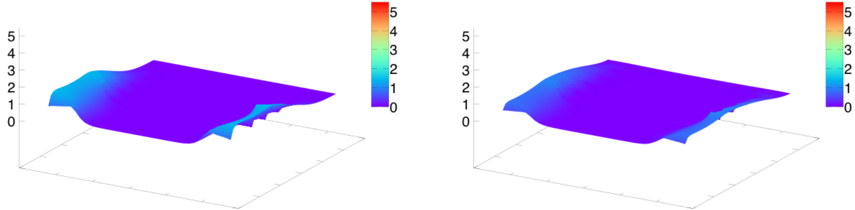


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

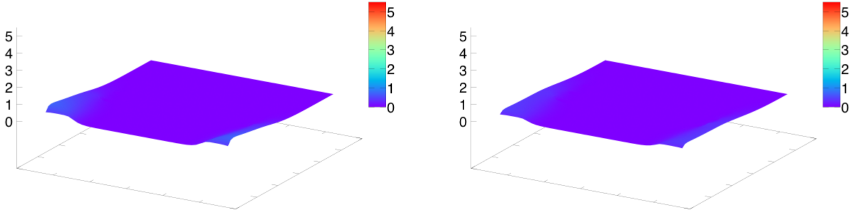


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

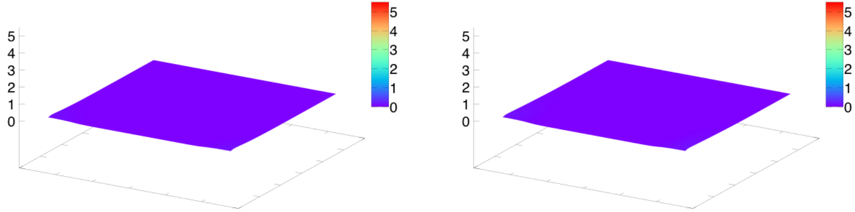


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

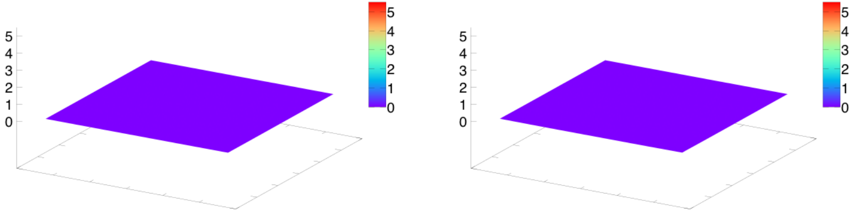


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

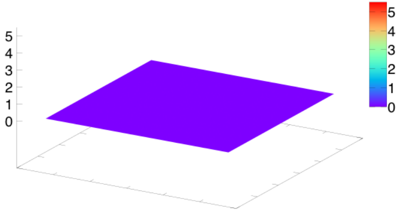


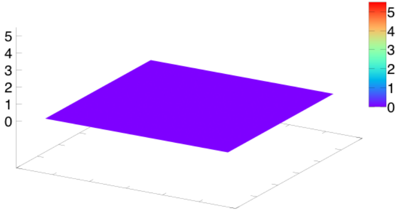
Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]



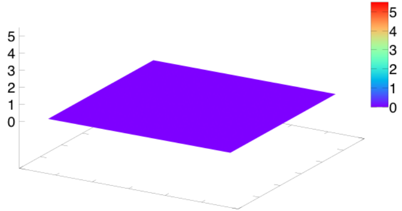


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

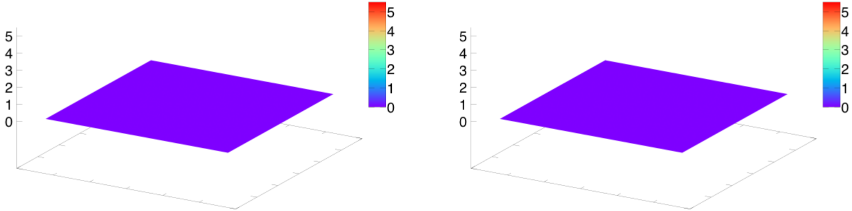


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]

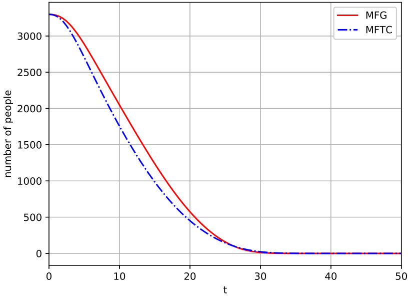


Density in MFGame (left) and MFControl (right)

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Example: Exit of a Room – Distribution

Example: evacuation of a room with obstacles and congestion [Achdou, L.'15][6]



Remaining mass inside the room

[6] Achdou, Y., & Laurière, M. (2015). On the system of partial differential equations arising in mean field type control. *Discrete & Continuous Dynamical Systems*, 35(9), 3879.

Outline

1. Introduction

2. A Finite Difference Scheme

3. A Semi-Lagrangian Scheme

MFG Setup

- Scheme introduced by [Carlini, Silva][7]
- For simplicity: $d = 1$, domain $\mathcal{S} = \mathbb{R}$, $\mathcal{A} = \mathbb{R}$
- $\nu = 0$ (degenerate second order case also possible; see [Carlini, Silva][8])
- Model:

$$b(x, m, v) = v$$

$$f(x, m, v) = \frac{1}{2}|v|^2 + f_0(x, m), \qquad g(x, m)$$

where f_0 and g depend on $m \in \mathcal{P}_1(\mathbb{R})$ in a potentially non-local way

[7] Carlini, E., & Silva, F. J. (2014). A fully discrete semi-Lagrangian scheme for a first order mean field game problem. *SIAM Journal on Numerical Analysis*, 52(1), 45-67.

[8] Carlini, E., and Silva, F. J. (2015). A semi-Lagrangian scheme for a degenerate second order mean field game system. *Discrete & Continuous Dynamical Systems* 35.9: 4269.

MFG Setup

- Scheme introduced by [Carlini, Silva][7]
- For simplicity: $d = 1$, domain $\mathcal{S} = \mathbb{R}$, $\mathcal{A} = \mathbb{R}$
- $\nu = 0$ (degenerate second order case also possible; see [Carlini, Silva][8])
- Model:

$$b(x, m, v) = v$$
$$f(x, m, v) = \frac{1}{2}|v|^2 + f_0(x, m), \qquad g(x, m)$$

where f_0 and g depend on $m \in \mathcal{P}_1(\mathbb{R})$ in a potentially non-local way

- MFG PDE system:

$$\begin{cases} -\dfrac{\partial u}{\partial t}(t,x) + \dfrac{1}{2}|\nabla u(t,x)|^2 = f_0(x, m(t,\cdot)), & \text{in } [0,T) \times \mathbb{R}, \\ \dfrac{\partial m}{\partial t}(t,x) - \operatorname{div}\left(m(t,\cdot)\nabla u(t,\cdot)\right)(x) = 0, & \text{in } (0,T] \times \mathbb{R}, \\ u(T,x) = g(x, m(T,\cdot)), \qquad m(0,x) = m_0(x), & \text{in } \mathbb{R}. \end{cases}$$

[7] Carlini, E., & Silva, F. J. (2014). A fully discrete semi-Lagrangian scheme for a first order mean field game problem. *SIAM Journal on Numerical Analysis*, 52(1), 45-67.

[8] Carlini, E., and Silva, F. J. (2015). A semi-Lagrangian scheme for a degenerate second order mean field game system. *Discrete & Continuous Dynamical Systems* 35.9: 4269.

Representation of the Value Function

- Dynamics:

$$X_t^v = X_0^v + \int_0^t v(s)ds, \qquad t \geq 0.$$

- **Representation formula** for the value function given $m = (m_t)_{t\in[0,T]}$:

$$u[m](t,x) = \inf_{v\in L^2([t,T];\mathbb{R})} \left\{ \int_t^T \left[\frac{1}{2}|v(s)|^2 + f_0(X_s^{v,t,x}, m(s,\cdot))\right] ds + g(X_T^{v,t,x}, m(T,\cdot)) \right\},$$

where $X^{v,t,x}$ starts from x at time t and is controlled by v

Discrete HJB equation

Discrete HJB: Given a flow of densities m,

$$\begin{cases} U_i^n = S_{\Delta t,h}[m](U^{n+1}, i, n), & (n,i) \in [\![N_T - 1]\!] \times \mathbb{Z}, \\ U_i^{N_T} = g(x_i, m(T,\cdot)), & i \in \mathbb{Z}, \end{cases}$$

where

- $S_{\Delta t,h}$ is defined as

$$S_{\Delta t,h}[m](W,n,i) = \inf_{v\in\mathbb{R}} \left\{ \left(\frac{1}{2}|v|^2 + f_0(x_i, m(t_n,\cdot)) \right) \Delta t + I[W](x_i + v\,\Delta t) \right\},$$

Discrete HJB equation

Discrete HJB: Given a flow of densities m,

$$\begin{cases} U_i^n = S_{\Delta t,h}[m](U^{n+1}, i, n), & (n,i) \in [\![N_T - 1]\!] \times \mathbb{Z}, \\ U_i^{N_T} = g(x_i, m(T,\cdot)), & i \in \mathbb{Z}, \end{cases}$$

where

- $S_{\Delta t,h}$ is defined as

$$S_{\Delta t,h}[m](W,n,i) = \inf_{v\in\mathbb{R}} \left\{ \left(\frac{1}{2}|v|^2 + f_0(x_i, m(t_n,\cdot)) \right) \Delta t + I[W](x_i + v\,\Delta t) \right\},$$

- with $I : \mathcal{B}(\mathbb{Z}) \to \mathcal{C}_b(\mathbb{R})$ is the **interpolation operator** defined as

$$I[W](\cdot) = \sum_{i\in\mathbb{Z}} W_i \beta_i(\cdot),$$

- where $\mathcal{B}(\mathbb{Z})$ is the set of bounded functions from $\mathbb{Z}$ to $\mathbb{R}$
- and $\beta_i = \left[1 - \frac{|x-x_i|}{h}\right]_+$: triangular function with support $[x_{i-1}, x_{i+1}]$ and s.t. $\beta_i(x_i) = 1$.

Discrete HJB equation – cont.

Before moving to the KFP equation:

- **Interpolation:** from $U = (U_i^n)_{n,i}$, construct the function $u_{\Delta t,h}[m](x,t) : [0,T] \times \mathbb{R} \to \mathbb{R}$,

$$u_{\Delta t,h}[m](t,x) = I[U^{[\frac{t}{\Delta t}]}](x), \qquad (t,x) \in [0,T] \times \mathbb{R}.$$

Discrete HJB equation – cont.

Before moving to the KFP equation:

- **Interpolation:** from $U = (U_i^n)_{n,i}$, construct the function $u_{\Delta t,h}[m](x,t) : [0,T] \times \mathbb{R} \to \mathbb{R}$,

$$u_{\Delta t,h}[m](t,x) = I[U^{[\frac{t}{\Delta t}]}](x), \qquad (t,x) \in [0,T] \times \mathbb{R}.$$

- **Regularization of HJB solution** with a mollifier ρ_ϵ:

$$u^{\epsilon}_{\Delta t,h}[m](t,\cdot) = \rho_\epsilon * u_{\Delta t,h}[m](t,\cdot), \qquad t \in [0,T].$$

Discrete KFP equation: intuition

- **Eulerian** viewpoint:
  - focus on a location
  - look at the flow passing through it
  - evolution characterized by the velocity at (t, x)
- **Lagrangian** viewpoint:
  - focus on a fluid parcel
  - look at how it flows
  - evolution characterized by the position at time t of a particle starting at x

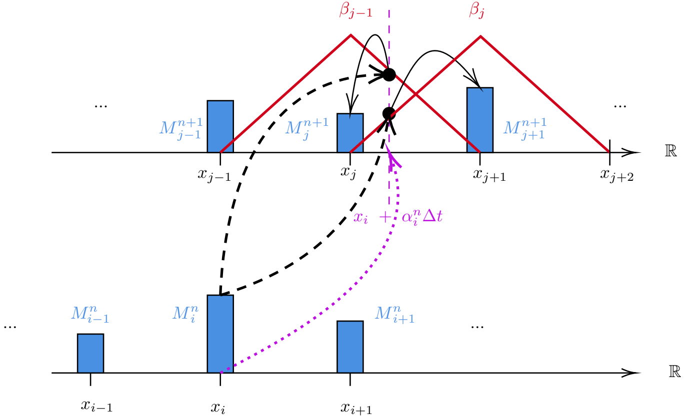
Discrete KFP equation: intuition

- **Eulerian** viewpoint:
  - focus on a location
  - look at the flow passing through it
  - evolution characterized by the velocity at (t, x)
- **Lagrangian** viewpoint:
  - focus on a fluid parcel
  - look at how it flows
  - evolution characterized by the position at time t of a particle starting at x
- Here, in our model:

$$X_t^v = X_0^v + \int_0^t v(s)ds, \qquad t \geq 0.$$

- Time and space discretization?

Discrete KFP equation: intuition – diagram



Movement of the mass when using control $v(t_n, x_i) = \alpha_i^n$.
Bottom: time t_n; top: time t_{n+1}.

Discrete KFP equation

- **Control** induced by value function:

$$\hat{v}^{\epsilon}_{\Delta t,h}[m](t,x) = -\nabla u^{\epsilon}_{\Delta t,h}[m](t,x),$$

and its discrete counter part: $\hat{v}^{\epsilon}_{n,i} = \hat{v}^{\epsilon}_{\Delta t,h}[m](t_n, x_i)$.

- **Discrete flow:**

$$\Phi^{\epsilon}_{n,n+1,i}[m] = x_i + \hat{v}^{\epsilon}_{\Delta t,h}[m](t_n, x_i)\Delta t\,.$$

Discrete KFP equation

- **Control** induced by value function:

$$\hat{v}^{\epsilon}_{\Delta t,h}[m](t,x) = -\nabla u^{\epsilon}_{\Delta t,h}[m](t,x),$$

  and its discrete counter part: $\hat{v}^{\epsilon}_{n,i} = \hat{v}^{\epsilon}_{\Delta t,h}[m](t_n, x_i)$.

- **Discrete flow:**

$$\Phi^{\epsilon}_{n,n+1,i}[m] = x_i + \hat{v}^{\epsilon}_{\Delta t,h}[m](t_n, x_i)\Delta t\,.$$

- **Discrete KFP equation:** for $M^{\epsilon}[m] = (M^{\epsilon,n}_i[m])_{n,i}$:

$$\begin{cases} M^{\epsilon,n+1}_i[m] = \sum_j \beta_i\left(\Phi^{\epsilon}_{n,n+1,j}[m]\right) M^{\epsilon,n}_j[m], & (n,i) \in [\![N_T - 1]\!] \times \mathbb{Z}, \\ M^{\epsilon,0}_i[m] = \displaystyle\int_{[x_i-h/2, x_i+h/2]} m_0(x)dx, & i \in \mathbb{Z}. \end{cases}$$

Fixed Point Formulation

- **Function** $m^{\epsilon}_{\Delta t,h}[m] : [0,T] \times \mathbb{R} \to \mathbb{R}$ defined as: for $n \in [\![N_T - 1]\!]$, for $t \in [t_n, t_{n+1})$,

$$m^{\epsilon}_{\Delta t,h}[m](t,x) = \frac{1}{h}\left[\frac{t_{n+1}-t}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x) + \frac{t-t_n}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n+1}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x)\right].$$

Fixed Point Formulation

- **Function** $m^{\epsilon}_{\Delta t,h}[m] : [0,T] \times \mathbb{R} \to \mathbb{R}$ defined as: for $n \in [\![N_T - 1]\!]$, for $t \in [t_n, t_{n+1})$,

$$m^{\epsilon}_{\Delta t,h}[m](t,x) = \frac{1}{h}\left[\frac{t_{n+1}-t}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x)\right.$$
$$\left. + \frac{t-t_n}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n+1}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x)\right].$$

- **Goal: Fixed-point problem:** Find $\hat{M} = (\hat{M}_i^n)_{i,n}$ such that:

$$\hat{M}_i^n = M_i^n\left[m^{\epsilon}_{\Delta t,h}[\hat{M}]\right].$$

Fixed Point Formulation

- **Function** $m^{\epsilon}_{\Delta t,h}[m] : [0,T] \times \mathbb{R} \to \mathbb{R}$ defined as: for $n \in [\![N_T - 1]\!]$, for $t \in [t_n, t_{n+1})$,

$$m^{\epsilon}_{\Delta t,h}[m](t,x) = \frac{1}{h}\left[\frac{t_{n+1}-t}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x) \right.$$
$$\left. + \frac{t-t_n}{\Delta t}\sum_{i\in\mathbb{Z}} M_i^{\epsilon,n+1}[m]\mathbf{1}_{[x_i-h/2,x_i+h/2]}(x)\right].$$

- **Goal: Fixed-point problem:** Find $\hat{M} = (\hat{M}_i^n)_{i,n}$ such that:

$$\hat{M}_i^n = M_i^n\left[m^{\epsilon}_{\Delta t,h}[\hat{M}]\right].$$

- **Solution strategy:** Fixed point iterations for example
- See [Carlini, Silva] for more details

Numerical Illustration

Costs:

$$g \equiv 0, \qquad f(x, m, v) = \frac{1}{2}|v|^2 + (x - c^*)^2 + \kappa_{MF} V(x, m),$$

with

$$V(x, m) = \rho_{\sigma_V} * \big(\rho_{\sigma_V} * m\big)(x),$$

Numerical Illustration

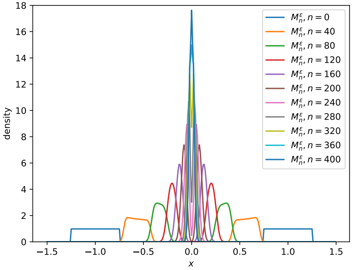
Costs:

$$g \equiv 0, \qquad f(x, m, v) = \frac{1}{2}|v|^2 + (x - c^*)^2 + \kappa_{MF} V(x, m),$$

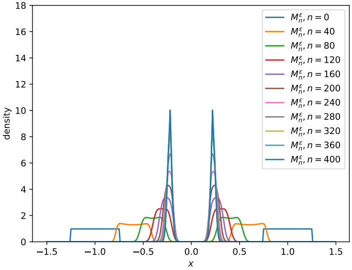
with

$$V(x, m) = \rho_{\sigma_V} * \big(\rho_{\sigma_V} * m\big)(x),$$

Experiments: target $c^* = 0$, m_0 = unif. on $[-1.25, -0.75]$ and on $[0.75, 1.25]$



$\kappa_{MF} = 0.5$



$\kappa_{MF} = 0.9$

Summary