Multiagent decision-making and control

Project information

Maryam Kamgarpour

Professor of Engineering (IGM, STI), EPFL

ME-429, Spring 2025, EPFL

Project information

We will provide suggested topics by week 3, and you can choose your topic from this list or propose your idea by week 5. Throughout the semester, you will work on the project with your partner.
You can do the following for projects:

- theoretical analysis of a general concept in games such as efficiency of equilibria and mechanism design in certain classes of games (e.g. electricity markets, transportation, autonomous driving).
- experiment: you can do an experimental study via implementing a multiagent decision-making problem.
- Problem formulation: you can formulate an engineering application, e.g. multi-robot systems, transportation, communication or energy systems, using a game theoretic framework discussed in class.

Example of past projects

- Dynamic M-Player Constrained Game for Collision Avoidance and Trajectory Planning in Serial Manipulators
- Dynamic Pricing in Ridesharing Apps
- Congestion Game: Modeling the Ischgl ski resort
- Multi-Agent Games for the Smart Grid Energy Management
- Multi Type Partially Observable Mean Field Multi-Agent Reinforcement Learning
- Multi-robot motion planning with kinematic constraints in dynamic potential games

Virtual bidding in multi-stage electricity markets

- In electricity markets, energy producers and consumers negotiate prices similar to traders in financial markets. As energy supply needs to be planned for a few days in advance, this market has been divided into a so-called forward stage where “virtual bids“ are made via contracts for future energy supplies, and a balancing stage where the real energy product is exchanged. Read more: [1, 3]
- **Questions / Ideas:**
  - How can the two-stage market structure be accurately modelled via a game theoretic formulation?
  - What methods could market participants use to algorithmically determine optimal bidding strategies given their estimated future demand/supply?
  - Could we use real-world electricity market data to simulate the performance achieved by different bidding strategies?

Zero-sum control games

- In some applications, e.g. autonomous racing, we aim to compute/learn policies for an agent in an environment in which it is competing with another learning agent. In particular, [7] models a two-vehicle racing scenario as a zero-sum game, while additionally incorporating constraints to avoid collisions among the vehicles. Another result on learning in zero-sum (linear quadratic) games that may be useful is [8].
- **Questions / Ideas:**
  - Implement a known algorithm for learning/control in a 2-player zero-sum game application. This could e.g. be a simulation for the racing scenario (can be significantly simplified compared to [7]).
  - In your simulation, study methods for achieving safe/robust interactions among the agents (e.g. no cars crashing). This could be achieved e.g. via a coupled constraint formulation as in [7].

Lyapunov iteration for linear quadratic games

- Computing the Nash linear feedback policy in infinite horizon linear quadratic games is a challenging problem. The solution is characterized by a set of coupled Riccati equations but solving these nonlinear equations is an open challenge [2]. Past work has proposed a so-called Lyapunov iteration to address this problem [4]. However, no proof of convergence of the approach is provided. In this project, you will formalize the Lyapunov iteration and explore approaches for proving convergence of the iterative algorithm based on discrete-time nonlinear dynamical system theory. Furthermore, you will support your theory with extensive principled experiments.
- **Questions/ideas**:
  - Formulate the Lyapunov iteration and verify that its fixed points are solution of the Riccati equations corresponding to linear quadratic games.
  - Considering a scalar dynamical system, derive sufficient conditions for the nonlinear system to converge to an equilibrium.
  - For non-scalar systems, evaluate the convergence by implementing the algorithm for randomly generated A, B, C, D matrices.
  - Consider extensions to output feedback setting where players observe noisy measurements.

Dynamic routing in transportation networks

- Consider traffic routing games, an example of congestion games. In this setting, each driver has the objective to minimize travel time from an origin to a destination, while the social welfare consists in total travel times of all agents. This project explores approaches to compute Nash equilibria in congestion games, and aims to evaluate the efficiency of equilibria in these settings. Furthermore, the mechanism design problem is considered, in which a central planner can provide incentives to ensure maximization of social welfare. The theory and algorithms should be verified on realistic traffic models. For background literature, see [6, Section 5], [5, Section VII].
- **Questions/ideas**:
  - Formulate the routing game as a congestion game. Implement an approach to compute the routes that maximize social welfare. Propose approaches to minimize travel times of each agents.
  - Modify your static routing problem to account for simple car dynamics and different arrival times to the network.
  - Explore approaches such as placing traffic lights and road tolls to optimize social welfare for the above setting.

Benchmarking board/card game algorithms

(Requires coding/python skills on a level where you feel comfortable using an advanced library for which only limited documentation is available.)

- OpenSpiel[1] is a comprehensive collection of (mostly board & card) game implementations. It provides the respective game environments as well as implementations of a variety of learning algorithms.
- An experimental benchmark could proceed as follows:
  - Pick one or several among the provided game environments for which training policies seems possible with limited resources.
  - Select a small subset of the pre-implemented algorithms which you can understand and for which you can train policies on the selected games.
  - Compare the results by letting policies obtained via different methods compete against each other.
  - Come up with meaningful ways (e.g. a ranking) to report the outcomes.

[1] https://github.com/google-deepmind/open_spiel

References

[1] Metin Celebi, Attila Hajos, and Philip Q Hanser. “Virtual bidding: the good, the bad and the ugly”. In: *The Electricity Journal* 23.5 (2010), pp. 16–25.

[2] João P Hespanha. *Noncooperative game theory: An introduction for engineers and computer scientists*. Princeton University Press, 2017.

[3] William W Hogan. “Virtual bidding and electricity market design”. In: *The Electricity Journal* 29.5 (2016), pp. 33–47.

[4] TY Li and Z Gajic. “Lyapunov iterations for solving coupled algebraic Riccati equations of Nash differential games and algebraic Riccati equations of zero-sum games”. In: *New trends in dynamic games and applications*. Springer, 1995, pp. 333–351.

[5] Dario Paccagnan et al. “Nash and Wardrop equilibria in aggregative games with coupling constraints”. In: *IEEE Transactions on Automatic Control* 64.4 (2018), pp. 1373–1388.

[6] Pier Giuseppe Sessa et al. “Contextual games: Multi-agent learning with side information”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 21912–21922.

[7] Mingyu Wang et al. “Game Theoretic Planning for Self-Driving Cars in Competitive Scenarios”. In: (). URL: https://roboticsproceedings.org/rss15/p48.pdf.

[8] Kaiqing Zhang, Zhuoran Yang, and Tamer Basar. “Policy optimization provably converges to Nash equilibria in zero-sum linear quadratic games”.