Numerical Methods for Mean Field Games

Lecture 2
Classical Numerical Methods – Part I
Linear-Quadratic MFGs

Mathieu LAURIÈRE

New York University Shanghai

UM6P Vanguard Center, Université Cadi AYYAD,
University Côte d'Azur, & GE2MI
Open Doctoral Lectures
July 5 – 7, 2023

Outline

Continuous time, continuous space MFG

- Time horizon $T < +\infty$, $t \in [0, T]$
- Player's control (deterministic) α_t, typically:
  - closed-loop Markovian: $\alpha_t = \alpha(t, X_t)$
  - open-loop: $\alpha_t = \alpha(t, \omega)$ progressively measurable
- Player's dynamics:

$$dX_t = b(t, X_t, \alpha_t, m_t)dt + \sigma dW_t, \qquad X_0 \sim m_0$$

- Population dynamics: Kolmogorov-Fokker-Planck equation

$$\partial_t m(t,x) - \frac{\sigma^2}{2}\Delta m(t,x) + \mathrm{div}(b(t,x,\alpha(t,x))m(t,x)) = 0, \qquad m_{|t=0} = m_0$$

- To stress the dependence on the control, we will sometimes write X^α and m^α.

Continuous time, continuous space MFG

Cost: dependence on the mean field

- non-local (typically "regularizing" operator)

$$f(t, X_t, \alpha_t, m_t)$$

- local (if the population distribution has a density, still denoted by m)

$$f(t, X_t, \alpha_t, m(t, X_t))$$

HJB equation

- Hamiltonian:

$$H(x,m,p) = \max_a -L(x,a,m,p), \quad L(x,a,m,p) = f(x,a,m) + b(x,a,m) \cdot p$$

- Hamilton-Jacobi-Bellman equation, given the mean field flow:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) + H(x, m(t), \nabla u(t,x))) = 0, \\ u(T,x) = g(x, m(T)) \end{cases}$$

- Recovering the optimal control: optimizer of the Hamiltonian

- Unique action minimizes H under strict convexity assumptions

HJB equation

- Hamiltonian:

$$H(x,m,p) = \max_a -L(x,a,m,p), \quad L(x,a,m,p) = f(x,a,m) + b(x,a,m)\cdot p$$

- Hamilton-Jacobi-Bellman equation, given the mean field flow:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) + H(x, m(t), \nabla u(t,x))) = 0, \\ u(T,x) = g(x, m(T)) \end{cases}$$

- Recovering the optimal control: optimizer of the Hamiltonian

- Unique action minimizes H under strict convexity assumptions

- **Warning:** Another convention: $H(x,m,p) = \min_a L(x,a,m,p) \Rightarrow -H$ in HJB.

Forward-backward PDE system for MFG

The equilibrium control minimizes the Hamiltonian:

$$\hat{\alpha}(t,x) = \underset{a}{\operatorname{argmax}} -L(x, a, m(t), \nabla u(t,x))$$

where (m,u) solve the forward-backward PDE system:

- Forward equation for the mean field:

$$\begin{cases} \partial_t m(t,x) - \frac{\sigma^2}{2}\Delta m(t,x) + \operatorname{div}(m(t,x)H_p(x, m(t), \nabla u(t,x))) = 0, \\ m(0,x) = m_0(x) \end{cases}$$

- Backward equation for the value function:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) + H(x, m(t), \nabla u(t,x))) = 0, \\ u(T,x) = g(x, m(T)) \end{cases}$$

Challenge: We cannot (fully) solve one equation before the other!

Exercises

Exercise

For the following drift and running cost functions ($d = 1$ to simplicity), write the KFP equation, the Hamiltonian and the HJB equation:

- Linear-quadratic (LQ):

$$b(x,a,m) = Ax+Ba+\bar{A}\bar{m}^2, f(x,a,m) = Qx^2+Ra^2+\bar{Q}\bar{m}^2, g(x,m) = Q_T x^2+\bar{Q}_T\bar{m}^2$$

  with $\bar{m} = \int \xi m(\xi) d\xi$
- Congestion: $b(x,a,m) = a$, $f(x,a,m) = m(x)|a|^2$
- Aversion: $b(x,a,m) = a$, $f(x,a,m) = |a|^2 + m(x)$

Exercise

Derive optimality conditions for the social optimum problem.

Social optimum: Mean Field Control

The social optimum problem is referred to as

- mean field (type) control
- control of McKean-Vlasov (MKV) dynamics

Definition (Mean field control (MFC) problem)

α^* is a solution to the MFC problem if it minimizes

$$J^{MFC}(\alpha) = \mathbb{E}\left[\int_0^T f(X_t^\alpha, \alpha_t, m_t^\alpha)dt + g(X_T^\alpha, m_T^\alpha)\right].$$

Main difference with MFG: here not only X but m too is controlled by α.

Social optimum: Mean Field Control

The social optimum problem is referred to as

- mean field (type) control
- control of McKean-Vlasov (MKV) dynamics

Definition (Mean field control (MFC) problem)

α^* is a solution to the MFC problem if it minimizes

$$J^{MFC}(\alpha) = \mathbb{E}\left[\int_0^T f(X_t^\alpha, \alpha_t, m_t^\alpha)dt + g(X_T^\alpha, m_T^\alpha)\right].$$

Main difference with MFG: here not only X but m too is controlled by α.

Optimality conditions? Several approaches:

- Dynamic programming value function depending on m; value function V
- Calculus of variations taking m as a state; adjoint state u
- Pontryagin's maximum principle for the (MKV process) X; adjoint state Y

Forward-backward PDE system for MFC

Approach by calculus of variations, assuming that X has a density in L^2. The optimal control minimizes the Hamiltonian:

$$\alpha^*(t,x) = \underset{a}{\operatorname{argmax}} -L(t,x,a,\nabla u(t,x))$$

where (m,u) solve the forward-backward PDE system:

- Forward equation for the mean field:

$$\begin{cases} \partial_t m(t,x) - \frac{\sigma^2}{2}\Delta m(t,x) + \operatorname{div}(m(t,x)H_p(x,m(t),\nabla u(t,x))) = 0, \\ m(0,x) = m_0(x) \end{cases}$$

- Backward equation for the ~~value function~~ adjoint state:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) + H(x,m(t),\nabla u(t,x))) \\ \qquad\qquad + \displaystyle\int \partial_m H(\xi,m(t),\nabla u(t,\xi))(x)m(t,\xi)d\xi = 0, \\ u(T,x) = g(x,m(T)) + \int \partial_m g(\xi,m(T))(x)m(t,\xi)d\xi \end{cases}$$

where $\partial_m H$ denotes the derivative wrt m, so that for a differentiable $\varphi : L^2(\mathbb{R}^d) \to \mathbb{R}$,

$$\frac{d}{d\theta}\varphi(m+\theta\tilde{m})\Big|_{\theta=0} = \int \partial_m\varphi(m)(\xi)\tilde{m}(\xi)d\xi.$$

See e.g. [Bensoussan et al., 2013], Section 4.1.

Outline

1. Introduction

2. Linear-Quadratic Setting

3. Algorithms

4. Preview of numerical schemes for the PDE system

5. Conclusion

Linear-Quadratic (LQ) Setting

In this section, we are going to focus on the following example.

Example (Linear-Quadratic (LQ) Setting)

$$\begin{aligned}
b(x,a,m) &= Ax + \bar{A}\bar{m} + Ba \\
f(x,a,m) &= \frac{1}{2}\left[x^\top Q x + (x - S\bar{m})^\top \bar{Q}(x - S\bar{m}) + a^\top C a\right] \\
g(x,m) &= \frac{1}{2}\left[x^\top Q_T x + (x - S_T\bar{m})^\top \bar{Q}_T(x - S_T\bar{m})\right] \\
\bar{m} &= \int \xi m(\xi) d\xi
\end{aligned}$$

where $A, \bar{A}, \ldots$ are constant matrices of suitable dimensions.

Linear-Quadratic (LQ) Setting

In this section, we are going to focus on the following example.

Example (Linear-Quadratic (LQ) Setting)

$$
\begin{aligned}
b(x,a,m) &= Ax + \bar{A}\bar{m} + Ba \\
f(x,a,m) &= \frac{1}{2}\left[x^\top Q x + (x - S\bar{m})^\top \bar{Q}(x - S\bar{m}) + a^\top C a\right] \\
g(x,m) &= \frac{1}{2}\left[x^\top Q_T x + (x - S_T\bar{m})^\top \bar{Q}_T(x - S_T\bar{m})\right] \\
\bar{m} &= \int \xi m(\xi) d\xi
\end{aligned}
$$

where $A, \bar{A}, \ldots$ are constant matrices of suitable dimensions.

So:

- The interactions are only through the mean.
- The drift is linear in the state, the action and the mean.
- The costs are quadratic in these variables.

Key point: MFG equilibrium can be computed with ODEs. No need to solve PDEs.

HJB equation

For simplicity, consider the case $d = 1$.

Hamiltonian:

$$H(x,m,p) = \max_a -L(x,a,m,p), \quad L(x,a,m,p) = f(x,a,m) + b(x,a,m)\cdot p$$

Here

$$L(x,a,m,p) = \frac{1}{2}(Qx^2 + \bar{Q}(x - S\bar{m})^2 + Ca^2) + (Ax + \bar{A}\bar{m} + Ba)p$$

The optimal a satisfies (first order optimality condition):

$$Ca + Bp = 0 \Rightarrow a = -\frac{B}{C}p$$

So

$$\begin{aligned} H(x,m,p) &= -[\frac{1}{2}(Qx^2 + \bar{Q}(x - S\bar{m})^2 + \frac{B^2}{C}p^2) + (Ax + \bar{A}\bar{m} - \frac{B^2}{C}p)p] \\ &= -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m})^2] - [Ax + \bar{A}\bar{m}]p + \frac{B^2}{2C}p^2 \end{aligned}$$

and $H_p(x,m,p) = -[Ax + \bar{A}\bar{m}] + \frac{B^2}{C}p$

For simplicity, consider the case $d = 1$.

Hamiltonian:

$$H(x,m,p) = \max_a -L(x,a,m,p), \quad L(x,a,m,p) = f(x,a,m) + b(x,a,m)\cdot p$$

Here

$$L(x,a,m,p) = \frac{1}{2}(Qx^2 + \bar{Q}(x - S\bar{m})^2 + Ca^2) + (Ax + \bar{A}\bar{m} + Ba)p$$

The optimal a satisfies (first order optimality condition):

$$Ca + Bp = 0 \Rightarrow a = -\frac{B}{C}p$$

So

$$\begin{aligned} H(x,m,p) &= -[\frac{1}{2}(Qx^2 + \bar{Q}(x - S\bar{m})^2 + \frac{B^2}{C}p^2) + (Ax + \bar{A}\bar{m} - \frac{B^2}{C}p)p] \\ &= -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m})^2] - [Ax + \bar{A}\bar{m}]p + \frac{B^2}{2C}p^2 \end{aligned}$$

and $H_p(x,m,p) = -[Ax + \bar{A}\bar{m}] + \frac{B^2}{C}p$

Hamilton-Jacobi-Bellman equation:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) \\ \qquad -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m}_t)^2] - [Ax + \bar{A}\bar{m}_t]\nabla u(t,x) + \frac{B^2}{2C}|\nabla u(t,x)|^2 = 0, \\ u(T,x) = Q_T x^2 + \bar{Q}_T(x - S\bar{m}(T))^2 \end{cases}$$

HJB equation: solution

Hamilton-Jacobi-Bellman equation:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) \\ \qquad -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m}_t)^2] - [Ax + \bar{A}\bar{m}_t]\nabla u(t,x) + \frac{B^2}{2C}|\nabla u(t,x)|^2 = 0, \\ u(T,x) = Q_T x^2 + \bar{Q}_T(x - S\bar{m}(T))^2 \end{cases}$$

First remark: The value function has a special form (ansatz):

$$u(t,x) = \frac{1}{2}p_t x^2 + r_t x + s_t,$$

with $p, r, s : [0,T] \to \mathbb{R}$ to be determined. We have:

- $\partial_t u(t,x) = \frac{1}{2}\dot{p}_t x^2 + \dot{r}_t x + \dot{s}_t$
- $\nabla u(t,x) = p_t x + r_t$, and $\Delta u(t,x) = p_t$

HJB equation: solution

Hamilton-Jacobi-Bellman equation:

$$\begin{cases} -\partial_t u(t,x) - \frac{\sigma^2}{2}\Delta u(t,x) \\ \qquad -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m}_t)^2] - [Ax + \bar{A}\bar{m}_t]\nabla u(t,x) + \frac{B^2}{2C}|\nabla u(t,x)|^2 = 0, \\ u(T,x) = Q_T x^2 + \bar{Q}_T(x - S\bar{m}(T))^2 \end{cases}$$

First remark: The value function has a special form (ansatz):

$$u(t,x) = \frac{1}{2}p_t x^2 + r_t x + s_t,$$

with $p, r, s : [0,T] \to \mathbb{R}$ to be determined. We have:

- $\partial_t u(t,x) = \frac{1}{2}\dot{p}_t x^2 + \dot{r}_t x + \dot{s}_t$
- $\nabla u(t,x) = p_t x + r_t$, and $\Delta u(t,x) = p_t$

Second remark: This equation depends on m only through $\bar{m}$. We do not need the full KFP equation

$$\partial_t m(t,x) - \frac{\sigma^2}{2}\Delta m(t,x) + \mathrm{div}(m(t,x)H_p(x, m(t), \nabla u(t,x))) = 0$$

but only the ODE for the mean, obtained by integrating the KFP:

$$\frac{d\bar{m}}{dt} - \int m(t,x)H_p(x, m(t), \nabla u(t,x))dx = 0,$$

Note: $\int m(t,x)H_p(x, m(t), \nabla u(t,x))dx = -[A\bar{m}_t + \bar{A}\bar{m}_t] + \frac{B^2}{C}[p_t\bar{m}_t + r_t]$

Forward-backward ODE system for MFG

Consequence: the MFG solution is given by:

$$\begin{cases} \text{Mean:} & \bar{m}_t^{\hat{\alpha}} = z_t, \\ \text{Control:} & \hat{\alpha}(t,x) = -\frac{B}{C}(p_t x + r_t), \\ \text{Value function:} & u(t,x) = \frac{1}{2} p_t x^2 + r_t x + s_t, \end{cases}$$

Forward-backward ODE system for MFG

Consequence: the MFG solution is given by:

$$\begin{cases} \text{Mean:} & \bar{m}_t^{\hat{\alpha}} = z_t, \\ \text{Control:} & \hat{\alpha}(t,x) = -\dfrac{B}{C}(p_t x + r_t), \\ \text{Value function:} & u(t,x) = \dfrac{1}{2} p_t x^2 + r_t x + s_t, \end{cases}$$

where (z, p, r, s) solve the following system of ordinary differential equations (ODEs):

$$\begin{cases} \dfrac{dz}{dt} = (A + \bar{A} - B^2 C^{-1} p_t) z_t - B^2 C^{-1} r_t, & z_0 = \bar{m}_0, \\ -\dfrac{dp}{dt} = 2A p_t - B^2 C^{-1} p_t^2 + Q + \bar{Q}, & p_T = Q_T + \bar{Q}_T, \\ -\dfrac{dr}{dt} = (A - B^2 C^{-1} p_t) r_t + (p_t \bar{A} - \bar{Q} S) z_t, & r_T = -\bar{Q}_T S_T z_T, \\ -\dfrac{ds}{dt} = \nu p_t - \dfrac{1}{2} B^2 C^{-1} r_t^2 + r_t \bar{A} z_t + \dfrac{1}{2} S^2 \bar{Q} z_t^2, & s_T = \dfrac{1}{2} \bar{Q}_T S_T^2 z_T^2. \end{cases}$$

Forward-backward ODE system for MFG

Consequence: the MFG solution is given by:

$$\begin{cases} \text{Mean:} & \bar{m}_t^{\hat{\alpha}} = z_t, \\ \text{Control:} & \hat{\alpha}(t,x) = -\frac{B}{C}(p_t x + r_t), \\ \text{Value function:} & u(t,x) = \frac{1}{2} p_t x^2 + r_t x + s_t, \end{cases}$$

where (z, p, r, s) solve the following system of ordinary differential equations (ODEs):

$$\begin{cases} \frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1} p_t) z_t - B^2 C^{-1} r_t, & z_0 = \bar{m}_0, \\ -\frac{dp}{dt} = 2A p_t - B^2 C^{-1} p_t^2 + Q + \bar{Q}, & p_T = Q_T + \bar{Q}_T, \\ -\frac{dr}{dt} = (A - B^2 C^{-1} p_t) r_t + (p_t \bar{A} - \bar{Q} S) z_t, & r_T = -\bar{Q}_T S_T z_T, \\ -\frac{ds}{dt} = \nu p_t - \frac{1}{2} B^2 C^{-1} r_t^2 + r_t \bar{A} z_t + \frac{1}{2} S^2 \bar{Q} z_t^2, & s_T = \frac{1}{2} \bar{Q}_T S_T^2 z_T^2. \end{cases}$$

Key points:

- coupling between z and r
- **forward-backward** structure

LQ MFC

We can apply the same strategy to the MFC PDE system.
Recall:

$$H(x,m,p) = -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m})^2] - [Ax + \bar{A}\bar{m}]p + \frac{B^2}{2C}p^2$$

So:

$$\begin{aligned}\frac{d}{d\theta}H(x, m + \theta\bar{\tilde{m}}, p)\Big|_{\theta=0} &= [\bar{Q}(x - S\bar{m})S\bar{\tilde{m}}] - [\bar{A}\bar{\tilde{m}}]p \\ &= \int \left[\bar{Q}(x - S\bar{m})S - \bar{A}p\right] \xi\tilde{m}(\xi)d\xi\end{aligned}$$

LQ MFC

We can apply the same strategy to the MFC PDE system.
Recall:

$$H(x,m,p) = -\frac{1}{2}[Qx^2 + \bar{Q}(x - S\bar{m})^2] - [Ax + \bar{A}\bar{m}]p + \frac{B^2}{2C}p^2$$

So:

$$\begin{aligned}\frac{d}{d\theta}H(x, m + \theta\bar{\tilde{m}}, p)\Big|_{\theta=0} &= [\bar{Q}(x - S\bar{m})S\bar{\tilde{m}}] - [\bar{A}\bar{\tilde{m}}]p \\ &= \int \left[\bar{Q}(x - S\bar{m})S - \bar{A}p\right] \xi\tilde{m}(\xi)d\xi\end{aligned}$$

Hence, by definition, $\partial_m H(x,m,p)(\xi) = [\bar{Q}(x - S\bar{m})S - \bar{A}p]\xi$, and thus (swap x and ξ)

$$\begin{aligned}\int \partial_m H(\xi, m, \nabla u(t,\xi))(x)m(\xi)d\xi &= \int [\bar{Q}(\xi - S\bar{m})S - \bar{A}\nabla u(t,\xi)]xm(\xi)d\xi \\ &= \left[\bar{Q}(S - S^2)\bar{m} - \bar{A}\int \nabla u(t,\xi)m(\xi)d\xi\right]x \\ &= \left[\bar{Q}(S - S^2)\bar{m} - \bar{A}(\check{p}_t\bar{m}_t + \check{r}_t)\right]x\end{aligned}$$

where we use an ansatz $u(t,x) = \frac{1}{2}\check{p}_t x^2 + \check{r}_t x + \check{s}_t$

Forward-backward ODE system for MFC

We obtain that the MFC optimum is given by:

$$
\begin{cases}
\text{Mean:} & \bar{m}_t^{\alpha^*} = \check{z}_t, \\
\text{Control:} & \alpha^*(t,x) = -\dfrac{B}{C}(\check{p}_t x + \check{r}_t), \\
\text{Value:} & J^{MFC}(\alpha^*) = \dfrac{1}{2}\check{p}_0(\sigma_0^2 + \bar{m}_0^2) + \check{r}_0\bar{m}_0 + \check{s}_0 + (1 - S_T)\bar{Q}_T S_T \check{z}_T^2 \\
& \qquad - \displaystyle\int_0^T \left[(\check{p}_t\check{z}_t + \check{r}_t)\bar{A}\check{z}_t - (1 - S_t)\bar{Q}S\check{z}_t^2\right] dt
\end{cases}
$$

Forward-backward ODE system for MFC

We obtain that the MFC optimum is given by:

$$\begin{cases} \text{Mean:} & \bar{m}_t^{\alpha^*} = \check{z}_t, \\ \text{Control:} & \alpha^*(t,x) = -\frac{B}{C}(\check{p}_t x + \check{r}_t), \\ \text{Value:} & J^{MFC}(\alpha^*) = \frac{1}{2}\check{p}_0(\sigma_0^2 + \bar{m}_0^2) + \check{r}_0\bar{m}_0 + \check{s}_0 + (1-S_T)\bar{Q}_T S_T \check{z}_T^2 \\ & \qquad - \int_0^T \left[(\check{p}_t\check{z}_t + \check{r}_t)\bar{A}\check{z}_t - (1-S_t)\bar{Q}S\check{z}_t^2\right] dt \end{cases}$$

where $(\check{z}, \check{p}, \check{r}, \check{s})$ solve the following system of ODEs:

$$\begin{cases} \frac{d\check{z}}{dt} = (A + \bar{A} - B^2C^{-1}\check{p}_t)\check{z}_t - B^2C^{-1}\check{r}_t, & \check{z}_0 = \bar{m}_0, \\ -\frac{d\check{p}}{dt} = 2A\check{p}_t - B^2C^{-1}\check{p}_t^2 + Q + \bar{Q}, & \check{p}_T = Q_T + \bar{Q}_T, \\ -\frac{d\check{r}}{dt} = (A + \bar{A} - B^2C^{-1}\check{p}_t)\check{r}_t + (2\check{p}_t\bar{A} - 2\bar{Q}S + \bar{Q}S^2)\check{z}_t, & \check{r}_T = (-2\bar{Q}_T S_T + \bar{Q}_T S_T^2)\check{z}_T, \\ -\frac{ds}{dt} = \nu\check{p}_t - \frac{1}{2}B^2C^{-1}\check{r}_t^2 + \check{r}_t\bar{A}\check{z}_t + \frac{1}{2}S^2\bar{Q}\check{z}_t^2, & \check{s}_T = \frac{1}{2}\bar{Q}_T S_T^2\check{z}_T^2. \end{cases}$$

Same system as for MFG, except for a few terms

Linear-Quadratic (LQ) Setting

Remarks:

- LQ models are useful because they have (almost) analytical solutions
- The above model is inspired by [Bensoussan et al., 2013], Chapter 6
- It is possible to have much more general LQ MFG models (see e.g., [Huang et al., 2006], [Barreiro-Gomez and Tembine, 2021], [Graber, 2016], . . .)
- Extension with common noise, see e.g. [Carmona et al., 2015, Graber, 2016]
- In some cases, using a different ansatz, the equations can be decoupled, see [Malhamé and Graves, 2020] (AMS'20 minicourse lecture notes)
- The equation for p can be solved by itself; sometimes it has an analytical solution, see e.g. [Carmona and Delarue, 2018], p. 110
- The equation for s can be solved by itself after computing p, z, r
- In the sequel, we focus on computing z and r

Outline

1. Introduction

2. Linear-Quadratic Setting

3. Algorithms
   - Pure Fixed Point Iterations (Banach-Picard)
   - Damped Fixed Point Iterations
   - Fictitious Play
   - Shooting Method
   - Newton Method
   - MFC & Price of Anarchy

4. Preview of numerical schemes for the PDE system

5. Conclusion

Time Discretization

The experiments that follow are borrowed from [Laurière, 2021], Section 2.

In practice, the following algorithms are implemented a discrete time system:

- We focus on the coupled system for (z, r)
- Uniform grid on $[0, T]$, step Δt, $t_n = n \times \Delta t, n = 0, \ldots, N_T$
- Approximate $z, r : [0, T] \to \mathbb{R}$ by vectors $Z, R \in \mathbb{R}^{N_T+1}$

Time Discretization

The experiments that follow are borrowed from [Laurière, 2021], Section 2.

In practice, the following algorithms are implemented a discrete time system:

- We focus on the coupled system for (z, r)
- Uniform grid on $[0, T]$, step Δt, $t_n = n \times \Delta t, n = 0, \dots, N_T$
- Approximate $z, r : [0, T] \to \mathbb{R}$ by vectors $Z, R \in \mathbb{R}^{N_T+1}$
- Discrete ODE system:

$$\begin{cases} \dfrac{Z^{n+1} - Z^n}{\Delta t} = (A + \bar{A} - B^2 C^{-1} P^n) Z^{n+1} - B^2 C^{-1} R^n, \\ Z^0 = \bar{m}_0, \\ -\dfrac{R^{n+1} - R^n}{\Delta t} = (A - B^2 C^{-1} P^n) R^n + (P^n \bar{A} - \bar{Q} S) Z^{n+1}, \\ R^{N_T} = -\bar{Q}_T S_T Z^{N_T}. \end{cases}$$

- To alleviate the notation, most of the algorithms are described using the ODEs

Outline

1. Introduction

2. Linear-Quadratic Setting

3. Algorithms
   - Pure Fixed Point Iterations (Banach-Picard)
   - Damped Fixed Point Iterations
   - Fictitious Play
   - Shooting Method
   - Newton Method
   - MFC & Price of Anarchy

4. Preview of numerical schemes for the PDE system

5. Conclusion

Algorithm 1: Banach-Picard Iterations

Algorithm: Fixed-point iterations

Input: Initial guess $(\tilde{z}, \tilde{r})$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{z}, \hat{r})$

Initialize $z^{(0)} = \tilde{z}, r^{(0)} = \tilde{r}$

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

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

$$-\frac{dr}{dt} = (A - p_t B^2 C^{-1}) r_t + (p_t \bar{A} - \bar{Q} S) z_t^{(\mathtt{k})}, \qquad r_T = -\bar{Q}_T S_T z_T^{(\mathtt{k})}$$

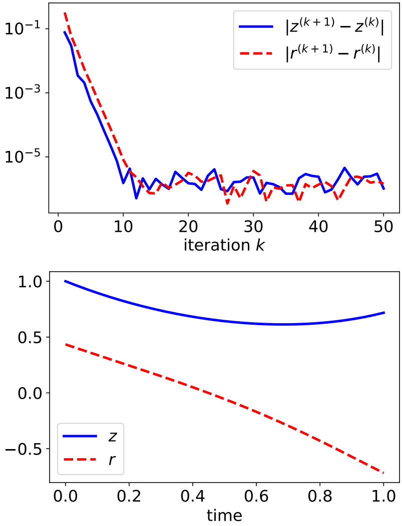
Let $z^{(\mathtt{k}+1)}$ be the solution to:

$$\frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1}) z_t - B^2 C^{-1} r_t^{(\mathtt{k}+1)}, \qquad z_0 = \bar{m}_0$$

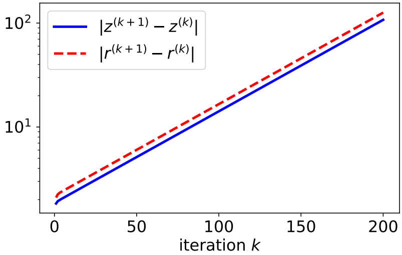
return $(z^{(\mathtt{K})}, r^{(\mathtt{K})})$

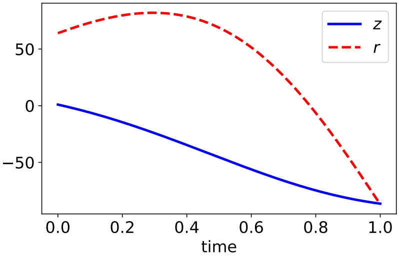
Algorithm 1: Banach-Picard Iterations – Illustration 1

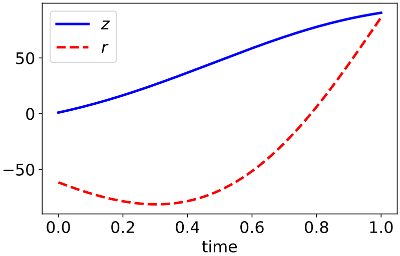
Test case 1 (for the values of $A, \bar{A}, \ldots$, see [Laurière, 2021], Section 2)



Test case 2 (for the values of $A, \bar{A}, \ldots$, see [Laurière, 2021], Section 2)







Algorithm 1: Banach-Picard Iterations – Remarks

- In fact this algorithm is related to a proof technique for the existence and uniqueness of a Nash equilibrium (see lecture 1)
- See e.g. [Huang et al., 2006]
- Here, the approach converges if $z^{(k)} \mapsto r^{(k)} \mapsto z^{(k+1)}$ is a strict contraction
- Typically true if T is small enough or the coefficients are small enough
- Otherwise, it is common to see non-convergence
- Can we “fix” this algorithm?

Outline

Banach-Picard Iterations with Damping

Algorithm: Fixed-point iterations with damping

Input: Initial guess $(\tilde{z}, \tilde{r})$; damping $\delta \in [0, 1)$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{z}, \hat{r})$

Initialize $z^{(0)} = \tilde{z}^{(0)} = \tilde{z}, r^{(0)} = \tilde{r}$

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

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

$$-\frac{dr}{dt} = (A - p_t B^2 C^{-1}) r_t + (p_t \bar{A} - \bar{Q} S) \tilde{z}_t^{(\mathtt{k})}, \qquad r_T = -\bar{Q}_T S_T \tilde{z}_T^{(\mathtt{k})}$$

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

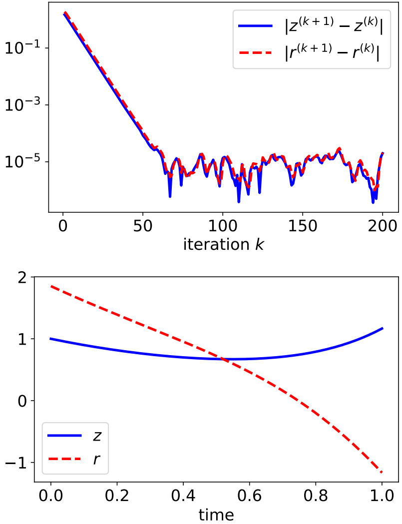
$$\frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1}) z_t - B^2 C^{-1} r_t^{(\mathtt{k}+1)}, \qquad z_0 = \bar{m}_0$$

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

return $(z^{(\mathtt{K})}, r^{(\mathtt{K})})$

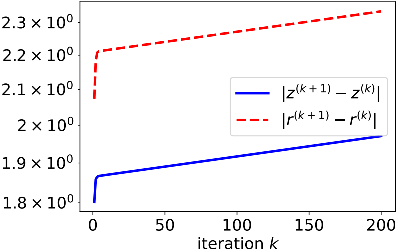
Algorithm 1': Banach-Picard Iterations with Damping – Illustration 1

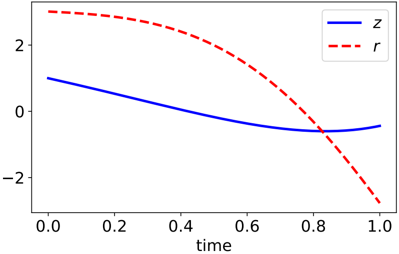
Test case 2
Damping = 0.1

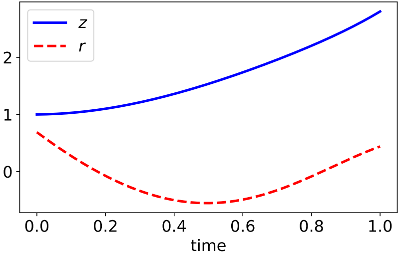


Test case 2

Damping = 0.01







Outline

Algorithm 2: Fictitious Play

- Introduced by [Brown, 1951], [Robinson, 1951]

- Converge proof for several classes of games

- In the MFG setting, FP has been introduced in [Cardaliaguet and Hadikhanloo, 2017], with a proof of convergence for potential MFGs; then extended to MFGs with monotonicity [Hadikhanloo, 2018], [Hadikhanloo and Silva, 2019]

- Related to learning in MFGs: [Perrin et al., 2020] for continuous-time FP under monotonicity condition, [Geist et al., 2022, Lavigne and Pfeiffer, 2022] for discrete-time FP in some potential MFGs; In linear-quadratic MFGs, a rate of convergence has been obtained by [Delarue and Vasileiadis, 2021]

- See Lecture 8 for more details on FP with RL for MFGs

Algorithm: Fictitious Play

Input: Initial guess $(\tilde{z}, \tilde{r})$; number of iterations $\mathtt{K}$

Output: Approximation of $(\hat{z}, \hat{r})$

Initialize $z^{(0)} = \tilde{z}^{(0)} = \tilde{z}, r^{(0)} = \tilde{r}$

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

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

$$-\frac{dr}{dt} = (A - p_t B^2 C^{-1}) r_t + (p_t \bar{A} - \bar{Q} S) \tilde{z}_t^{(\mathtt{k})}, \qquad r_T = -\bar{Q}_T S_T \tilde{z}_T^{(\mathtt{k})}$$

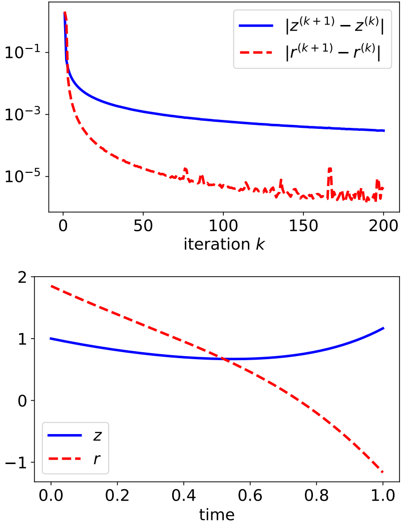
Let $z^{(\mathtt{k}+1)}$ be the solution to:

$$\frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1}) z_t - B^2 C^{-1} r_t^{(\mathtt{k}+1)}, \qquad z_0 = \bar{m}_0$$

Let $\tilde{z}^{(\mathtt{k}+1)} = \frac{\mathtt{k}}{\mathtt{k}+1} \tilde{z}^{(\mathtt{k})} + \frac{1}{\mathtt{k}+1} z^{(\mathtt{k}+1)}$

return $(z^{(\mathtt{K})}, r^{(\mathtt{K})})$

Test case 2


$|z^{(k+1)} - z^{(k)}|$
$|r^{(k+1)} - r^{(k)}|$
10^{-1}
10^{-3}
10^{-5}
0
50
100
150
200
iteration k
2
1
0
−1
z
r
0.0
0.2
0.4
0.6
0.8
1.0
time

Algorithms 1, 1' & 2: Common Framework

Algorithm: General fixed-point iterations

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

Output: Approximation of $(\hat{z}, \hat{r})$

Initialize $z^{(0)} = \tilde{z}^{(0)} = \tilde{z}, r^{(0)} = \tilde{r}$

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

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

$$-\frac{dr}{dt} = (A - p_t B^2 C^{-1}) r_t + (p_t \bar{A} - \bar{Q} S) \tilde{z}_t^{(\mathtt{k})}, \qquad r_T = -\bar{Q}_T S_T \tilde{z}_T^{(\mathtt{k})}$$

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

$$\frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1}) z_t - B^2 C^{-1} r_t^{(\mathtt{k}+1)}, \qquad z_0 = \bar{m}_0$$

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

return $(z^{(\mathtt{K})}, r^{(\mathtt{K})})$

Pure fixed point and Fictitious play are special cases

Remark: Could put the damping on r instead of z.

Outline

Algorithm 3: Shooting Method

- Intuition: *instead of solving a backward equation, choose a starting point and try to shoot for the right terminal point*
- Concretely: replace the forward-backward system

$$\begin{cases} \dfrac{dz}{dt} = (A + \bar{A} - B^2 C^{-1} p_t) z_t - B^2 C^{-1} r_t, & z_0 = \bar{m}_0, \\ -\dfrac{dr}{dt} = (A - B^2 C^{-1} p_t) r_t + (p_t \bar{A} - \bar{Q} S) z_t, & r_T = -\bar{Q}_T S_T z_T \end{cases}$$

by the forward-forward system

$$\begin{cases} \dfrac{d\zeta}{dt} = (A + \bar{A} - B^2 C^{-1} p_t) \zeta_t - B^2 C^{-1} \rho_t, & z_0 = \bar{m}_0, \\ -\dfrac{d\rho}{dt} = (A - B^2 C^{-1} p_t) \rho_t + (p_t \bar{A} - \bar{Q} S) \zeta_t, & \rho_0 = \text{chosen} \end{cases}$$

and try to ensure: $\rho_T = -\bar{Q}_T S_T \zeta_T$

Outline

Algorithm 4: Newton Method – Intuition

Newton method in dimension 1:

- Look for x^* such that: $\mathtt{f}(x^*) = 0$
- Start from initial guess x_0
- Repeat:

$$x_{k+1} = x_k - \frac{\mathtt{f}(x_k)}{\mathtt{f}'(x_k)}$$

Algorithm 4: Newton Method – Intuition

Newton method in dimension 1:

- Look for x^* such that: $\mathtt{f}(x^*) = 0$
- Start from initial guess x_0
- Repeat:

$$x_{k+1} = x_k - \frac{\mathtt{f}(x_k)}{\mathtt{f}'(x_k)}$$

- In high dimension, we avoid computing the inverse of $\mathtt{f}'(x_k)$
- $x_{k+1} = x_k + \tilde{x}_k$, where $\tilde{x}_k$ solves:

$$\mathtt{f}'(x_k)\,\tilde{x}_k = -\mathtt{f}(x_k)$$

  which boils down to solving a linear system

Algorithm 4: Newton Method – Implementation

- Recast the problem:

$$(Z, R) \text{ solve forward-forward discrete system} \Leftrightarrow \mathcal{F}(Z, R) = 0$$

- $\mathcal{F}$ takes into account the initial and terminal conditions

- $D\mathcal{F} =$ differential of this operator

Exercise

Express $\mathcal{F}$ and $D\mathcal{F}$.

Algorithm 4: Newton Method – Implementation

Algorithm: Newton Iterations

Input: Initial guess $(\tilde{Z}, \tilde{R})$; number of iterations $\mathtt{K}$
Output: Approximation of $(\hat{z}, \hat{r})$

```
Initialize (Z^(0), R^(0)) = (Z̃, R̃)
for k = 0, 1, 2, ..., K − 1 do
    Let (Z̃^(k+1), R̃^(k+1)) solve
        DF(Z^(k), R^(k))(Z̃^(k+1), R̃^(k+1)) = −F(Z^(k), R^(k))
    Let (Z^(k+1), R^(k+1)) = (Z̃^(k+1), R̃^(k+1)) + (Z^(k), R^(k))
    return (Z^(K), R^(K))
```

Initialize $(Z^{(0)}, R^{(0)}) = (\tilde{Z}, \tilde{R})$

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

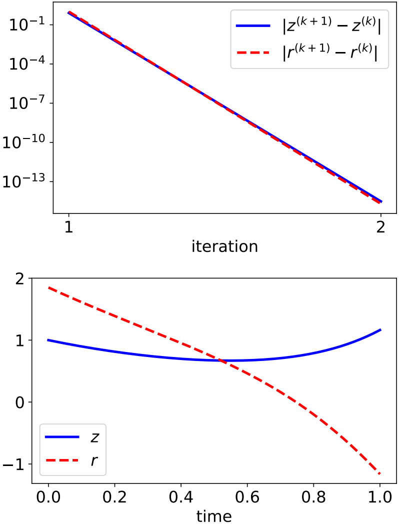
Let $(\tilde{Z}^{(\mathtt{k}+1)}, \tilde{R}^{(\mathtt{k}+1)})$ solve

$$D\mathcal{F}(Z^{(\mathtt{k})}, R^{(\mathtt{k})})(\tilde{Z}^{(\mathtt{k}+1)}, \tilde{R}^{(\mathtt{k}+1)}) = -\mathcal{F}(Z^{(\mathtt{k})}, R^{(\mathtt{k})})$$

Let $(Z^{(\mathtt{k}+1)}, R^{(\mathtt{k}+1)}) = (\tilde{Z}^{(\mathtt{k}+1)}, \tilde{R}^{(\mathtt{k}+1)}) + (Z^{(\mathtt{k})}, R^{(\mathtt{k})})$

return $(Z^{(\mathtt{K})}, R^{(\mathtt{K})})$

Test case 2


10^{-1}
10^{-4}
10^{-7}
10^{-10}
10^{-13}
1
2
iteration
$|z^{(k+1)} - z^{(k)}|$
$|r^{(k+1)} - r^{(k)}|$
2
1
0
−1
0.0
0.2
0.4
0.6
0.8
1.0
time
z
r

Algorithm 4: Newton Method – Explanation

- Reminder: Discrete ODE system:

$$\begin{cases} \dfrac{Z^{n+1}-Z^n}{\Delta t} = (A+\bar{A}-B^2C^{-1}P^n)Z^{n+1} - B^2C^{-1}R^n, \\ Z^0 = \bar{m}_0, \\ -\dfrac{R^{n+1}-R^n}{\Delta t} = (A-B^2C^{-1}P^n)R^n + (P^n\bar{A}-\bar{Q}S)Z^{n+1}, \\ R^{N_T} = -\bar{Q}_T S_T Z^{N_T}. \end{cases}$$

Algorithm 4: Newton Method – Explanation

- Reminder: Discrete ODE system:

$$\begin{cases} \dfrac{Z^{n+1} - Z^n}{\Delta t} = (A + \bar{A} - B^2 C^{-1} P^n) Z^{n+1} - B^2 C^{-1} R^n, \\ Z^0 = \bar{m}_0, \\ -\dfrac{R^{n+1} - R^n}{\Delta t} = (A - B^2 C^{-1} P^n) R^n + (P^n \bar{A} - \bar{Q} S) Z^{n+1}, \\ R^{N_T} = -\bar{Q}_T S_T Z^{N_T}. \end{cases}$$

- Can be rewritten as a linear system:

$$\mathbf{M} \begin{pmatrix} Z \\ R \end{pmatrix} + \mathbf{B} = 0$$

- Newton's method solves a linear system in a single iteration.

- In hindsight: we did not need any of the previous methods! We could have simply used a solver for linear systems of equations.

- The methods were applied in the LQ setting only for pedagogical purposes.

Outline

Price of Anarchy – Illustration

- Introduced by [Koutsoupias and Papadimitriou, 1999]

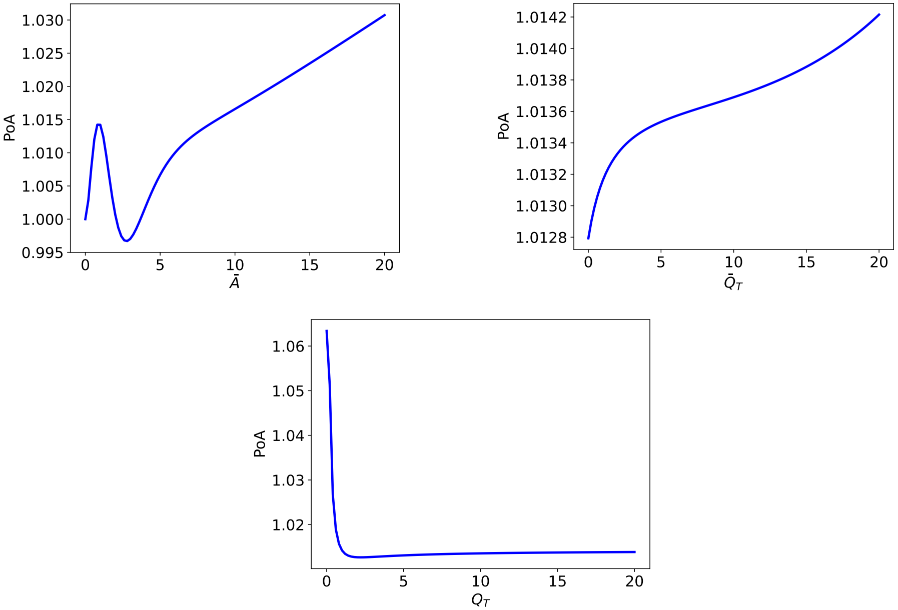
- Extension to MFGs: assuming there exist a unique MFG equilibrium $(\hat{\alpha}, \hat{m})$ and a unique MFC optimum α^*

$$PoA = \frac{J^{MFG}(\hat{\alpha}; \hat{m})}{J^{MFC}(\alpha^*)}$$

- Ratio of the expected cost for a typical player in the MFG by her expected cost in the MFC

- See in particular [Carmona et al., 2019] for explicit computations in the LQ case

Price of Anarchy – Illustration



Sample code

Code

Sample code to illustrate: IPython notebook

`https://colab.research.google.com/drive/1a0TKAnc1Ng5LQ36ZqBPTToJX6oOkoSkd?usp=sharing`

- ODE system for Linear-quadratic MFG
- Solved by fixed point, damped fixed point, fictitious play and Newton's method

Exercises

Exercise

Modify the previous code to solve the ODE system for MFC.

Compute the price of anarchy.

MFG PDE System

Recall the MFG PDE system:

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

Goals:

1. introduce a discrete version of this system $\rightarrow$ numerical scheme
2. solve it numerically $\rightarrow$ algorithm

Properties

For (1): some desirable properties:

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

Properties

For (1): some desirable properties:

- **Mass** and **positivity** of distribution: $\int_{\mathcal{X}} m(t,x)dx = 1$, $m \geq 0$
- **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

Properties

For (1): some desirable properties:

- **Mass** and **positivity** of distribution: $\int_{\mathcal{X}} m(t,x)dx = 1$, $m \geq 0$
- **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?

Two Numerical Schemes

Numerical schemes: We are going to illustrate two approaches:

1. Finite difference scheme introduced in [Achdou and Capuzzo-Dolcetta, 2010]
2. Semi-Lagrangian scheme introduced in [Carlini and Silva, 2014]

There are other options such as finite elements, see e.g. [Benamou and Carlier, 2015, Andreev, 2017].

Outline

Summary

1. Linear-Quadratic MFG and MFC
2. Forward-backward ODE system
3. Several algorithms

Remarks:

1. In the LQ case, these algorithms are just for pedagogical purposes
2. But analogous algorithms can be useful for finite-state MFGs
3. Similarly for continuous-space MFGs up to space-discretization

Thank you for your attention

Questions?

Feel free to reach out: mathieu.lauriere@nyu.edu

References I

[Achdou and Capuzzo-Dolcetta, 2010] Achdou, Y. and Capuzzo-Dolcetta, I. (2010).
Mean field games: numerical methods.
SIAM J. Numer. Anal., 48(3):1136–1162.

[Andreev, 2017] Andreev, R. (2017).
Preconditioning the augmented lagrangian method for instationary mean field games with diffusion.
SIAM Journal on Scientific Computing, 39(6):A2763–A2783.

[Barreiro-Gomez and Tembine, 2021] Barreiro-Gomez, J. and Tembine, H. (2021).
Mean-field-type Games for Engineers.
CRC Press.

[Benamou and Carlier, 2015] Benamou, J.-D. and Carlier, G. (2015).
Augmented lagrangian methods for transport optimization, mean field games and degenerate elliptic equations.
Journal of Optimization Theory and Applications, 167(1):1–26.

[Bensoussan et al., 2013] Bensoussan, A., Frehse, J., and Yam, S. C. P. (2013).
Mean field games and mean field type control theory.
Springer Briefs in Mathematics. Springer, New York.

[Brown, 1951] Brown, G. W. (1951).
Iterative solution of games by fictitious play.
Activity analysis of production and allocation, 13(1):374–376.

References II

[Cardaliaguet and Hadikhanloo, 2017] Cardaliaguet, P. and Hadikhanloo, S. (2017).
Learning in mean field games: the fictitious play.
ESAIM Control Optim. Calc. Var., 23(2):569–591.

[Carlini and Silva, 2014] Carlini, E. and Silva, F. J. (2014).
A fully discrete semi-Lagrangian scheme for a first order mean field game problem.
SIAM J. Numer. Anal., 52(1):45–67.

[Carmona and Delarue, 2018] Carmona, R. and Delarue, F. (2018).
Probabilistic theory of mean field games with applications. I, volume 83 of *Probability Theory and Stochastic Modelling*.
Springer, Cham.
Mean field FBSDEs, control, and games.

[Carmona et al., 2015] Carmona, R., Fouque, J.-P., and Sun, L.-H. (2015).
Mean field games and systemic risk.
Commun. Math. Sci., 13(4):911–933.

[Carmona et al., 2019] Carmona, R., Graves, C. V., and Tan, Z. (2019).
Price of anarchy for mean field games.
In *CEMRACS 2017—numerical methods for stochastic models: control, uncertainty quantification, mean-field*, volume 65 of *ESAIM Proc. Surveys*, pages 349–383. EDP Sci., Les Ulis.

References III

[Delarue and Vasileiadis, 2021] Delarue, F. and Vasileiadis, A. (2021).
Exploration noise for learning linear-quadratic mean field games.
arXiv preprint arXiv:2107.00839.

[Geist et al., 2022] Geist, M., Pérolat, J., Laurière, M., Elie, R., Perrin, S., Bachem, O., Munos, R., and Pietquin, O. (2022).
Concave utility reinforcement learning: The mean-field game viewpoint.
In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, pages 489–497.

[Graber, 2016] Graber, P. J. (2016).
Linear quadratic mean field type control and mean field games with common noise, with application to production of an exhaustible resource.
Applied Mathematics & Optimization, 74:459–486.

[Hadikhanloo, 2018] Hadikhanloo, S. (2018).
Learning in mean field games.
PhD thesis, PSL Research University.

[Hadikhanloo and Silva, 2019] Hadikhanloo, S. and Silva, F. J. (2019).
Finite mean field games: fictitious play and convergence to a first order continuous mean field game.
Journal de Mathématiques Pures et Appliquées, 132:369–397.

References IV

[Huang et al., 2006] Huang, M., Malhamé, R. P., Caines, P. E., et al. (2006).
Large population stochastic dynamic games: closed-loop mckean-vlasov systems and the nash certainty equivalence principle.
Communications in Information & Systems, 6(3):221–252.

[Koutsoupias and Papadimitriou, 1999] Koutsoupias, E. and Papadimitriou, C. (1999).
Worst-case equilibria.
In *Annual symposium on theoretical aspects of computer science*, pages 404–413. Springer.

[Laurière, 2021] Laurière, M. (2021).
Numerical methods for mean field games and mean field type control.
arXiv preprint arXiv:2106.06231.

[Lavigne and Pfeiffer, 2022] Lavigne, P. and Pfeiffer, L. (2022).
Generalized conditional gradient and learning in potential mean field games.
arXiv preprint arXiv:2209.12772.

[Malhamé and Graves, 2020] Malhamé, R. P. and Graves, C. (2020).
Mean field games: A paradigm for individual-mass interactions.
In *Proceedings of Symposia in Applied Mathematics*, volume 78, pages 3–32.

[Perrin et al., 2020] Perrin, S., Pérolat, J., Laurière, M., Geist, M., Elie, R., and Pietquin, O. (2020).
Fictitious play for mean field games: Continuous time analysis and applications.
Advances in Neural Information Processing Systems.

References V

[Robinson, 1951] Robinson, J. (1951).
An iterative method of solving a game.
Annals of mathematics, pages 296–301.