

Mean Field Games: Numerical Methods and Applications in Machine Learning

Part 1

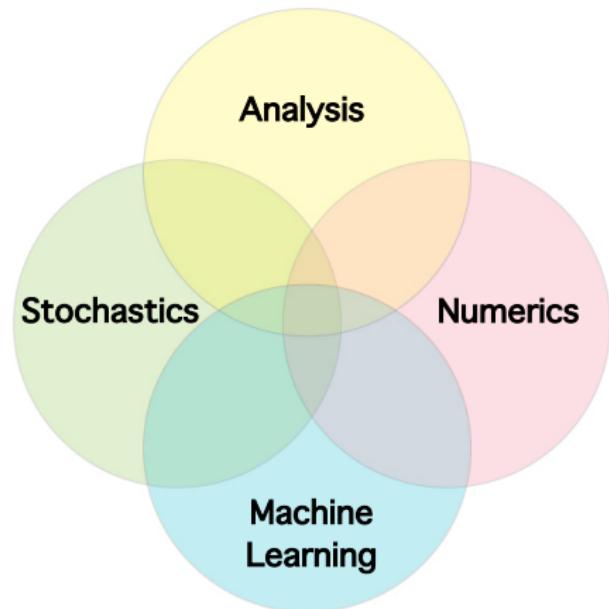
Mathieu LAURIÈRE

<https://mlauriere.github.io/teaching/MFG-PKU-1.pdf>

August, 2021

Outline

1. Introduction
2. From N to infinity
3. Warm-up: LQMFG



Initiated by [Lasry and Lions](#), and [Caines, Huang and Malhamé](#) around 2006

Main research directions:

(1) Modeling: economics, crowd motion, flocking, risk management, smart grid, energy production, distributed robotics, opinion dynamics, epidemics, ...

Initiated by Lasry and Lions, and Caines, Huang and Malhamé around 2006

Main research directions:

(1) Modeling: economics, crowd motion, flocking, risk management, smart grid, energy production, distributed robotics, opinion dynamics, epidemics, ...

(2) Mean field theory: justification of the approximation:

- ◊ N -agent problem → mean field: convergence of equilibria / optimal control
- ◊ N -agent problem ← mean field: ϵ -Nash equilibrium / ϵ -optimality

Initiated by Lasry and Lions, and Caines, Huang and Malhamé around 2006

Main research directions:

(1) Modeling: economics, crowd motion, flocking, risk management, smart grid, energy production, distributed robotics, opinion dynamics, epidemics, ...

(2) Mean field theory: justification of the approximation:

- ◊ N -agent problem → mean field: convergence of equilibria / optimal control
- ◊ N -agent problem ← mean field: ϵ -Nash equilibrium / ϵ -optimality

(3) Characterization of the mean field problems solutions (**optimality conditions**):

- ◊ analytical: partial differential equations (PDEs)
- ◊ probabilistic: stochastic differential equations (SDEs)

Initiated by Lasry and Lions, and Caines, Huang and Malhamé around 2006

Main research directions:

(1) Modeling: economics, crowd motion, flocking, risk management, smart grid, energy production, distributed robotics, opinion dynamics, epidemics, ...

(2) Mean field theory: justification of the approximation:

- ◊ N -agent problem → mean field: convergence of equilibria / optimal control
- ◊ N -agent problem ← mean field: ϵ -Nash equilibrium / ϵ -optimality

(3) Characterization of the mean field problems solutions (**optimality conditions**):

- ◊ analytical: partial differential equations (PDEs)
- ◊ probabilistic: stochastic differential equations (SDEs)

(4) Computation of solutions

- ◊ crucial for applications
- ◊ challenging (coupling between optimization & mean-field)



Key ingredients:

- state
- action
- cost

Games



Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”

Games



Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Games

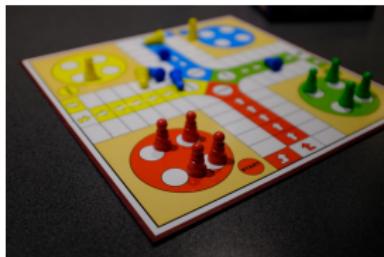


Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Example: 2 players, 2 actions each, matrix of **costs** (to be **minimized**):

		Bob	
		b_1	b_2
Alice		a_1	(4, 6)
		a_2	(7, 5)



Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Example: 2 players, 2 actions each, matrix of **costs** (to be **minimized**):

		Bob	
		b_1	b_2
		a_1	a_2
Alice	a_1	(4, 6), SC = 5	(6, 8), SC = 7
	a_2	(7, 5), SC = 6	

Games



Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Example: 2 players, 2 actions each, matrix of **costs** (to be **minimized**):

		Bob	
		b_1	b_2
		a_1	a_2
Alice	a_1	(4, 6), SC = 5	(6, 8), SC = 7
	a_2	(7, 5), SC = 6	(1, 7), SC = 4

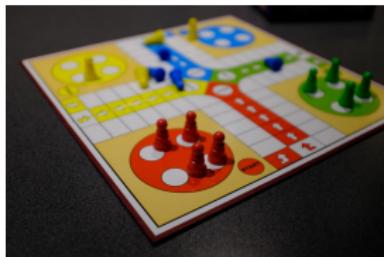


Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Example: 2 players, 2 actions each, matrix of **costs** (to be **minimized**):

		Bob	
		b_1	b_2
		a_1	a_2
Alice	a_1	(4, 6), SC = 5	(6, 8), SC = 7
	a_2	(7, 5), SC = 6	(2, 2), SC = 2



Multiple agents:

- Competition: **Nash equilibrium**, individual cost → “game”
- Cooperation: **Social optimum**, social cost → “control”

Example: 2 players, 2 actions each, matrix of **costs** (to be **minimized**):

		Bob	
		b_1	b_2
		a_1	a_2
Alice	a_1	(4, 6), SC = 5	(6, 8), SC = 7
	a_2	(7, 5), SC = 6	(2, 2), SC = 2

Outline

1. Introduction

2. From N to infinity

- Example 1: Population Distribution
- Example 2: Flocking
- Example 3: Price Impact

3. Warm-up: LQMFG

Intuition

Outline

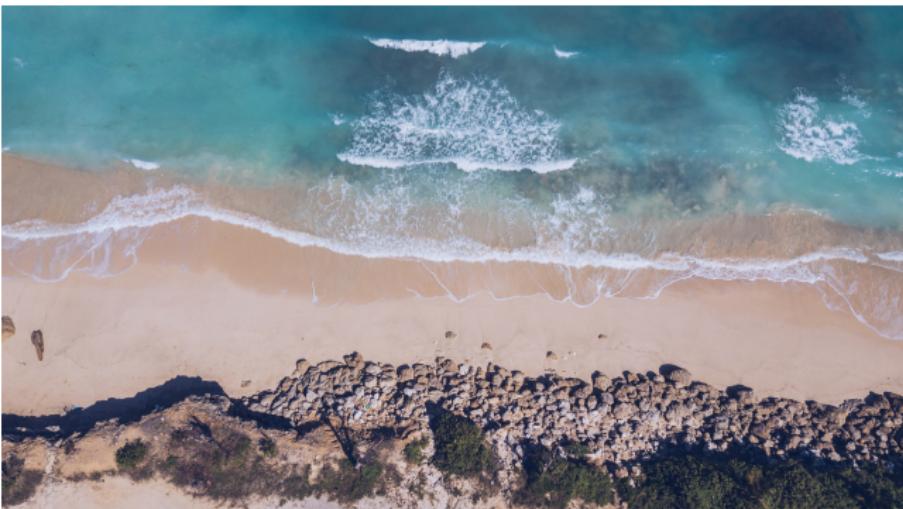
1. Introduction

2. From N to infinity

- Example 1: Population Distribution
- Example 2: Flocking
- Example 3: Price Impact

3. Warm-up: LQMFG

A Static Example: Towel on the Beach



A Static Example: Towel on the Beach



A Static Example: Towel on the Beach

- N players (people)
- State = position of the towel. Space:
 $\mathcal{S} = \{-M, -M + 1, \dots, -1, 0, 1, \dots, M - 1, M\}$

A Static Example: Towel on the Beach

- N players (people)
- State = position of the towel. Space:
 $\mathcal{S} = \{-M, -M + 1, \dots, -1, 0, 1, \dots, M - 1, M\}$
- Simultaneously choose their location: $x^i, i = 1, \dots, N$

A Static Example: Towel on the Beach

- N players (people)
- State = position of the towel. Space:
 $\mathcal{S} = \{-M, -M + 1, \dots, -1, 0, 1, \dots, M - 1, M\}$
- Simultaneously choose their location: $x^i, i = 1, \dots, N$
- Population distribution:

$$\mu(x) = |\{j : x^j = x\}|/N, \quad x \in \mathcal{S}$$

A Static Example: Towel on the Beach

- N players (people)
- State = position of the towel. Space:
 $\mathcal{S} = \{-M, -M + 1, \dots, -1, 0, 1, \dots, M - 1, M\}$
- Simultaneously choose their location: $x^i, i = 1, \dots, N$
- Population distribution:

$$\mu(x) = |\{j : x^j = x\}|/N, \quad x \in \mathcal{S}$$

- Each player pays a cost:
 - ▶ distance to a point of interest
 - ▶ density of people at their location
 - ▶ mean position of the population
 - ▶ ...

A Static Example: Towel on the Beach

- Infinitely many players (people)
- Simultaneously choose their location
- Population distribution μ on \mathcal{S}

A Static Example: Towel on the Beach

A Static Example: Towel on the Beach

What if people cooperate instead of competing?

Outline

1. Introduction

2. From N to infinity

- Example 1: Population Distribution
- **Example 2: Flocking**
- Example 3: Price Impact

3. Warm-up: LQMFG

A Dynamic Example: Flocking



A Dynamic Example: Flocking

Flocking model [Cucker, Smale; ...]:

- N players (birds)
- State: (position, velocity). Space: $\mathcal{S} = \mathbb{R}^3 \times \mathbb{R}^3$

A Dynamic Example: Flocking

Flocking model [Cucker, Smale; ...]:

- N players (birds)
- State: (position, velocity). Space: $\mathcal{S} = \mathbb{R}^3 \times \mathbb{R}^3$
- Player i chooses their acceleration: $\textcolor{red}{a^i} \in \mathbb{R}^3$, $i = 1, \dots, N$
- Dynamics:

$$\begin{cases} x_{n+1}^i = x_n^i + v_n^i \Delta t, \\ v_{n+1}^i = v_n^i + \textcolor{red}{a_n^i} \Delta t + \epsilon_{n+1}^i \end{cases}$$

A Dynamic Example: Flocking

Flocking model [Cucker, Smale; ...]:

- N players (birds)
- State: (position, velocity). Space: $\mathcal{S} = \mathbb{R}^3 \times \mathbb{R}^3$
- Player i chooses their acceleration: $\textcolor{red}{a^i} \in \mathbb{R}^3$, $i = 1, \dots, N$
- Dynamics:

$$\begin{cases} x_{n+1}^i = x_n^i + v_n^i \Delta t, \\ v_{n+1}^i = v_n^i + \textcolor{red}{a_n^i} \Delta t + \epsilon_{n+1}^i \end{cases}$$

- Each player pays a cost of velocity misalignment:

$$f_\beta^{\text{flock}, i}(\underline{x}, \underline{v}) = \left\| \frac{1}{N} \sum_{j=1}^N \frac{(v^i - v^j)}{(1 + \|x^i - x^j\|^2)^\beta} \right\|^2,$$

where $\beta \geq 0$ is a parameter

A Dynamic Example: Flocking

Flocking model [Cucker, Smale; ...]:

- N players (birds)
- State: (position, velocity). Space: $\mathcal{S} = \mathbb{R}^3 \times \mathbb{R}^3$
- Player i chooses their acceleration: $\textcolor{red}{a^i} \in \mathbb{R}^3$, $i = 1, \dots, N$
- Dynamics:

$$\begin{cases} x_{n+1}^i = x_n^i + v_n^i \Delta t, \\ v_{n+1}^i = v_n^i + \textcolor{red}{a_n^i} \Delta t + \epsilon_{n+1}^i \end{cases}$$

- Each player pays a cost of velocity misalignment:

$$f_\beta^{\text{flock}, i}(\underline{x}, \underline{v}) = \left\| \frac{1}{N} \sum_{j=1}^N \frac{(v^i - v^j)}{(1 + \|x^i - x^j\|^2)^\beta} \right\|^2,$$

where $\beta \geq 0$ is a parameter

- Population distribution μ_n^N on \mathcal{S}

$$\mu_n^N = \frac{1}{N} \sum_{j=1}^N \delta_{x_n^j}$$

- New writing for $f_\beta^{\text{flock}, i}$

A Dynamic Example: Flocking

Mean Field Game version [Nourian, Caines, Malhamé; ...]:

- Infinitely many players (birds)
- Population distribution μ on \mathcal{S} :

$$\mu_n^N \xrightarrow[N \rightarrow \infty]{} \mu_n$$

A Dynamic Example: Flocking

Mean Field Game version [Nourian, Caines, Malhamé; ...]:

- Infinitely many players (birds)
- Population distribution μ on \mathcal{S} :

$$\mu_n^N \xrightarrow[N \rightarrow \infty]{} \mu_n$$

- Each player pays a cost of velocity misalignment:

$$f_\beta^{\text{flock}}(x, v, \mu) = \left\| \int_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{(v - v')}{(1 + \|x - x'\|^2)^\beta} d\mu(x', v') \right\|^2,$$

where $\beta \geq 0$

A Dynamic Example: Flocking

Outline

1. Introduction

2. From N to infinity

- Example 1: Population Distribution
- Example 2: Flocking
- Example 3: Price Impact

3. Warm-up: LQMFG

Another Dynamic Example: Price Impact



Another Dynamic Example: Price Impact

- N players (traders)
- State of player i : $(S^i, X^i, K^i) \in \mathbb{R}^3$
 - ▶ Price process:

$$dS_t = \sigma_0 dW_t^0$$

- ▶ Inventory: action = trading speed v_t^i

$$dX_t^i = v_t^i dt + \sigma dW_t^i$$

- ▶ Wealth:

$$dK_t^i = -\left(v_t^i S_t + |v_t^i|^2\right) dt$$

Another Dynamic Example: Price Impact

- N players (traders)
- State of player i : $(S^i, X^i, K^i) \in \mathbb{R}^3$
 - ▶ Price process:

$$dS_t = \sigma_0 dW_t^0$$

- ▶ Inventory: action = trading speed v_t^i

$$dX_t^i = v_t^i dt + \sigma dW_t^i$$

- ▶ Wealth:

$$dK_t^i = -\left(v_t^i S_t + |v_t^i|^2\right) dt$$

- Payoff of player i :

$$J(\mathbf{v}) = \mathbb{E} \left[V_T^i - \int_0^T |X_t^i|^2 dt - |X_T^i|^2 \right]$$

where $V_t^i = K_t^i + X_t^i S_t$ = portfolio value

Another Dynamic Example: Price Impact

- N players (traders)
- State of player i : $(S^i, X^i, K^i) \in \mathbb{R}^3$
 - ▶ Price process: with $\nu = \text{population's distribution over actions} = \frac{1}{N} \sum_{j=1}^N \delta_{v^j}$,

$$dS_t = \sigma_0 dW_t^0 + \gamma \int_{\mathbb{R}} ad\nu_t(a) dt$$

- ▶ Inventory: action = trading speed v_t^i

$$dX_t^i = v_t^i dt + \sigma dW_t^i$$

- ▶ Wealth:

$$dK_t^i = -\left(v_t^i S_t + |v_t^i|^2\right) dt$$

- Payoff of player i :

$$J(v) = \mathbb{E} \left[V_T^i - \int_0^T |X_t^i|^2 dt - |X_T^i|^2 \right]$$

where $V_t^i = K_t^i + X_t^i S_t$ = portfolio value

Another Dynamic Example: Price Impact

Mean Field Game version [Carmona, Lacker; Carmona, Delarue; ...]:

- Infinitely many players (traders)
- State of a typical player: $(S, X, K) \in \mathbb{R}^3$
 - ▶ Price process: with $\nu =$ population's distribution over actions,

$$dS_t = \sigma_0 dW_t^0 + \gamma \int_{\mathbb{R}} ad\nu_t(a) dt$$

- ▶ Inventory: Typical agent's inventory:

$$dX_t^{\textcolor{red}{v}} = \textcolor{red}{v}_t dt + \sigma dW_t$$

- ▶ Wealth: Typical agent's wealth:

$$dK_t^{\textcolor{red}{v}} = -(\textcolor{red}{v}_t S_t + |\textcolor{red}{v}_t|^2) dt$$

Another Dynamic Example: Price Impact

Mean Field Game version [Carmona, Lacker; Carmona, Delarue; ...]:

- Infinitely many players (traders)
- State of a typical player: $(S, X, K) \in \mathbb{R}^3$
 - ▶ Price process: with $\nu =$ population's distribution over actions,

$$dS_t = \sigma_0 dW_t^0 + \gamma \int_{\mathbb{R}} ad\nu_t(a) dt$$

- ▶ Inventory: Typical agent's inventory:

$$dX_t^v = v_t dt + \sigma dW_t$$

- ▶ Wealth: Typical agent's wealth:

$$dK_t^v = -\left(v_t S_t + |v_t|^2\right) dt$$

- Payoff of a typical player:

$$J(v, \nu) = \mathbb{E} \left[V_T^v - \int_0^T |X_t^v|^2 dt - |X_T^v|^2 \right]$$

where $V_t^v = K_t^v + X_t^v S_t =$ portfolio value

Another Dynamic Example: Price Impact

- Simpler rewriting:

- ▶ By the self-financing condition,

$$dV_t^{\textcolor{red}{v}} = \left[-|\textcolor{red}{v}_t|^2 + \gamma X_t^{\textcolor{red}{v}} \int_{\mathbb{R}} ad\nu_t(a) \right] dt + \sigma S_t dW_t + \sigma_0 X_t^{\textcolor{red}{v}} dW_t^0$$

- ▶ Hence: maximize

$$J(\textcolor{red}{v}, \nu) = \mathbb{E} \left[\int_0^T \left(\gamma X_t^{\textcolor{red}{v}} \int_{\mathbb{R}} ad\nu_t(a) - |\textcolor{red}{v}_t|^2 - |X_t^{\textcolor{red}{v}}|^2 \right) dt + |X_T^{\textcolor{red}{v}}|^2 \right]$$

subject to inventory dynamics:

$$dX_t^{\textcolor{red}{v}} = \textcolor{red}{v}_t dt + \sigma dW_t$$

- Linear-Quadratic (LQ) structure

Another Dynamic Example: Price Impact

More Examples



Outline

1. Introduction

2. From N to infinity

3. Warm-up: LQMFG

- Definition of the Problem
- Algorithms
- MFC & Price of Anarchy

Outline

1. Introduction

2. From N to infinity

3. Warm-up: LQMFG

- Definition of the Problem
- Algorithms
- MFC & Price of Anarchy

Linear-Quadratic N-Player Game

- N players
- State space: $\mathcal{S} = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, \bar{v}_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and

$$b(x, \bar{m}, \bar{v}) = Ax + \bar{A}\bar{m} + B\bar{v}$$

where X_0^i and W^i are i.i.d.

Linear-Quadratic N-Player Game

- N players
- State space: $\mathcal{S} = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, v_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and

$$b(x, m, v) = Ax + \bar{A}m + Bv$$

where X_0^i and W^i are i.i.d.

- Cost for player i :

$$J^i(v^1, \dots, v^N) = \mathbb{E} \left[\int_0^T f(X_t^i, \bar{\mu}_t^N, v_t^i)dt + g(X_T^i, \bar{\mu}_T^N) \right]$$

with

$$f(x, m, v) = \frac{1}{2} [Qx^2 + \bar{Q}(x - Sm)^2 + Cv^2]$$

$$g(x, m) = \frac{1}{2} [Q_Tx^2 + \bar{Q}_T(x - S_Tm)^2]$$

Linear-Quadratic N-Player Game

- N players
- State space: $\mathcal{S} = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, v_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and

$$b(x, m, v) = Ax + \bar{A}m + Bv$$

where X_0^i and W^i are i.i.d.

- Cost for player i :

$$J^i(v^1, \dots, v^N) = \mathbb{E} \left[\int_0^T f(X_t^i, \bar{\mu}_t^N, v_t^i)dt + g(X_T^i, \bar{\mu}_T^N) \right]$$

with

$$f(x, m, v) = \frac{1}{2} [Qx^2 + \bar{Q}(x - Sm)^2 + Cv^2]$$

$$g(x, m) = \frac{1}{2} [Q_Tx^2 + \bar{Q}_T(x - S_Tm)^2]$$

- **Nash equilibrium:** $\hat{v} = (\hat{v}^1, \dots, \hat{v}^N)$ s.t. for all i , for all v^i

$$J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \leq J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N)$$

- Reminder: N player **Nash equilibrium**: $\underline{\hat{v}} = (\hat{v}^1, \dots, \hat{v}^N)$ s.t. for all i , for all v^i

$$J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \leq J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N)$$

- Reminder: N player **Nash equilibrium**: $\underline{\hat{v}} = (\hat{v}^1, \dots, \hat{v}^N)$ s.t. for all i , for all v^i

$$J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \leq J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N)$$

- By symmetry & homogeneity, we can write $J^i(v^1, \dots, v^N) = J^{MFNE}(v^i, \bar{\mu}^N)$

- Reformulation: $\underline{\hat{v}} = \hat{v}^1, \dots, \hat{v}^N$ s.t. for all i , for all v^i

$$J^{MFNE}(\hat{v}^i, \bar{\mu}^N) \leq J^{MFNE}(v^i, \tilde{\mu}^N)$$

where

$$\begin{cases} \bar{\mu}^N = \text{mean process with } (\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \\ \tilde{\mu}^N = \text{mean process with } (\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \end{cases}$$

Linear-Quadratic Mean Field Game

- Reminder: N player **Nash equilibrium**: $\hat{v} = (\hat{v}^1, \dots, \hat{v}^N)$ s.t. for all i , for all v^i
$$J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \leq J^i(\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N)$$
- By symmetry & homogeneity, we can write $J^i(v^1, \dots, v^N) = J^{MFNE}(v^i, \bar{\mu}^N)$
- Reformulation: $\hat{v} = \hat{v}^1, \dots, \hat{v}^N$ s.t. for all i , for all v^i

$$J^{MFNE}(\hat{v}^i, \bar{\mu}^N) \leq J^{MFNE}(v^i, \tilde{\mu}^N)$$

where

$$\begin{cases} \bar{\mu}^N = \text{mean process with } (\hat{v}^1, \dots, \hat{v}^{i-1}, \hat{v}^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \\ \tilde{\mu}^N = \text{mean process with } (\hat{v}^1, \dots, \hat{v}^{i-1}, v^i, \hat{v}^{i+1}, \dots, \hat{v}^N) \end{cases}$$

- Mean Field Nash equilibrium:** $(\hat{v}, \bar{\mu})$ s.t. for all v

$$J^{MFNE}(\hat{v}, \bar{\mu}) \leq J^{MFNE}(v, \bar{\mu})$$

where

$$\bar{\mu} = \text{mean process if everybody uses } \hat{v}$$

What does it mean to “solve” this MFG?

- population behavior $\bar{\mu} = (\bar{\mu}_t)_{t \in [0, T]}$
- individual behavior $\hat{v} = (\hat{v}_t)_{t \in [0, T]}$
- individual value function u

Value function:

$$u(t, x) = \text{optimal cost-to-go}$$

for a player starting at x at time t while the population flow is at equilibrium

Explicit Solution

Taking $d = 1$ to alleviate notation, it can be shown:

$$\begin{cases} \bar{\mu}_t = z_t, \\ \hat{v}(t, x) = -B(p_t x + r_t)/C, \\ u(t, x) = \frac{1}{2}p_t x^2 + r_t x + s_t, \end{cases}$$

Explicit Solution

Taking $d = 1$ to alleviate notation, it can be shown:

$$\begin{cases} \bar{\mu}_t = z_t, \\ \hat{v}(t, x) = -B(p_t x + r_t)/C, \\ u(t, x) = \frac{1}{2}p_t x^2 + r_t x + s_t, \end{cases}$$

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

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

Explicit Solution

Taking $d = 1$ to alleviate notation, it can be shown:

$$\begin{cases} \bar{\mu}_t = z_t, \\ \hat{v}(t, x) = -B(p_t x + r_t)/C, \\ u(t, x) = \frac{1}{2}p_t x^2 + r_t x + s_t, \end{cases}$$

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

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

Key points:

- coupling between z and r
- forward-backward structure

Outline

1. Introduction

2. From N to infinity

3. Warm-up: LQMFG

- Definition of the Problem
- **Algorithms**
- MFC & Price of Anarchy

Algorithm 1: Banach-Picard Iterations

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

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

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

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

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

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

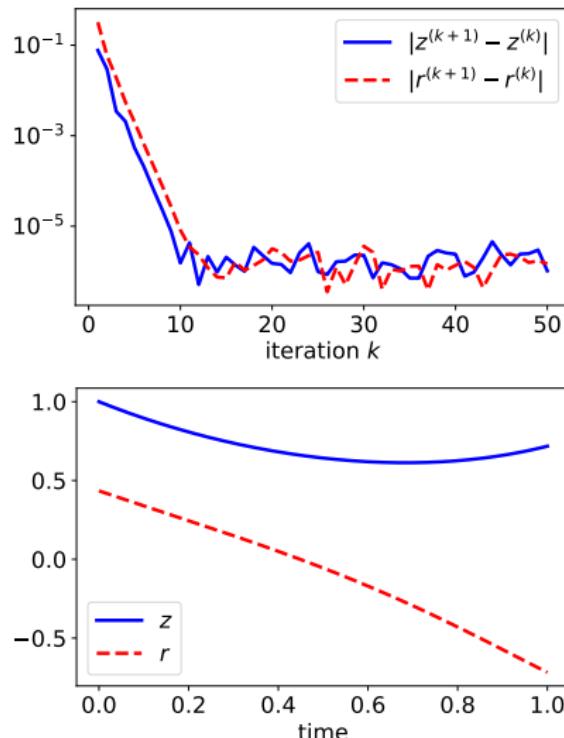
4 Let $z^{(k+1)}$ be the solution to:

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

5 **return** $(z^{(K)}, r^{(K)})$

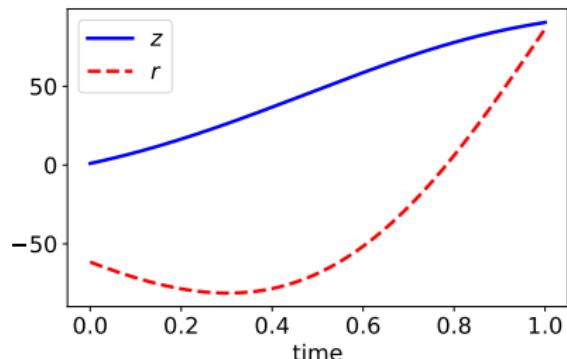
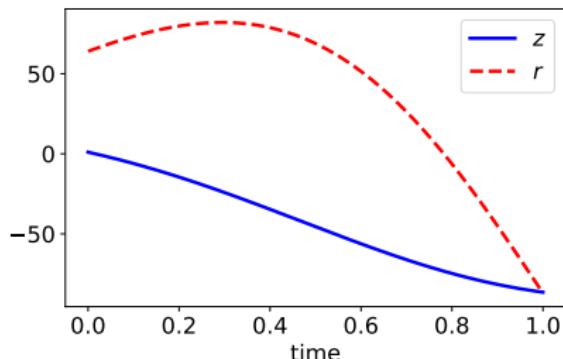
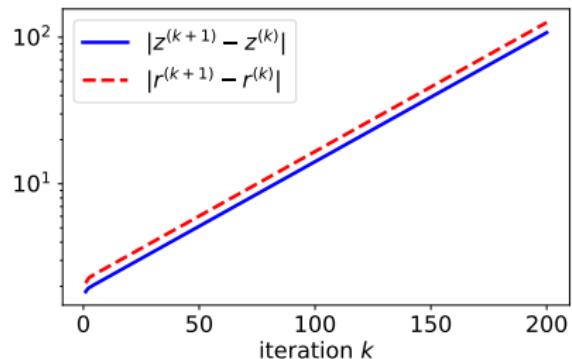
Algorithm 1: Banach-Picard Iterations – Illustration 1

Test case 1 (see [L., AMS notes'21])



Algorithm 1: Banach-Picard Iterations – Illustration 2

Test case 2 (see [L., AMS notes'21])



Note: Banach-Picard Iterations with Damping

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

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

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

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

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

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

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

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

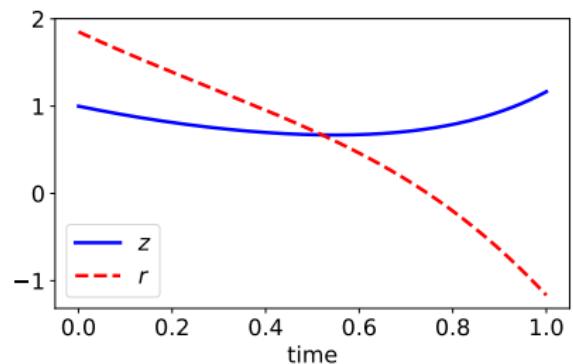
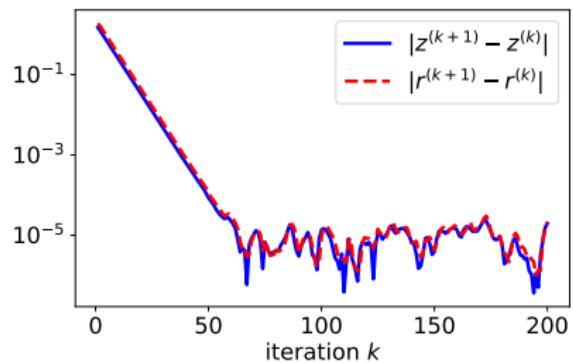
5 Let $\tilde{z}^{(k+1)} = \delta \tilde{z}^{(k)} + (1 - \delta) z^{(k+1)}$

6 **return** $(z^{(K)}, r^{(K)})$

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

Test case 2

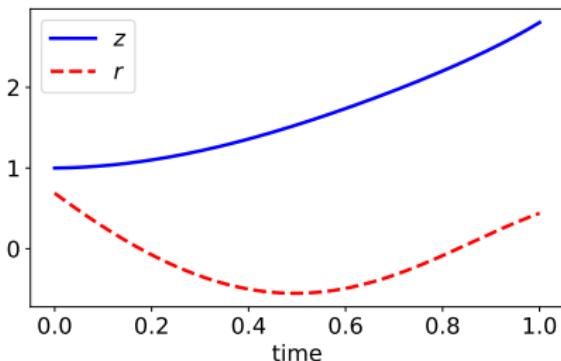
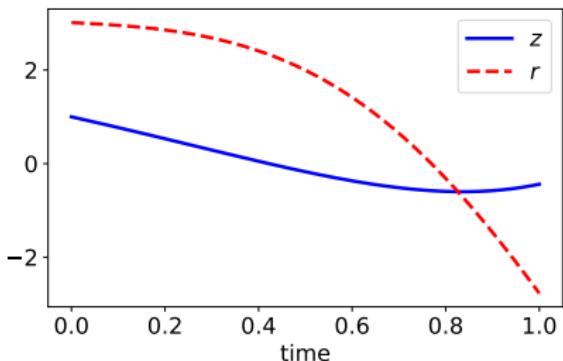
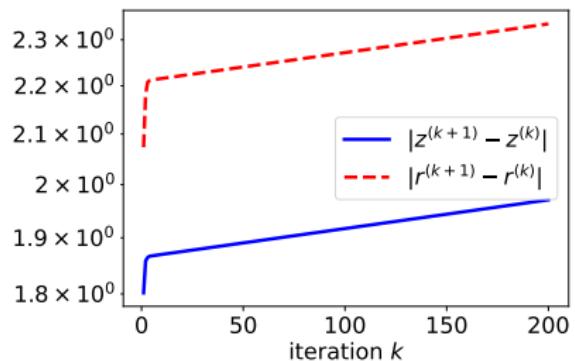
Damping = 0.1



Algorithm 1': Banach-Picard Iterations with Damping – Illustration 2

Test case 2

Damping = 0.01



Algorithm 2: Fictitious Play

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

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

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

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

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

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

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

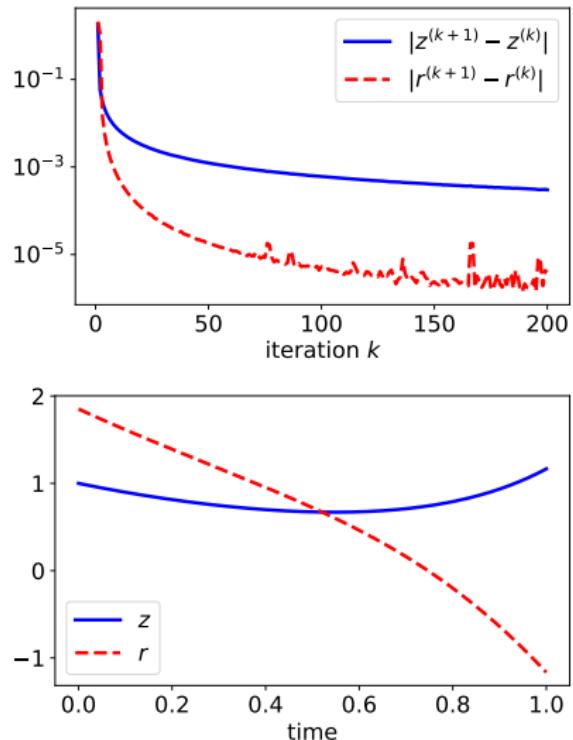
$$\frac{dz}{dt} = (A + \bar{A} - B^2 C^{-1})z_t - B^2 C^{-1}r_t^{(k+1)}, \quad z_0 = \bar{x}_0$$

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

6 **return** $(z^{(K)}, r^{(K)})$

Algorithm 2: Fictitious Play – Illustration

Test case 2



Algorithms 1, 1' & 2: Common Framework

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

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

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

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

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

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

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

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

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

6 **return** $(z^{(K)}, r^{(K)})$

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

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

by the forward-forward system

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

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

Algorithm 4: Newton Method – Intuition

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

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

- Uniform grid on $[0, T]$, step Δt

- Discrete ODE system:

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

Algorithm 4: Newton Method – Implementation

- Recast the problem:

(Z, R) 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

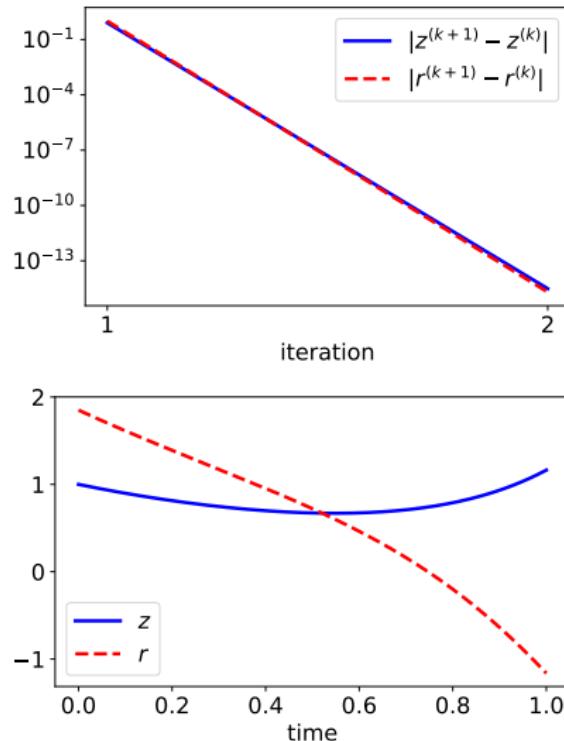
Input: Initial guess (\tilde{Z}, \tilde{R}) ; number of iterations K

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

- 1 Initialize $(Z^{(0)}, R^{(0)}) = (\tilde{Z}, \tilde{R})$
 - 2 **for** $k = 0, 1, 2, \dots, K - 1$ **do**
 - 3 Let $(\tilde{Z}^{(k+1)}, \tilde{R}^{(k+1)})$ solve
$$D\mathcal{F}(Z^{(k)}, R^{(k)})(\tilde{Z}^{(k+1)}, \tilde{R}^{(k+1)}) = \mathcal{F}(Z^{(k)}, R^{(k)})$$
 - 4 Let $(Z^{(k+1)}, R^{(k+1)}) = (\tilde{Z}^{(k+1)}, \tilde{R}^{(k+1)}) + (Z^{(k)}, R^{(k)})$
 - 5 **return** $(Z^{(K)}, R^{(K)})$
-

Algorithm 4: Newton Method – Illustration

Test case 2



Algorithm 4: Newton Method – Explanation

Reminder: Discrete ODE system:

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

Algorithm 4: Newton Method – Explanation

Reminder: Discrete ODE system:

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

Can be rewritten as a linear system:

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

Outline

1. Introduction

2. From N to infinity

3. Warm-up: LQMFG

- Definition of the Problem
- Algorithms
- MFC & Price of Anarchy

Linear-Quadratic N-Agent Control

- N agents
- State space: $S = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, v_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and same $b(\cdot, \cdot, \cdot)$

where X_0^i and W^i are i.i.d.

Linear-Quadratic N-Agent Control

- N agents
- State space: $\mathcal{S} = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, v_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and same $b(\cdot, \cdot, \cdot)$

where X_0^i and W^i are i.i.d.

- Cost for player i :

$$J^i(v^1, \dots, v^N) = \mathbb{E} \left[\int_0^T f(X_t^i, \bar{\mu}_t^N, v_t^i)dt + g(X_T^i, \bar{\mu}_T^N) \right]$$

with same $f(\cdot, \cdot, \cdot)$ and $g(\cdot, \cdot)$

Linear-Quadratic N-Agent Control

- N agents
- State space: $S = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \bar{\mu}_t^N, v_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\bar{\mu}_t^N$ = mean position at time t and same $b(\cdot, \cdot, \cdot)$
where X_0^i and W^i are i.i.d.

- Cost for player i :

$$J^i(v^1, \dots, v^N) = \mathbb{E} \left[\int_0^T f(X_t^i, \bar{\mu}_t^N, v_t^i)dt + g(X_T^i, \bar{\mu}_T^N) \right]$$

with same $f(\cdot, \cdot, \cdot)$ and $g(\cdot, \cdot)$

- Social cost for the population:

$$J^{Soc}(\underline{v}) = \frac{1}{N} \sum_{i=1}^N J^i(\underline{v})$$

Linear-Quadratic N-Agent Control

- N agents
- State space: $S = \mathbb{R}^d$; action space: $\mathcal{A} = \mathbb{R}^k$
- Dynamics for player i : initial position $X_0^i \sim \mathcal{N}(\bar{x}_0, \sigma_0^2)$,

$$dX_t^i = b(X_t^i, \underline{\mu}_t^N, \underline{v}_t^i)dt + \sigma dW_t^i, \quad t \geq 0,$$

with $\underline{\mu}_t^N$ = mean position at time t and same $b(\cdot, \cdot, \cdot)$
where X_0^i and W^i are i.i.d.

- Cost for player i :

$$J^i(\underline{v}^1, \dots, \underline{v}^N) = \mathbb{E} \left[\int_0^T f(X_t^i, \underline{\mu}_t^N, \underline{v}_t^i)dt + g(X_T^i, \underline{\mu}_T^N) \right]$$

with same $f(\cdot, \cdot, \cdot)$ and $g(\cdot, \cdot)$

- Social cost for the population:

$$J^{Soc}(\underline{v}) = \frac{1}{N} \sum_{i=1}^N J^i(\underline{v})$$

- **Social optimum:** $\underline{v}^* = (\underline{v}^{*,1}, \dots, \underline{v}^{*,N})$ s.t. for all i , all $\underline{v} = (\underline{v}^1, \dots, \underline{v}^N)$

$$J^{Soc}(\underline{v}^*) \leq J^{Soc}(\underline{v})$$

- Infinitely many agents
- Mean field social cost:

$$J^{MFSoc}(\mathbf{v}) = \mathbb{E} \left[\int_0^T f(X_t, \bar{\mu}_t, v_t) dt + g(X_T, \bar{\mu}_T) \right]$$

where

$$dX_t = b(X_t, \bar{\mu}_t, v_t) dt + \sigma dW_t, \quad t \geq 0,$$

and

$\bar{\mu} = \bar{\mu}^{\mathbf{v}}$ = mean process if everybody uses \mathbf{v}

- **Mean field social optimum:** v^* , s.t. for all \mathbf{v}

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(\mathbf{v})$$

- Infinitely many agents
- Mean field social cost:

$$J^{MFSoc}(\mathbf{v}) = \mathbb{E} \left[\int_0^T f(X_t, \bar{\mu}_t, v_t) dt + g(X_T, \bar{\mu}_T) \right]$$

where

$$dX_t = b(X_t, \bar{\mu}_t, v_t) dt + \sigma dW_t, \quad t \geq 0,$$

and

$\bar{\mu} = \bar{\mu}^v$ = mean process if everybody uses v

- **Mean field social optimum:** v^* , s.t. for all v

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(v)$$

- Key point: v changes $\Rightarrow \bar{\mu}^v$ changes

Price of Anarchy

- MFG solution: mean field Nash equilibrium: $(\hat{v}, \bar{\mu})$ s.t. for all v

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \leq J^{MFNE}(v, \bar{\mu}^{\hat{v}})$$

- MFC solution: mean field social optimum: v^* s.t. for all v

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(v)$$

Price of Anarchy

- MFG solution: mean field Nash equilibrium: $(\hat{v}, \bar{\mu})$ s.t. for all v

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \leq J^{MFNE}(v, \bar{\mu}^{\hat{v}})$$

- MFC solution: mean field social optimum: v^* s.t. for all v

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(v)$$

- For any v ,

$$J^{MFSoc}(v) = J^{MFNE}(v, \bar{\mu}^v)$$

Price of Anarchy

- MFG solution: mean field Nash equilibrium: $(\hat{v}, \bar{\mu})$ s.t. for all v

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \leq J^{MFNE}(v, \bar{\mu}^{\hat{v}})$$

- MFC solution: mean field social optimum: v^* s.t. for all v

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(v)$$

- For any v ,

$$J^{MFSoc}(v) = J^{MFNE}(v, \bar{\mu}^v)$$

- In general:

$$\hat{v} \neq v^*$$

$$\bar{\mu}^{\hat{v}} \neq \bar{\mu}^{v^*}$$

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \neq J^{MFSoc}(v^*)$$

Price of Anarchy

- MFG solution: mean field Nash equilibrium: $(\hat{v}, \bar{\mu})$ s.t. for all v

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \leq J^{MFNE}(v, \bar{\mu}^{\hat{v}})$$

- MFC solution: mean field social optimum: v^* s.t. for all v

$$J^{MFSoc}(v^*) \leq J^{MFSoc}(v)$$

- For any v ,

$$J^{MFSoc}(v) = J^{MFNE}(v, \bar{\mu}^v)$$

- In general:

$$\hat{v} \neq v^*$$

$$\bar{\mu}^{\hat{v}} \neq \bar{\mu}^{v^*}$$

$$J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}}) \neq J^{MFSoc}(v^*)$$

- **Price of Anarchy (PoA):**

$$PoA = \frac{J^{MFNE}(\hat{v}, \bar{\mu}^{\hat{v}})}{J^{MFSoc}(v^*)},$$

Explicit Solution

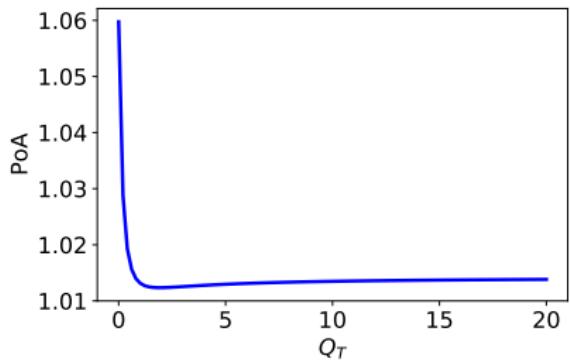
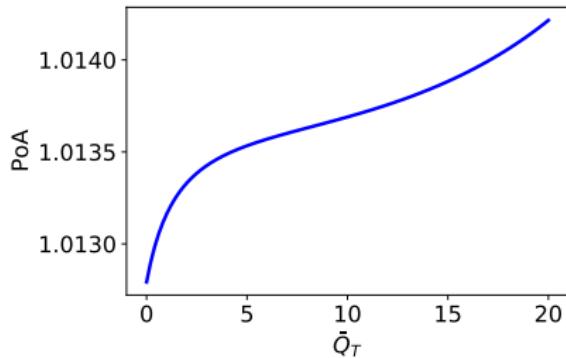
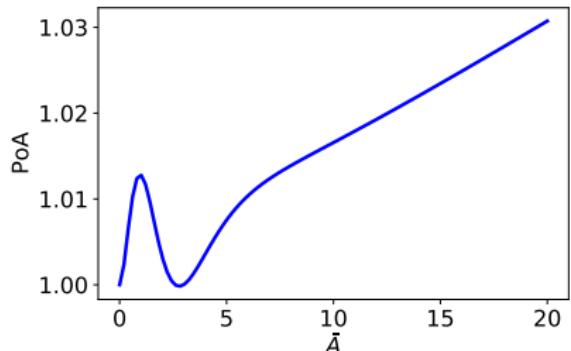
Mean field social optimum:

$$\begin{cases} \bar{\mu}^{\textcolor{red}{v}^*} = \check{z}_t, \\ \textcolor{red}{v}^*(t, x) = -B(\check{p}_t x + \check{r}_t)/C, \end{cases}$$

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

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

Price of Anarchy – Illustration



Preview of Next Lectures
