Explicit receding-horizon dynamic games

Emilio Benenati*, and Giuseppe Belgioioso*
*Division of Decision and Control Systems, KTH Royal Institute of Technology, Sweden

***Abstract*—In the context of non-cooperative constrained multi-agent systems, we show that the finite-horizon coupled optimal control problem (also known as dynamic game) admits a solution which is an affine function of the initial state when the agents' objectives are quadratic and the constraints are affine. We present an algorithm for computing explicitly such solution for every initial state, thus reducing the online computational burden needed in the implementation of non-cooperative receding horizon controllers.**

I. Introduction

Dynamic games have recently emerged as a modeling framework for non-cooperative multi-agent control problems with applications that span from autonomous driving [1] and racing [2], [3] to supply chains management [4]. In these applications, dynamic games are typically solved in receding-horizon, that is, by recomputing a finite-horizon open-loop Nash equilibrium (ol-NE) input sequence [5] at each time-step, and by applying the first input of the resulting sequence. Specifically, the receding-horizon game framework leads to complex interactive behaviors [6], as every agent effectively models the other agents as rational entities, and the recomputation enables real-time adaptation to disturbances within operating constraints. This method generalizes the commonly employed Model Predictive Control (MPC) architecture to the case of non-cooperative agents. Crucially, the sampling time of the resulting control action is highly dependent on the solution speed of the finite-horizon ol-NE problem. For example, the authors in [7] show that the computation time required by two state-of-the-art solvers for nonlinear constrained games (ALGAMES and iLQGames) is estimated to $860 \pm 251\text{ms}$ and $705 \pm 209\text{ms}$, respectively, for an autonomous driving scenario with only 4 agents. This computation time would result in a controller with update frequency of approximately 2Hz. In this work, we reduce the online computational burden for the subclass of strongly monotone dynamic games with linear constraints and quadratic objectives (LQ games) by extending the Explicit MPC concept [8] to game-theoretic settings. First, we reformulate the finite-horizon LQ game as an affine variational inequality (AVI) [9]. Then, we observe that the resulting optimal control trajectory satisfying the Karush-Kuhn-Tucker (KKT) conditions of the AVI [9, §1.3.2] can be expressed as a piecewise-affine function of the initial state. This observation allows one to derive an explicit expression of the ol-NE for every initial state via an offline computation. The online implementation of the controller then simply amounts to the evaluation of a piecewise-affine function. This approach could significantly improve the performance of receding-horizon game theoretic controllers in applications where limited computing capabilities is available and high clock frequencies are required, such as multi-agent autonomous driving.

II. Problem formulation

We consider N non-cooperative agents, where each agent i has decision authority over a control input u_i for the system

$$x[t+1] = Ax[t] + \sum_{i=1}^{N} B_i u_i[t]. \tag{1}$$

We denote as $\phi(x_0, u_i, \boldsymbol{u}_{-i}; t)$ the state evolution at time t from the initial state x_0 of the system in (1) when the agent i applies the input sequence u_i and the remaining agents apply the input $\boldsymbol{u}_{-i} = (u_j)_{j \neq i}$. Let us define the ol-NE $\boldsymbol{u}^* = (u_i^*)_{i=1,\dots,N}$ as the input sequence that, for all i, solves the optimal control problem with horizon T:

$$u_i^* \in \arg \min_{u_i \in \mathbb{R}^{mT}} \quad J_i(x_0, u_i, \boldsymbol{u}_{-i}^*) \tag{2a}$$

$$\text{s.t.} \quad C^{\mathrm{x}} \phi(x_0, u_i, \boldsymbol{u}_{-i}^*; t) + c^{\mathrm{x}} \leq 0, \quad \forall t \tag{2b}$$

$$C_i^{\mathrm{u}} u_i[t] + c_i^{\mathrm{u}} + \textstyle\sum_{j \neq i} C_j^{\mathrm{u}} u_j^*[t] \leq 0, \quad \forall t, \tag{2c}$$

where the objective is quadratic:

$$J_i(x_0, u_i, \boldsymbol{u}_{-i}) := \frac{1}{2}\Big(\|\phi(x_0, u_i, \boldsymbol{u}_{-i}; T)\|_{P_i}^2 + \sum_{t=0}^{T-1} \|\phi(x_0, u_i, \boldsymbol{u}_{-i}; t)\|_{Q_i}^2 + \|u_i\|_{R_i}^2 \Big). \tag{3}$$

An interpretation of the condition in (2) is that each agent simultaneously computes a control trajectory that is the "best response" to the control trajectories of the other agents, while satisfying shared input and state constraints. We are concerned with computing an explicit formulation for the mapping

$$x_0 \mapsto \boldsymbol{u}^* \text{ such that } u_i^* \text{ solves (2) for all } i. \tag{4}$$

III. Methodology

We reformulate the state constraints in (2b) as coupling input constraints with an affine dependence on the initial state x_0 by substituting the dynamics (1) into (2b). Via straightforward calculations, one can find matrices C, Θ and a vector c such that

$$\boldsymbol{u} \text{ satisfies (2b), (2c) for all } t \iff C\boldsymbol{u} + \Theta x_o + c \leq 0.$$

Furthermore, the partial gradients with respect to u_i of the objective function J_i are linear. We then find matrices M, Γ such that

$$\begin{bmatrix} \nabla_{u_1} J_1(x_0, u_1, \boldsymbol{u}_{-1}) \\ \vdots \\ \nabla_{u_N} J_N(x_0, u_N, \boldsymbol{u}_{-N}) \end{bmatrix} = M\boldsymbol{u} + \Gamma x_0. \tag{5}$$

By specifying a known result [10] to the case at hand, if $Q_i, P_i \succeq 0$, $R_i \succ 0$ for all i then an ol-NE for the game in (2) can be found as a solution to the VI

$$\text{find } \boldsymbol{u}^* \text{ such that } \begin{cases} C\boldsymbol{u}^* + \Theta x_o + c \leq 0 \\ (\boldsymbol{u} - \boldsymbol{u}^*)^\top (M\boldsymbol{u} + \Gamma x_0) \geq 0, \end{cases} \tag{6}$$
$$\forall \boldsymbol{u} \text{ such that } C\boldsymbol{u} + \Theta x_o + c \leq 0.$$

Furthermore, the solution to the VI in (6) is unique for each x_0 if $M \succ 0$. The solution to (6) is equivalently the pair $(\boldsymbol{u}^*, \lambda^*)$ that solves the KKT system

$$0 = M\boldsymbol{u}^* + \Gamma x_0 + C^\top \lambda^* \tag{7a}$$
$$0 \in \mathcal{N}_{\mathbb{R}_+}(\lambda^*) - C\boldsymbol{u}^* - \Theta x_0 - c, \tag{7b}$$

where $\mathcal{N}_{\mathbb{R}_+}$ is the normal cone of the set of non-negative real vectors. Given an initial state x_0, denote as $\widetilde{\lambda}$ the multipliers associated with the active constraints at the solution, and as $\widetilde{C}, \widetilde{\Theta}, \widetilde{c}$ the associated rows of C, Θ and c, respectively. Then, if $\widetilde{C}$ is full row rank:

from (7a):

$$\boldsymbol{u}^* = -M^{-1}(\widetilde{C}^\top \widetilde{\lambda} + \Gamma x_0) \tag{8a}$$

substitute in (7b), note $\mathcal{N}_{\mathbb{R}_+}(\widetilde{\lambda}) = 0$:

$$0 = -\widetilde{C}M^{-1}(\widetilde{C}^\top \widetilde{\lambda} + \Gamma x_0) - \widetilde{\Theta} x_0 - \widetilde{c} \tag{8b}$$
$$\iff \quad \widetilde{\lambda} = -(\widetilde{C}M^{-1}\widetilde{C}^\top)^{-1}(\Gamma x_0 + \widetilde{\Theta} x_0 + \widetilde{c}) \tag{8c}$$

substitute in (8a):

$$\boldsymbol{u}^* = M^{-1}\widetilde{C}^\top(\widetilde{C}M^{-1}\widetilde{C}^\top)^{-1}(\Gamma x_0 + \widetilde{\Theta} x_0 + \widetilde{c}) - M^{-1}\Gamma x_0. \tag{8d}$$

Thus, the ol-NE input sequence is an affine function of the initial state for all the initial states whose associated solution has the same active constraints as $\boldsymbol{u}^*$. The region of the state space with these active constraints is a polyhedron characterized by the following inequalities:

$$CM^{-1}\widetilde{C}^\top(\widetilde{C}M^{-1}\widetilde{C}^\top)^{-1}(\Gamma x_0 + \widetilde{\Theta} x_0 + \widetilde{c}) + \Theta x_0 + c - CM^{-1}\Gamma x_0 \leq 0, \tag{9a}$$
$$-(\widetilde{C}M^{-1}\widetilde{C}^\top)^{-1}(\Gamma x_0 + \widetilde{\Theta} x_0 + \widetilde{c}) \geq 0. \tag{9b}$$

where (9a) imposes that the input in (8d) is feasible, and (9b) imposes that the multipliers computed according to (8c) are non-negative. We then derive Algorithm 1 by evaluating via (8d) the control action in the associated region defined by (9) for an initial state x_0. Then, we partition the remainder of the state space into convex polyhedra, and we repeat the process recursively for each of the resulting regions. The resulting piecewise-affine solution mapping can then be used for the online implementation of the controller in Figure 1.

Algorithm 1

INITIALIZATION
Append $\{x \in \mathbb{R}^n | C^{\mathrm{x}} x + c^{\mathrm{x}} \leq 0\}$ to R_list
while R_list is not empty **do**
  $\mathcal{R}_0 \leftarrow$ Pop(R_list)
  Solve (6) for some $x_0 \in \mathcal{R}_0$
  Determine the active constraints, $\widetilde{C}, \widetilde{\Theta}, \widetilde{c}$.
  Compute $\mathcal{C} = \{x \in \mathbb{R}^n | (9)\}$
  Compute $\boldsymbol{u}^*$ for $\mathcal{C}$ via (8d)
  $\{R_i\} \leftarrow$ Partition($\mathcal{R}_0 \setminus \mathcal{C}$)
  Append $\{R_i\}$ to R_list
end while

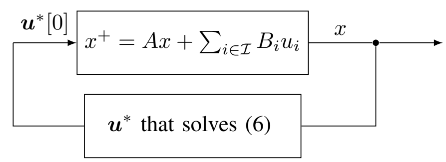

Fig. 1. Block scheme of the closed-loop dynamics with receding-horizon open-loop Nash equilibrium controller.

IV. Conclusion

The open-loop Nash equilibrium of a finite-horizon linear-quadratic, monotone game is a piecewise affine map of the initial state. We present an algorithm to compute such mapping, which extends the well-known framework of explicit MPC to non-cooperative multi-agent scenarios. The explicit computation enables for an online solution of the game, which is of practical interest given the growing interest on game-theoretic receding-horizon control. We aim to further extend this work by showing its effectiveness on a simulated autonomous overtaking system, comparing the computational time with state-of-the-art Nash equilibrium solvers.

References

[1] S. Le Cleac'h, M. Schwager, and Z. Manchester, "LUCIDGames: Online Unscented Inverse Dynamic Games for Adaptive Trajectory Prediction and Planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5485–5492, Jul. 2021.

[2] M. Wang, Z. Wang, J. Talbot, J. C. Gerdes, and M. Schwager, "Game-Theoretic Planning for Self-Driving Cars in Multivehicle Competitive Scenarios," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1313–1325, Aug. 2021.

[3] R. Spica, E. Cristofalo, Z. Wang, E. Montijano, and M. Schwager, "A Real-Time Game Theoretic Planner for Autonomous Two-Player Drone Racing," *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1389–1403, Oct. 2020.

[4] S. Hall, L. Guerrini, F. Dörfler, and D. Liao-McPherson, "Game-theoretic Model Predictive Control for Modelling Competitive Supply Chains," in *8th IFAC Conference on Nonlinear Model Predictive Control*, vol. 58, 2024, pp. 8–14.

[5] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, 2nd ed., ser. Classics in Applied Mathematics. Philadelphia: SIAM, 1999, no. 23.

[6] D. Fridovich-Keil, E. Ratner, L. Peters, A. D. Dragan, and C. J. Tomlin, "Efficient Iterative Linear-Quadratic Approximations for Nonlinear Multi-Player General-Sum Differential Games," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France: IEEE, May 2020, pp. 1475–1481.

[7] S. Le Cleac'h, M. Schwager, and Z. Manchester, "ALGAMES: A fast augmented Lagrangian solver for constrained dynamic games," *Autonomous Robots*, vol. 46, no. 1, pp. 201–215, Jan. 2022.

[8] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," 2002.

[9] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, ser. Springer Series in Operations Research. New York: Springer, 2007.

[10] F. Facchinei and C. Kanzow, "Generalized Nash equilibrium problems," *Annals of Operations Research*, vol. 175, no. 1, pp. 177–211, 2010.