

Mean-field games for traffic: modeling, numerics, and Nash equilibria

Amal MACHTALAY

Mohammed VI Polytechnic University (UM6P), Morocco
`amal.machtalay@um6p.ma`

Based on part of my PhD work

Karlstad Applied Analysis Seminar (KAAS)

30 September 2026

Outline

- 1 Introduction and Motivation
- 2 From finite-player differential games to mean-field games
- 3 Solving the mean-field game PDE system using Newton-GMRES solver
- 4 From mean-field games back to finite player differential games
- 5 Conclusion

1 Introduction and Motivation

2 From finite-player differential games to mean-field games

3 Solving the mean-field game PDE system using Newton-GMRES solver

4 From mean-field games back to finite player differential games

5 Conclusion

Many agents, individual decisions, collective behavior



Flocking

collective motion



Crowd dynamics

pedestrians



Traffic flow

drivers



Energy systems

EV charging / smart grids



Financial markets

many traders



Autonomous systems

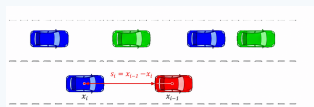
drone swarms

How can we describe the behavior of thousands of interacting decision-makers?

Microscopic viewpoint

► Microscopic level → individual behaviors

- Track individual agents and their interactions
- Agents/players = individual drivers



– Agent-based model (ABM)

Rule-based individual behavior

$$\dot{x}_i = v_i, \quad \dot{v}_i = A(s_i, v_i, v_{i-1}), \quad s_i = x_{i-1} - x_i.$$

e.g., in traffic: car-following, lane-changing, cellular automata

– Finite-player differential game (FPDG)

[V. Neumann and Morgenstern 1944; Nash 1950, 1951]

$$\dot{x}_i = v_i, \quad \dot{v}_i = a_i, \quad a_i^* = \arg \min_{a_i} J_i(a_i, a_{-i}^*).$$

Strategic/optimizing individual behavior; Nash equilibrium

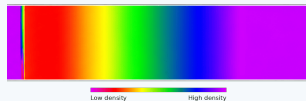
- Computationally expensive

Many agents interact; possible stochasticity

Macroscopic viewpoint

- ▶ **Macroscopic level** → collective dynamics

- Describe the evolution of the whole population
- Individual agents represented by **density** m



- Mean-field game/control (MFG/C)

[Huang et al. 2006; Lasry and Lions 2007]

$$-\partial_t u - \nu \Delta u + H(x, \nabla u, m) = 0 \quad \text{HJB: optimal decision}$$

$$\partial_t m - \nu \Delta m - \nabla \cdot (m D_p H(x, \nabla u, m)) = 0 \quad \text{FP: population evolution}$$

- Many individual interactions are replaced by a **mean-field interaction**
- Coupled forward–backward equations

$$u(T, x) = g(x, m(T, x)), \quad m(0, x) = m_0(x).$$

- Often nonlinear, high-dimensional, and possibly stochastic
- Discretization $\rightarrow$ large nonlinear systems

Grid size, number of unknowns, dimension $\rightarrow$ curse of dimensionality

MFG theory over the last 20 years

- ▶ Micro-to-macro game-theoretical framework

Mean-field equilibrium lead to ϵ -Nash equilibrium for FPDG.

Huang et al. 2006; Lasry and Lions 2007; Carmona and Delarue 2012; Nourian and Caines 2013; Bensoussan et al. 2014; Carmona and Delarue 2018a; Lacker 2020; Flandoli et al. 2022...

- ▶ Numerical methods for solving mean-field systems

Camilli et al. 2015; Cacace et al. 2017; Chassagneux et al. 2019; Ruthotto et al. 2020; Achdou and Kobeissi 2021; Laurière 2021; Bonnans et al. 2023; Baudalet et al. 2024; Carlini et al. 2024...

- ▶ Applications: economics, finance, crowd dynamics, sociology, engineering, epidemiology, computer science, traffic flow...

Guéant et al. 2011; Couillet et al. 2012; Gomes et al. 2014; Kolokoltsov and Bensoussan 2015; Bauso et al. 2016; Chan and Sircar 2017; Koulali et al. 2018; Barreiro-Gomez and Tembine 2019; Tembine 2020; Mo et al. 2024...

- ▶ Micro-macro bridge + Numerically validate ϵ -Nash

single-class traffic [Huang et al. 2020] , multi-class traffic [Machtalay et al. 2025]

20 Years of Mean-Field Games

IMSI · February 22–26, 2027



Mean-field and finite-player games

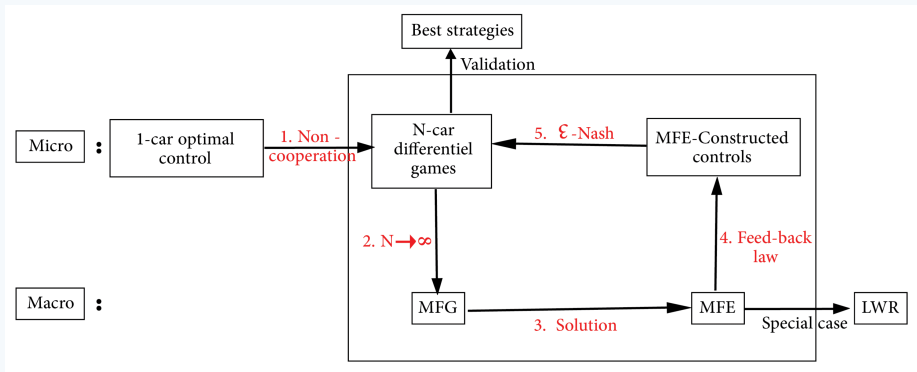


Figure: micro-macro game approach adapted from [Huang et al. 2020] .

MFE: Mean-Field Equilibrium

LWR: Lighthill-Whitham-Richards model [Lighthill and Whitham 1955; Richards 1956]

1 Introduction and Motivation

2 From finite-player differential games to mean-field games

3 Solving the mean-field game PDE system using Newton-GMRES solver

4 From mean-field games back to finite player differential games

5 Conclusion

Finite-player game

- Multi-class vehicle indexing:
 - J vehicle classes, indexed $j \in \{1, \dots, J\}$.
 - Class j contains N_j vehicles, total number $N = \sum_{j=1}^J N_j$.
 - The vehicles of class j are indexed $i_j \in \mathcal{I}_j = \{1 + \sum_{k=1}^{j-1} N_k, \dots, \sum_{k=1}^j N_k\}$.
- Every i_j -th vehicle:
 - Has velocity $v_{i_j}(t)$ and position $x_{i_j}(t)$.
 - Follows the dynamics: $dx_{i_j}(t) = v_{i_j}(t) dt$, $x_{i_j}(0) = x_{i_j,0}$.
Possible extension: stochastic setting, e.g., $dx_{i_j}(t) = v_{i_j}(t) dt + \sigma_{i_j} dW_{i_j}(t)$.
 - Minimizes the cost:
 - One-vehicle OC + interactions

$$\mathcal{L}_{i_j}(v_{i_j}, \textcolor{red}{v}_{-i_j}, \textcolor{red}{v}^{-j}) = \int_0^T f_{i_j} \left(v_{i_j}(t), x_{i_j}(t), \textcolor{red}{x}_{-i_j}(t), \textcolor{red}{x}^{-j}(t) \right) dt + G_{i_j}(x_{i_j}(T)).$$

f_{i_j} running cost, G_{i_j} terminal cost, $v^j = (v_{i_j}, v_{-i_j})$, $v_{-i_j} = (v_k)_{k \neq i_j}$.

Finite-player game

$$\mathcal{L}_{i_j}(v_{i_j}, v_{-i_j}, v^{-j}) = \int_0^T f_{i_j}(v_{i_j}(t), x_{i_j}(t), v_{-i_j}(t), v^{-j}(t)) dt + G_{i_j}(x_{i_j}(T)).$$

- ▶ Admissible set of control functions

$$\mathcal{V}_{i_j} := \left\{ v_{i_j} : [0, T] \rightarrow \mathbb{R} \mid 0 \leq v_{i_j}(t) \leq u_{\max}^j, \forall t \in [0, T] \right\}.$$

- ▶ Minimizing the cost

$$v_{i_j}^* = \operatorname{argmin}_{v_{i_j} \in \mathcal{V}_{i_j}} \left\{ \mathcal{L}_{i_j}(v_{i_j}, v_{-i_j}^*, v^{-j,*}) \right\},$$

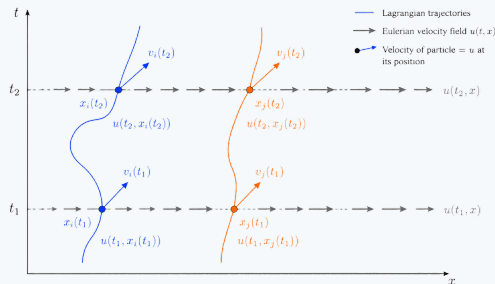
- ▶ N -player differential game:

$$\mathcal{L}_{i_j}(v_{i_j}^*, v_{-i_j}^*, v^{-j,*}) \leq \mathcal{L}_{i_j}(v_{i_j}, v_{-i_j}^*, v^{-j,*}), \quad v_{i_j} \in \mathcal{V}_{i_j}.$$

- ▶ Nash-equilibrium: $(v^{1,*}, \dots, v^{J,*})$

Mean-field limit

- ▶ Two main assumptions:
 - Infinite-population: $N \rightarrow \infty$.
 - Homogeneous driver behavior: $f_j := f_{i_j}$, $G_j := G_{i_j}$. $\Rightarrow$ Representative vehicle of class j .
- ▶ Micro-macro approach
 - Lagrangian description $\Leftrightarrow$ Eulerian description
 - Velocity field: $u^j(t, x_{i_j}(t)) = v_{i_j}(t)$.

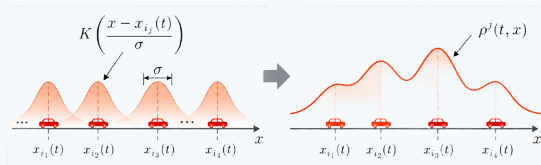


► Micro-macro approach

- Particle-based $\Leftrightarrow$ density-based
- Kernel density estimation (KDE) [Węglarczyk 2018]

- Density function: $\rho^j(t, x) \approx \frac{1}{\sigma N_j} \sum_{i_j \in \mathcal{I}_j} K\left(\frac{x - x_{i_j}(t)}{\sigma}\right)$,

$$\forall s \in \mathbb{R}, \int_{\mathbb{R}} K(s) ds = 1, \quad \sigma > 0.$$



Mean-field game

► Every i_j -th vehicle:

- Follows $dx_{i_j}(t) = v_{i_j}(t) dt$, $x_{i_j}(0) = x_{i_j,0}$.
- Minimizes

$$\mathcal{L}_{i_j}(v_{i_j}, v_{-i_j}, v^{-j}) = \int_0^T f_{i_j}(v_{i_j}(t), x_{i_j}(t), x_{-i_j}(t), x^{-j}(t)) dt + G_{i_j}(x_{i_j}(T)).$$

► Representative vehicle of class j

- Follows $dx = u^j(t, x(t)) dt$, $x(0) = x_{j,0}$.
- Minimizes

$$\mathcal{L}_j(u^j; \rho) = \int_0^T f_j(u^j(t, x(t)), \rho(t, x(t))) dt + G_j(x(T)),$$

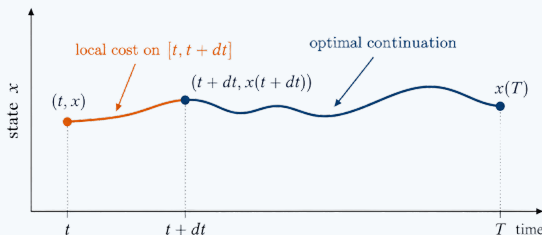
$x_{j,0}$ is a given initial position of the generic vehicle j , and $\rho = (\rho^1, \dots, \rho^J)$.

Hamilton-Jacobi-Bellman (HJB) equation

- Optimal cost (Bellman value function)

$$V^j(t, x) = \min_{0 \leq u^j \leq u_{max}^j} \int_t^T f_j \left(u^j(s, x(s)), \rho(s, x(s)) \right) ds + V^j(T, x(T)).$$

- Dynamic programming principle [Bellman 1957; Bensoussan and Frehse 1984]



- Over an infinitesimal time interval $[t, t+dt]$

$$V^j(t, x) = \min_{u^j} \int_t^{t+dt} f_j \left(u^j(s, x(s)), \rho(s, x(s)) \right) ds + V^j(t+dt, x(t+dt)).$$

Hamilton-Jacobi-Bellman (HJB) equation

- ▶ Assumptions over $[t, t + dt]$, $dt \rightarrow 0$:
 - The state trajectory $x(\cdot)$ is Lipschitz,
 - The map $s \mapsto f_j(u^j(s, x(s)), \rho(s, x(s)))$ is continuous,

- ▶ First-order expansions

$$V^j(t + dt, x(t + dt)) = V^j(t, x) + dt \frac{\partial V^j}{\partial t}(t, x) + dx \frac{\partial V^j}{\partial x}(t, x) + o(dt),$$
$$\int_t^{t+dt} f_j(u^j(s, x(s)), \rho(s, x(s))) ds = f_j(u^j(t, x(t)), \rho(t, x(t))) dt + o(dt).$$

- ▶ Plugging

$$\cancel{V^j(t, x)} = \min_{u^j} \left\{ f_j(u^j(t, x(t)), \rho(t, x(t))) \cancel{dt} + \cancel{V^j(t, x)} + \cancel{dt} \frac{\partial V^j}{\partial t}(t, x) + \underbrace{\frac{dx}{dt}}_{u^j} \frac{\partial V^j}{\partial x}(t, x) + \frac{o(dt)}{dt} \right\}.$$

Subtracting $\cancel{V^j}$ from both sides, dividing by dt , using the dynamics, $dt \rightarrow 0 \Rightarrow o(dt)/dt \rightarrow 0$.

Hamilton-Jacobi-Bellman (HJB) equation

- ▶ HJB equation

$$0 = \frac{\partial V^j}{\partial t}(t, x) + \min_{u^j} \left\{ f_j(u^j(t, x(t)), \rho(t, x(t))) + u^j(t, x(t)) \frac{\partial V^j}{\partial x}(t, x) \right\}.$$

- ▶ Hamiltonian function: $H_j(p_j, \rho) = \min_{0 \leq \alpha \leq u_{max}^j} \{f_j(\alpha, \rho) + \alpha p_j\}.$

$p_j = \frac{\partial V^j}{\partial x}$ the derivative of V^j w.r.t x .

- ▶ HJB equation [Hamilton 1834; Jacobi 1884; Bellman 1954]

$$\frac{\partial V^j}{\partial t}(t, x) + H_j \left(\frac{\partial V^j}{\partial x}(t, x), \rho(t, x) \right) = 0, \quad V^j(T, x) = G_j(x).$$

Terminal condition $V^j(T, x(T)) = G_j(x(T)).$

Fokker-Planck-Kolmogorov (FPK) equation

► FPK [Fokker 1914; Planck 1917; Kolmogorov 1931]

- Describe how the traffic density evolves in time.
- Probabilistic derivation
 - Stochastic dynamics: FPK equation
probability density $m^j(t, x)$, Brownian motion, Itô's formula.
 - Deterministic dynamics: Continuity equation (CE)
physical density $\rho^j(t, x) = N_j m^j(t, x)$, chain rule.
- Conservation of vehicles
within each class, over $[t, t + dt]$, $dt \rightarrow 0$, one-directional continuous road section
 $[x, x + dx]$, $dx \rightarrow 0$.

$$\frac{\partial}{\partial t} \rho^j(t, x) + \frac{\partial}{\partial x} (u^j(t, x) \rho^j(t, x)) = 0, \quad \rho(0, x) = \rho_0(x).$$

Mean-field game system

► Multi-class MFG:

- Interaction type: competition within and between classes.
- Mean-field classification [\[Bensoussan et al. 2013, 2018\]](#) .

⇒ Nash mean-field game (NMFG).

► Hamiltonian ⇒ minimizer $u^{j,*}(p_j, \rho) = \operatorname{argmin}_{0 \leq \alpha \leq u_{max}^j} \{f_j(\alpha, \rho) + \alpha p_j\}$,

$f_j(\alpha, \rho)$ is C^1 and strictly convex in α .

► Optimality condition ⇒ feedback control $u^{j,*}(t, x) = \frac{\partial H_j}{\partial p_j}(p_j, \rho)$.

► NMFG system:

$$\begin{cases} \frac{\partial \rho^j}{\partial t}(t, x) + \frac{\partial}{\partial x}(\rho^j(t, x) u^j(t, x)) = 0, & \rho^j(0, x) = \rho_0^j(x), & (i) \\ \frac{\partial V^j}{\partial t}(t, x) + H_j\left(\frac{\partial V^j}{\partial x}(t, x), \rho(t, x)\right) = 0, & V^j(T, x) = G_j(x), & (ii) \\ u^j(t, x) = H_j^*\left(\frac{\partial V^j}{\partial x}(t, x), \rho(t, x)\right), & & (iii) \end{cases}$$

$p_j = \partial V^j / \partial x$, $H_j^* = \partial H_j / \partial p_j$, (i) forward nonlinear FPK or continuity equation, (ii) backward nonlinear HJB equation, (iii) feedback.

- 1 Introduction and Motivation
- 2 From finite-player differential games to mean-field games
- 3 Solving the mean-field game PDE system using Newton-GMRES solver
- 4 From mean-field games back to finite player differential games
- 5 Conclusion

Finite difference schemes

► Uniform grid: $\Delta t = T/N_t$, $\Delta x = L/N_x$, $x_{k+1} = x_k + \Delta x$, $t^{n+1} = t^n + \Delta t$.

► Numerical approximations:

$$\rho_k^{j,n} \approx \frac{1}{\Delta x} \int_{x_{k-1}}^{x_k} \rho^j(t^n, s) ds, \quad u_k^{j,n} \approx \frac{1}{\Delta x} \int_{x_{k-1}}^{x_k} u^j(t^n, s) ds, \quad V_k^{j,n} \approx V^j(t^n, x_k),$$

ρ^j, u^j cell averages $\longrightarrow$ preserve mass, V^j pointwise.

► Continuity equation (non-linear conservative hyperbolic)

$$\frac{\rho_k^{j,n+1} - \frac{1}{2}(\rho_{k-1}^{j,n} + \rho_{k+1}^{j,n})}{\Delta t} + \frac{\rho_{k+1}^{j,n} u_{k+1}^{j,n} - \rho_{k-1}^{j,n} u_{k-1}^{j,n}}{2\Delta x} = 0,$$

Lax-Friedrichs, $\Delta t = \frac{c \Delta x}{u_{max}}$, $c \in (0, 1] \longrightarrow$ robustness, monotonicity, $\rho^j \geq 0$.

► HJB equation (first-order non-linear hyperbolic)

$$\frac{V_k^{j,n+1} - V_k^{j,n}}{\Delta t} + H_j \left(\frac{V_{k+1}^{j,n+1} - V_k^{j,n+1}}{\Delta x}, \rho_k^n \right) = 0.$$

upwind backward $\longrightarrow$ monotonicity, convergence to the viscosity solution.

Finite difference schemes

► Viscous HJB equation

$$\frac{V_k^{j,n+1} - V_k^{j,n}}{\Delta t} + H_j \left(\frac{V_{k+1}^{j,n+1} - V_k^{j,n+1}}{\Delta x}, \rho_k^n \right) + \nu \frac{V_{k+1}^{j,n+1} - 2V_k^{j,n+1} + V_{k-1}^{j,n+1}}{\Delta x^2} = 0,$$

regularity, viscosity solution, stability $\Delta t = \min \left(\frac{c \Delta x}{u_{max}}, \frac{\beta \Delta x^2}{\nu} \right)$, $\beta \in (0, \frac{1}{2}]$, $\nu \in (0, 1]$.

► Feedback-law equation

$$u_k^{j,n} - H_j^* \left(\frac{V_{k+1}^{j,n+1} - V_k^{j,n+1}}{\Delta x}, \rho_k^n \right) = 0,$$

upwind difference, consistent with HJB.

► Initial and terminal conditions

$$\rho_k^{j,0} - \frac{1}{\Delta x} \int_{x_{k-1}^j}^{x_k^j} \zeta_0^j(x) dx = 0, \quad V_k^{j,N_t} - V_T(x_k^j) = 0.$$

► Boundary conditions

$$V_0^{j,n} = V_{N_x}^{j,n}, \quad \rho_0^{j,n} = \rho_{N_x}^{j,n}.$$

Newton iterations

- ▶ Vector unknown: $w \in \mathbb{R}^{3N_t(JN_x)+2(JN_x)}$,

contains: $\{\rho_k^{j,n}\}_{1 \leq k \leq N_x}^{0 \leq n \leq N_t}$, $\{u_k^{j,n}\}_{1 \leq k \leq N_x}^{0 \leq n \leq N_t-1}$, $\{V_k^{j,n}\}_{1 \leq k \leq N_x}^{0 \leq n \leq N_t}$, $j = 1, \dots, J$.

- ▶ Non-linear NMFG system: $\mathcal{F}(w) = 0$, $\mathcal{F} \in \mathbb{R}^{3N_t(JN_x)+2(JN_x)}$,

encodes discrete: CE, HJB, feedback-law, initial, terminal, and boundary conditions.

- ▶ Linear system (Newton iteration m)

$$\mathcal{J}\left(w^{(m-1)}\right) \cdot \left(w^{(m)} - w^{(m-1)}\right) = -\mathcal{F}\left(w^{(m-1)}\right),$$

large, initial guess $w^{(0)}$, Krylov-subspace (LGMRES)

- ▶ Jacobian matrix of $\mathcal{F}$: $\mathcal{J} \in \mathbb{R}^{(3N_t(JN_x)+2(JN_x)) \times (3N_t(JN_x)+2(JN_x))}$,

non-symmetric, extremely sparse, $\text{nnz}(\mathcal{J}) = [(2J + 13)N_t + 2](JN_x) < 1\%$.

- ▶ Jacobian-free: Finite-difference Jacobian-vector product

$$\mathcal{J}(w)v \approx \frac{\mathcal{F}(w + \varepsilon v) - \mathcal{F}(w)}{\varepsilon}.$$

Preconditioning and multi-grid initialization

- ▶ Jacobian: non-symmetric, non-normal, ill-conditioned ($\kappa_2(\mathcal{J}) \geq \frac{1}{\Delta x}$), mesh-dependent, $\kappa_2(\mathcal{J}) \rightarrow \infty$ as $\Delta x \rightarrow 0$.
- ▶ Preconditioning $\mathcal{P}_0^{-1}$: $\mathcal{P}_0 = \mathcal{P}(w^{(0)})$, with $\mathcal{J}(w) = \mathcal{P}(w) + \mathcal{R}_H(w)$, $\mathcal{P}_0^{-1} \mathcal{J}(w^{(k)}) = I + \mathcal{P}_0^{-1} (\mathcal{P}(w^{(k)}) - \mathcal{P}(w^{(0)})) + \mathcal{P}_0^{-1} \mathcal{R}_H(w^{(k)})$.
- ▶ Multi-grid initialization: $w_{\mathcal{G}-1} \xrightarrow{\text{interpolate}} w_{\mathcal{G}}^{(0)} \xrightarrow{\text{solve}} w_{\mathcal{G}}$.
- ▶ Root Mean Square Error: $\text{RMSE} = \left(\frac{1}{N} \sum_{i=1}^N |\omega_i - \tilde{\omega}_i|^2 \right)^{1/2}$.

		without preconditioning		with preconditioning		with preconditioning & multi-grid			
N_x	N_t	iter	Res	iter	Res	iter	Res	RMSE	time
30	120	1000	4.96	9	1.4e-07	2	7.2e-07	0.028	3.5
60	240	1000	5.42	30	2.3e-06	3	1.6e-08	0.022	2e01
120	480	1000	5.01	117	4.5e-06	5	7.5e-07	0.016	1e02
240	960	1000	5.33	659	6.0e-06	15	2.0e-06	0.011	2e03
480	1920	1000	12.80	1000	0.262	168	5.9e-06	0.008	7e04
960	3840	1000	12.01	1000	0.404	840	6.0e-06	0.005	2e06

$$f_j^{\text{GLWR}}(w^j, \rho) = \frac{1}{2} \left(U_j(\rho) - w^j \right)^2, \quad J = 1, T = 3, L = 1, \rho_{jam} = 1, u_{max} = 1$$

$\text{Res} = \|\mathcal{F}(w)\|_2 \leq 6.10^{-6}$, CPU time in seconds, number of iterations for convergence (iter).

Regularization-continuation

► $\mathcal{J}_{V,V}^\nu = \mathcal{J}_{V,V} + \nu \frac{\Delta t}{\Delta x^2} \mathcal{L}$, viscosity-free $w_\nu \rightarrow w$ as $\nu \rightarrow 0$

[GS]: $\nu = 0.0$.

N_x	N_t	iter	Res
30	120	1000	0.35
60	240	1000	0.71
120	480	1000	0.9
240	960	1000	1.1

[GS]: $0.01 \leq \nu \leq 0.04$.

N_x	N_t	ν	iter	Res	RMSE	time
30	240	0.04	5	2.1e-06	0.0100	2e01
60	720	0.03	5	2.2e-06	0.0068	1e02
120	1920	0.02	19	3.3e-06	0.0107	3e03
240	3840	0.01	959	5.5e-06	0.0170	5e05

[GNS]: $\nu = 0.0$.

N_x	N_t	iter	Res
30	120	1000	0.097
60	240	1000	0.137
120	480	1000	0.164
240	960	1000	0.9

[GNS]: $0.01 \leq \nu \leq 0.04$.

N_x	N_t	ν	iter	Res	RMSE	time
30	240	0.04	2	4.1e-06	0.0083	6e01
60	720	0.03	3	1.6e-07	0.0057	1e02
120	1920	0.02	5	1.0e-06	0.0068	7e02
240	3840	0.01	112	4.5e-06	0.0040	5e04

$$f_j^{\text{GS}}(u^j, \rho) = \frac{1}{2} \left(\frac{u^j}{u_{\max}^j} \right)^2 - \frac{u^j}{u_{\max}^j} + \frac{s}{s_{jam}}, \quad f_j^{\text{GNS}}(u^j, \rho) = \frac{1}{2} \left(\frac{u^j}{u_{\max}^j} \right)^2 - \frac{u^j}{u_{\max}^j} + \frac{u^j}{u_{\max}^j} \frac{s}{s_{jam}}$$

$$s = \sum_{j=1}^J \frac{\rho^j}{\rho_j} \in [0, 1], \quad s_{jam} = J$$

Parallel solver

- ▶ Preconditioner $\mathcal{J}_0 = \mathcal{J}(w^{(0)})$, $\mathcal{J}_0^{-1} \mathcal{J}(w^{(k)}) = I + \mathcal{J}_0^{-1} (\mathcal{J}(w^{(k)}) - \mathcal{J}(w^{(0)}))$.
- ▶ Parallel solver
 - MPI [Dalcin and Fang 2021] : inter-process communication.
 - PETSc [Balay et al. 2019] : large-scale numerical simulations.
 - Pyccel [Bourne et al. 2023] : C/Fortran compiled-style kernel.
- ▶ Percentage improvement of the saved time: $PIST = \left(\frac{t_{old} - t_{new}}{t_{old}} \times 100 \right) \%$.
- ▶ GLWR ($J = 1$), PIST: LGMRES $\xrightarrow{98\%}$ 1 core $\xrightarrow{71\%}$ 16 cores $\xrightarrow{<20\%}$ 56 cores.

				parallel (CPU cores)								RMSE
		LGMRES		1		16		56		112		
N_x	N_t	iter	time	iter	time	iter	time	iter	time	iter	time	
30	120	2	30.4822	4	0.6479	4	0.1957	4	0.1865	4	0.1614	0.0289
60	240	3	173.088	4	2.9317	4	0.8708	4	0.7639	4	0.6494	0.0226
120	480	3	702.323	4	13.152	4	3.9964	4	3.5443	4	2.8693	0.0161
240	960	3	9637.26	4	61.594	4	17.838	4	16.601	4	12.605	0.0114
480	1920	3	72571.4	4	312.04	4	81.798	4	74.985	4	56.501	0.0080
960	3840	-	-	-	-	4	415.64	4	342.05	4	265.41	0.0056
1920	7680	-	-	-	-	-	-	4	2005.9	4	1374.0	0.0039

TOUBKAL supercomputer, PowerEdge R840 server, 4 Intel Xeon Platinum 8276 CPUs (112 cores in total), 1.5 TB of RAM,

Performance CPU governor mode, SLURM-managed jobs.

Parallel solver

► GS ($J = 1$): LGMRES $\xrightarrow{85\%}$ 1 core $\xrightarrow{74\%}$ 16 cores $\xrightarrow{<20\%}$ 56 cores.

		parallel (CPU cores)										RMSE
		LGMRES		1		16		56		112		
N_x	N_t	iter	time	iter	time	iter	time	iter	time	iter	time	
30	120	4	3.77560	5	0.8363	5	0.2356	5	0.21561	5	0.23177	0.0204
60	240	5	22.8240	5	4.0176	5	1.1025	5	0.92241	5	0.92102	0.0136
120	480	6	134.806	6	23.654	6	6.3526	6	5.33464	6	5.06342	0.0087
240	960	9	40000.0	6	128.77	6	31.042	6	26.4452	6	24.0049	0.0054
480	1920	-	-	7	924.39	7	193.27	7	154.744	7	114.693	0.0032
960	3840	-	-	-	-	9	2188.8	9	1277.61	9	1183.04	0.0019
1920	7680	-	-	-	-	-	-	10	13025.0	10	12151.2	0.0011

► GNS ($J = 1$): LGMRES $\xrightarrow{40\%}$ 1 core $\xrightarrow{77\%}$ 16 cores $\xrightarrow{<20\%}$ 56 cores.

		parallel (CPU cores)										RMSE
		LGMRES		1		16		56		112		
N_x	N_t	iter	time	iter	time	iter	time	iter	time	iter	time	
30	120	4	1.9892	9	1.4183	9	0.3808	9	0.33785	9	0.36964	0.0142
60	240	4	8.6943	11	8.4206	11	2.0786	11	1.66433	11	1.63943	0.0086
120	480	5	89.412	15	56.405	15	13.663	15	11.0040	15	10.2259	0.0050
240	960	5	5000.0	17	350.53	17	73.656	17	59.1687	17	52.8208	0.0028
480	1920	-	-	19	2417.9	19	406.71	19	317.431	19	281.389	0.0015
960	3840	-	-	-	-	22	3445.0	22	2021.92	22	1775.82	0.0008
1920	7680	-	-	-	-	-	-	25	21228.1	25	20045.9	0.0004

- 1 Introduction and Motivation
- 2 From finite-player differential games to mean-field games
- 3 Solving the mean-field game PDE system using Newton-GMRES solver
- 4 From mean-field games back to finite player differential games**
- 5 Conclusion

Discrete optimal controls

- ▶ NMFE-constructed controls $(\hat{v}_{i_j})_{i_j \in \mathcal{I}_j, j \in \{1, \dots, J\}}$

$$\begin{cases} \hat{v}_{i_j}(t) = u^{j,*}(t, x_{i_j}(t)), \\ \dot{x}_{i_j} = \hat{v}_{i_j}(t), \quad x_{i_j}(0) = x_{i_j,0}, \end{cases}$$

$(x_{i_j,0})_{i_j \in \mathcal{I}_j}$: N_j random samples generated from $\rho^j(0, x)$.

- ▶ ε -Nash equilibrium $(\tilde{v}_{i_j}(t))_{i_j \in \mathcal{I}_j, j \in \{1, \dots, J\}}$

$$\forall v_{i_j} \in [0, u_{max}^j], \mathcal{L}_{i_j}(\tilde{v}_{i_j}, \tilde{v}_{-i_j}, \tilde{v}^{-j}) \leq \mathcal{L}_{i_j}(v_{i_j}, \tilde{v}_{-i_j}, \tilde{v}^{-j}) + \varepsilon_{i_j}.$$

[Huang et al. 2006; Lasry and Lions 2007; Carmona and Delarue 2012; Bensoussan et al. 2014; Carmona and Delarue 2018a; Carmona and Delarue 2018b]

- ▶ ε -Nash equilibrium with lower bound $\hat{\varepsilon}_{i_j}$

$$\hat{\varepsilon}_{i_j} = \max_{v_{i_j} \in [0, u_{max}^j]} \left\{ \mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) - \mathcal{L}_{i_j}(v_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) \right\}.$$

Best responses

- ε -Nash equilibrium with lower bound $\hat{\varepsilon}_{i_j}$

$$\begin{aligned}\hat{\varepsilon}_{i_j} &= \mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) - \min_{v_{i_j} \in [0, u_{max}^j]} \left\{ \mathcal{L}_{i_j}(v_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) \right\} \\ &= \mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) - \mathcal{L}_{i_j}(\bar{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}).\end{aligned}$$

exploitability in game theory, $\hat{\varepsilon}_{i_j} \approx 0$ suggests $\hat{v}_{i_j} \rightarrow \bar{v}_{i_j}$.

- Best responses $(\bar{v}_{i_j}(t))_{i_j \in \mathcal{I}_j, j \in \{1, \dots, J\}}$

$$\bar{v}_{i_j} = \operatorname{argmin}_{v_{i_j} \in [0, u_{max}^j]} \left\{ \mathcal{L}_{i_j}(v_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j}) \right\},$$

$$\dot{x}_m = v_m(t), \text{ if } m = i_j, \quad \dot{x}_m = \hat{v}_m(t), \text{ if } m \neq i_j, \quad x_m(0) = x_{m,0}.$$

adjoint equation $-\dot{P}_{i_j}(t) = \nabla_x f_j(v_{i_j}(t), \rho(t, x_{i_j}(t)))$, $P_{i_j}(T) = 0$,

gradient function $g_{i_j}(t) = \nabla_v f_j(v_{i_j}(t), \rho(t, x_{i_j}(t))) + P_{i_j}(t)$,

gradient descent method $v_{i_j}^{(k+1)}(t) = v_{i_j}^{(k)}(t) - \tau_k g_{i_j}^{(k)}(t)$.

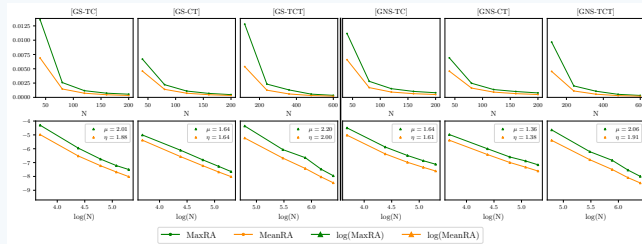
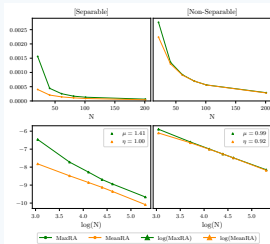
ε-Nash validation

► Relative accuracy:

- $\text{MaxRA} = \max_j \max_{i_j \in \mathcal{I}_j} |\hat{\varepsilon}_{i_j}| / \max_j \max_{i_j \in \mathcal{I}_j} |\mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j})|$.
- $\text{MeanRA} = \sum_j \sum_{i_j \in \mathcal{I}_j} |\hat{\varepsilon}_{i_j}| / \sum_j \sum_{i_j \in \mathcal{I}_j} |\mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j})|$.

► Theorem 6.7 [Carmona and Delarue 2018b] $\Rightarrow |\hat{\varepsilon}_{i_j}| \leq CN^{-1/4}, \forall j \in \{1, \dots, J\}, i_j \in \mathcal{I}_j$.

$$|\mathcal{L}_{i_j}(\hat{v}_{i_j}, \hat{v}_{-i_j}, \hat{v}^{-j})| \geq c_0 > 0 \Rightarrow \text{MaxRA}, \text{MeanRA} \leq \frac{C}{c_0} N^{-1/4}.$$



$$\text{MaxRA} \sim C_{\max} N^{-\mu}, \text{MeanRA} \sim C_{\text{mean}} N^{-\eta}, \mu, \eta > \frac{1}{4}.$$

- 1 Introduction and Motivation
- 2 From finite-player differential games to mean-field games
- 3 Solving the mean-field game PDE system using Newton-GMRES solver
- 4 From mean-field games back to finite player differential games
- 5 Conclusion**

► Contributions

- Extending the micro-macro bridge from single-class to multi-class traffic.
- Combining tailored numerical methods and parallel implementations.
- Solving very large systems with up to 88M unknowns and 600 vehicles.
- Original micro-macro comparisons for different traffic scenarios.
- Numerically validate ϵ -Nash with convergence rates better than theoretically expected.



PhD dissertation



[Machtalay et al. 2025]

► Future directions

- Richer traffic models (second-order dynamics, multi-lane traffic)
- Theoretical analysis (existence and uniqueness of solutions, convergence)
- Scalable numerical methods (e.g., improved preconditioners)

Thank you for your attention!

References I

- Hamilton, W. R. (1834). "On a General Method in Dynamics". In: *Philosophical Transactions of the Royal Society of London* 124, pp. 247–308.
- Jacobi, C. G. J. (1884). *Vorlesungen über Dynamik*. Ed. by A. Clebsch. 2nd ed. Lectures delivered at the University of Königsberg (1842–1843). Berlin: Georg Reimer.
- Fokker, A. D. (1914). "Die mittlere Energie rotierender elektrischer Dipole im Strahlungsfeld". In: *Annalen der Physik* 348.5, pp. 810–820.
- Planck, M. (1917). "Über einen Satz der statistischen Dynamik und seine Erweiterung in der Quantentheorie". In: *Sitzungsberichte der Königlich-Preussischen Akademie der Wissenschaften zu Berlin XXIV*. Physikalisch-mathematische Classe vom 10. Mai, pp. 324–341.
- Kolmogorov, A. (1931). "Über die analytischen Methoden in der Wahrscheinlichkeitsrechnung". In: *Mathematische Annalen* 104.1, pp. 415–458.
- V. Neumann, J. and O. Morgenstern (1944). *Theory of games and economic behavior*. Princeton university press.
- Nash, J. F. (1950). "Equilibrium Points in N-Person Games.". In: *Proceedings of the National Academy of Sciences of the United States of America* 36 1, pp. 48–9.
- (1951). "Non-Cooperative Games". In: *Annals of Mathematics* 54.2, pp. 286–295.
- Bellman, R. (1954). "The theory of dynamic programming". In: *Bulletin of the American Mathematical Society* 60, pp. 503–515.
- Lighthill, M. J. and G. B. Whitham (May 10, 1955). "On kinematic waves II. A theory of traffic flow on long crowded roads". In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*.
- Richards, P. I. (Feb. 1, 1956). "Shock Waves on the Highway". In: *Operations Research* 4.1, pp. 42–51.
- Bellman, R. (1957). *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press.
- Bensoussan, A. and J. Frehse (1984). "Nonlinear elliptic systems in stochastic game theory.". In: *Journal für die reine und angewandte Mathematik* 350, pp. 23–67.
- Huang, M. et al. (Jan. 1, 2006). "Large population stochastic dynamic games: Closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle". In: *Communications in Information & Systems* 6.3, pp. 221 –252.
- Lasry, J.-M. and P.-L. Lions (2007). "Mean field games". In: *Japanese journal of mathematics* 2.1, pp. 229–260.

References II

- Guéant, O. et al. (2011). “Mean Field Games and Applications”. In: *Paris-Princeton Lectures on Mathematical Finance 2010*, pp. 205–266.
- Carmona, R. A. and F. Delarue (2012). “Probabilistic Analysis of Mean-Field Games”. In: *SIAM J. Control. Optim.* 51, pp. 2705–2734.
- Couillet, R. et al. (2012). “Electrical Vehicles in the Smart Grid: A Mean Field Game Analysis”. In: *IEEE Journal on Selected Areas in Communications* 30, pp. 1086–1096.
- Bensoussan, A. et al. (2013). *Mean field games and mean field type control theory*. Vol. 101. Springer.
- Nourian, M. and P. E. Caines (2013). “Epsilon-Nash mean field game theory for nonlinear stochastic dynamical systems with major and minor agents”. In: *SIAM Journal on Control and Optimization* 51.4, pp. 3302–3331.
- Bensoussan, A. et al. (2014). “Mean Field Games with a Dominating Player”. In: *Applied Mathematics & Optimization* 74, pp. 91–128.
- Gomes, D. A. et al. (2014). “Socio-economic applications of finite state mean field games”. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 372.
- Camilli, F. et al. (2015). “A model problem for Mean Field Games on networks”. In: *Discrete and Continuous Dynamical Systems* 35.9, pp. 4173–4192.
- Kolokoltsov, V. N. and A. Bensoussan (2015). “Mean-Field-Game Model for Botnet Defense in Cyber-Security”. In: *Applied Mathematics & Optimization* 74, pp. 669–692.
- Bauso, D. et al. (2016). “Opinion Dynamics in Social Networks through Mean-Field Games”. In: *SIAM Journal on Control and Optimization* 54.6, pp. 3225–3257.
- Cacace, S. et al. (2017). “A numerical method for mean field games on networks”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 51.1, pp. 63–88.
- Chan, P. and R. Sircar (2017). “Fracking, Renewables, and Mean Field Games”. In: *SIAM Rev.* 59, pp. 588–615.
- Bensoussan, A. et al. (Oct. 29, 2018). “Mean Field Control and Mean Field Game Models with Several Populations”. In: *Minimax Theory and its Applications* 3.2, SI, 173–209.
- Carmona, R. A. and F. Delarue (2018a). “Probabilistic Theory of Mean Field Games with Applications I: Mean Field FBSDEs, Control, and Games”. In.

References III

- Carmona, R. and F. Delarue (2018b). *Probabilistic Theory of Mean Field Games with Applications II Mean Field Games with Common Noise and Master Equations*. Springer.
- Koulali, M. A. et al. (2018). “Industrial Internet of Things-Based Prognostic Health Management: A Mean-Field Stochastic Game Approach”. In: *IEEE Access* 6, pp. 54388–54395.
- Węglarczyk, S. (2018). “Kernel density estimation and its application”. In: *ITM Web of Conferences*. Vol. 23. EDP Sciences, p. 00037.
- Balay, S. et al. (2019). *PETSc users manual*. Tech. rep. Argonne National Laboratory.
- Barreiro-Gomez, J. and H. Tembine (2019). “Blockchain Token Economics: A Mean-Field-Type Game Perspective”. In: *IEEE Access* 7, pp. 64603–64613.
- Chassagneux, J.-F. et al. (2019). “Numerical method for FBSDEs of McKean–Vlasov type”. In: *The Annals of Applied Probability*.
- Huang, K. et al. (2020). “A Game-Theoretic Framework for Autonomous Vehicles Velocity Control: Bridging Microscopic Differential Games and Macroscopic Mean Field Games”. In: *Discrete and Continuous Dynamical Systems-Series B* 25.12, pp. 4869–4903.
- Lacker, D. (2020). “On the convergence of closed-loop Nash equilibria to the mean field game limit”. In: *The Annals of Applied Probability* 30.4, pp. 1693–1761.
- Ruthotto, L. et al. (2020). “A machine learning framework for solving high-dimensional mean field game and mean field control problems”. In: *Proceedings of the National Academy of Sciences* 117.17, pp. 9183–9193.
- Tembine, H. (2020). “Covid-19: Data-Driven Mean-Field-Type Game Perspective”. In: *Games* 11.4, p. 51.
- Achdou, Y. and Z. Kobeissi (2021). “Mean field games of controls: Finite difference approximations”. In: *Mathematics in Engineering* 3.3, pp. 1–35.
- Dalcin, L. and Y.-L. L. Fang (2021). “mpi4py: Status Update After 12 Years of Development”. In: *Computing in Science & Engineering* 23, pp. 47–54.
- Laurière, M. (2021). “Numerical Methods for Mean Field Games and Mean Field Type Control”. In: *ArXiv abs/2106.06231*.
- Flandoli, F. et al. (2022). “N-player games and mean field games of moderate interactions”. In: *Applied Mathematics & Optimization* 85.3, p. 38.

References IV

- Bonnans, J. F. et al. (2023). “Discrete potential mean field games: duality and numerical resolution”. In: *Mathematical Programming* 202.1-2, pp. 241–278.
- Bourne, E. et al. (2023). “Pyccel: a Python-to-X transpiler for scientific high-performance computing”. In: *Journal of Open Source Software* 8.83, p. 4991.
- Baudelet, S. et al. (2024). “Deep learning for mean field optimal transport”. In: *ESAIM: ProcS* 77, pp. 145–175.
- Carlini, E. et al. (2024). “A Lagrange–Galerkin scheme for first order mean field game systems”. In: *SIAM Journal on Numerical Analysis* 62.1, pp. 167–198.
- Mo, Z. et al. (2024). “A Game-Theoretic Framework for Generic Second-Order Traffic Flow Models Using Mean Field Games and Adversarial Inverse Reinforcement Learning”. In: *Transportation Science* 58.6, pp. 1403–1426.
- Machtalay, A. et al. (2025). “Computational investigations of a multi-class traffic flow model: Mean-field and microscopic dynamics”. In: *Transportation Research Part B: Methodological* 195, p. 103196.