Title

Filippo Ficarola, Gioele Migno, Leonardo Pio Lo Porto

Abstract—abstract test Leo

I. INTRODUCTION

During system design, a crucial step is to determine if the system works according to the desired specifications such as performance and safety. This verification step is challenging for many reasons, the first one is the need to take into account all possible system behaviors, this makes necessary the use of a formal verification method since most simulation-based approaches are insufficient. The other main ones are the presence of unpredictable disturbance in practical systems, and the complexity of system dynamics that in general is nonlinear, evolves in continuous time and has high dimensional state space.

Hamilton-Jacobi Reachability Analysis (HJ-RA) is a verification method for guaranteeing performance and safety properties of systems based on reachability analysis, which computes the set of states from which the system can be driven to a target set, while satisfying time-varying state constraints at all times. HJRA overcomes some of the above challenges, it is applicable to general nonlinear systems and easily handles control and disturbance variables.

II. THEORETICAL FUNDAMENTALS

A. Reachability Analysis

The goal of reachability analysis is to compute the reach-avoid set (RAS) defined as the set of initial states from which the system, using an optimal input, can be driven to a target set within a finite time horizon and satisfying time-varying state constraints at all times. In a reachability problem formulation, the target set can represent a set of undesired states (unsafe), or a set of desired states, in the first case, the RAS contains states to avoid since there exists an optimal input (disturbance) that leads the state into an unsafe region. In the second case instead, the RAS represents safe states from which, applying an optimal input (control law), the system can reach a desired state. In the literature, the just described reach-avoid set, is also called backward reachable set (BRS), in alternative to it, in some cases one might be interested in computing a forward reachable set (FRS), defined as the set of all states that the system can reach from a given initial set of states after a finite time horizon. In order to understand the difference between BRS and FRS, consider a reachability problem in which the target set contains unsafe states, the FRS can be used to check whether the set of possible future states of the system includes undesired states, the BRS instead, can be used to compute, by starting from known unsafe conditions, those states that must be avoided. In this paper HJ-RA is used to compute the BRS of the system, however, it can be used also to compute the FRS. All systems in the real world are subject to a disturbance, hence they have two different kinds of inputs, a controllable one (control) and another uncontrollable (disturbance); for that reason the computation of the BRS can be formulated in terms of a two-player game. For example, consider an aircraft that has to follow a trajectory to complete a task, the system has two inputs: a control input (Player 1) and a disturbance (Player 2), in this scenario, the disturbance could be the wind. Suppose now that there is a goal position to reach (target set) along the trajectory, therefore the control input tries to bring the state at the target and the disturbance to steer it away, in this case, the BRS contains all the initial states for which exists an optimal control command that despite the worst disturbance, brings the system at the goal position. Suppose now instead of a goal, there is an obstacle along the trajectory, now the target set is defined by all states of the system that correspond to a collision with the obstacle, therefore, the BRS contains those states which could lead to a collision despite the best possible control actions. In both cases, the BRS can be computed by studying the outcome of the game between the two players, however those two games are "games of kind" due to the way they are formulated, namely games in which the outcome is binary: system reaches or not the target set. In order to solve this type of game it is necessary to translate it into a "games of degree" in which players want to optimize a cost function, the players have opposite goals, one tries to maximize and the other to minimize it. An approach we will see later, called Level Set Method, can perform this kind of transformation by translating the problem into a standard differential game.

B. Differential Games

To understand better the concept of differential games and how it is formulated, assume we are in the case in which we have a set to reach. In this case we want to reach with our control input (Player 1) the set of safe states of the reachable set; in other words Player 1 tries to bring the system towards the target with its input while in the meanwhile the disturbance (Player 2) tries to steer the system away the target with its input. To have a more intuitive understanding of the inputs, consider that in our example the target set will represent the goal set in a pursuit-evasion game. Our evader (control input) will then be player I and the pursuer (adversarial disturbance) will be player II and the evader tries with its input to reach the goal while the pursuer tries to steer away from reaching it. In a differential game setting, it is important to address what information the players know about each other's decisions. To specify our information pattern, first define a strategy for the first player as a map from the set of disturbances (D) to the set of control inputs (U) which specifies a disturbance signal for player 2 as a function of the disturbance signal that player 1 chooses.

In problems in which our target set is a reach set, we assume that the Player 1 uses only non-anticipative strategies that are defined in this way:

$$\gamma \in \Gamma \triangleq \{\alpha : \mathcal{D} \to \mathcal{U} | d_1(r) \implies \} \tag{1}$$

What the formula says is that, a map $\alpha : D \to U$ is a nonanticipative strategy if for any $s \geq 0$, for any $d_1(\cdot)$ and $d_2(\cdot)$ belonging to D such that $d_1(\cdot)$ and $d_2(\cdot)$ coincide almost everywhere on $[t, s]$, the image $\alpha(d_1(\cdot))$ and $\alpha(d_2(\cdot))$ coincide almost everywhere on $[t, s]$

This restriction means that if player 1 cannot distinguish between input signals $\alpha(d_1(\cdot))$ and $\alpha(d_2(\cdot))$ of player 2 until after time s, then player 1 cannot respond differently $(\alpha[d_1](r) = \alpha[d_2](r))$ to those signals until after time s. Or in other words, player 1 cannot respond differently to two different player 2 disturbances until they become different.

The aim for this problem is based on a two-person-zero-sum differential game. Zero-sum game is a mathematical representation in game theory of a situation which involves two sides, where the result is an advantage for one side and a loss for the other. If the total gains of the participants are added up, and the total losses are subtracted, they will sum to zero. In the context of reachability analysis our aim is to reach the goal by finding the control input (Player 1) that minimizes the distance from the reach set while the disturbance (Player 2) at the same time tries to maximize its action to steer the trajectory away from the set. We can translate this as an optimization problem in which we want to optimize a cost function (that depends on the state, control input and disturbance) of the final state and a running cost that can be seen as a reward accumulated over system trajectories. It is possible to formalize what has been said above in this way: let $J_t(x, a(-), b(-))$ denote the cost accumulated during horizon $[i, f]$ (in Backward Reachable Set the set interval is $[t, 0]$), when Player 1 and Player 2 play control $u(-)$ and $d(-)$ respectively. $J_t(-)$ can be expressed as:

In the zero-sum setting, given this cost function, for the discussion held above, the aim of the game is, for Player 1, to minimize this function, while for Player 2 the aim is to maximize it. The solution of this problem is a binary solution, this means that given our problem formulation we can say if we avoid or not the set considered; or in other words the outcome of the game says if our control input (the evader) wins or not. This kind of differential game problem is also called game of kind. This problem is formulated in this way in order to compute correctly our BRS. Infact, the computation of the Backward Reachable Set is based on the solution of a game of kind. However, our aim is to find what are the control inputs such that we can reach the set; or in other words, considering the pursuit-evasion game, we don't want to know if the evader (our control input) wins or loses but we want to understand which are the control inputs such that we can win the pursuit-evasion game. So we want to pass from a game of kind in which the outcome is a binary value Win = 1, Loss = 0 for Player 1 to a game of degree in which the outcome is a value

in the real space for Player 1 that can give us information about the final result of the game. A solution to do this transformation is the level set method that will be discussed in section 2.5. The BRS, as said in previous section, comes from the solution of a differential game; more precisely, the lower value function V(t,x) (see "Level Set Method" section) is the viscosity solution of a Hamilton-Jacobi-Isaacs (HJI) partial differential equation (PDE) whose the concepts are explained better in the next sections.

C. Hamilton-Jacobi Equation

In physics, the Hamilton-Jacobi equation is an alternative formulation of classical mechanics. It is a first-order nonlinear partial differential equation of the form $H(x, u_x(x, \alpha, t), t) + u_t(x, \alpha, t) = K(\alpha, t)$ with independent variables $(x, t) \in \mathbb{R}^n \times \mathbb{R}$ and parameters $\alpha \in \mathbb{R}^n$. It has wide applications in many scientific fields like mechanics. Its solutions determine infinite families of solutions of Hamilton's ordinary differential equations, which are the equations of motion of a mechanical system. In our case, the family of solutions that come from the computation of the HJI PDE and the variational inequality (see Theorems 1 and 2 in "Problem Formulation") is the viscosity solution.

D. Viscosity Solution

Viscosity solution is a concept introduced by M.Crandall and P. Lions in 1983; it is a notion of solution that allows it to be, for example, nowhere differentiable but for which strong uniqueness theorems, stability theorems and general existence theorems are all valid. It is a generalization of the solution to a partial differential equation (PDE). It has been found that the viscosity solution is that kind of solution to use in many applications of PDE's, including for example first order equations arising in dynamic programming (the Hamilton–Jacobi–Bellman equation) or differential games (the Hamilton–Jacobi–Isaacs equation). This means that, given the classical concept of PDE under the viscosity solution concept, a certain variable u does not need to be everywhere differentiable. There may be points where Du does not exist and yet u satisfies the equation in an appropriate generalized sense. Taking the main concepts of viscosity solutions from M. Crandall, L.C Evans and P.L Lions, (let us first formulate the definition of viscosity solutions in the form we think is the most appealing. We begin by recalling that a function u from 0 into R is said to be differentiable at $y0$ belonging to O and that $Du(y0) = p0$ belonging to R^m, if we have

Here a dot b is the Euclidean scalar product of a and b. and $g(y) = o(|y|y_0|)$ means .

let u be a function from O into R and let y0 belong to O. The superdifferential and subdifferential of u at y_0, denoted, respectively, by $D^+u(y\_0)$ and $D^-u(y\_0)$ are the set of points for which, respectively, the following two inequalities hold.:
and

Given these concepts it is possible to define what is a viscosity solution mathematically. If $D^+u(y\_0)$ and $D^-u(y\_0)$ are nonempty at some x and u is differentiate at x, a viscosity

solution of (0) is a function u belonging to C(O) satisfying the two following conditions:

E. Level Set Method

As mentioned before, in order to compute the BRS is necessary to solve a game of kind where the outcome is boolean: the system state either reaches the target set or not. Level Set Method can be used to translate this game into a game of degree, where players share an objective function to optimize. The basic idea of this approach is to encode the boolean outcome through a quantitative function and compare its value at the end of the game to a threshold value, usually zero, to determine whether or not the system reached the target set. The first step is to define a Lipschitz function $g(x)$, where x represents the system state, such that the target set R corresponds to the zero sublevel set of $g(x)$, that is, $x \in R \Leftrightarrow g(x) \leq 0$. We indicated the target set with R (reach) since from now on we suppose that the set contains goal states, namely states to reach. Now we can define the cost function of the game $J(\cdot)$, we are not interested in any kind of running cost, therefore we consider only the value of $g(x)$ at the end of a game in which $t \in [\tau_i, \tau_f]$:

$$J(x, t, u(\cdot), d(\cdot)) = g(x(\tau_f)) \tag{2}$$

The lower value of the game is given by the following value function $V(x, t)$ in which the control u tries to minimize and the disturbance d to maximize the cost $J(\cdot)$. We assume that the player that wants to reach the target set R, namely the control input u (Player 1), is restricted to use a non-anticipative strategies $\gamma[d](t)$ and we indicates the class of strategies admissible in a time interval $[\tau_i, \tau_f]$ as $\Gamma_{[\tau_i, \tau_f]}$.

$$
\begin{aligned}
V(x, t) &= \inf_{\gamma(\cdot) \in \Gamma(\cdot)} \sup_{d(\cdot)} J(x, t, \gamma(\cdot), d(\cdot)) \\
&= \inf_{\gamma(\cdot) \in \Gamma(\cdot)} \sup_{d(\cdot)} g(x(\tau_f))
\end{aligned}
\tag{3}
$$

In practical scenarios, along the trajectory of a dynamical system there may be both goals to reach and obstacles to avoid. The goals to reach can be represented by the target set R as previously done, the set of states to avoid instead, can be defined with another set A (avoid) that contains all the system state x that corresponds to an object collision, this new kind of set can be defined using a function $h(x)$ similar to $g(x)$. Formally: consider the sets R, A related respectively to the level sets of two Lipschitz continuous and bounded functions $g : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}$, then the two sets can be characterized as:

$$R = \{x \in \mathbb{R}^n | g(x) \leq 0\} \text{ and } A = \{x \in \mathbb{R}^n | h(x) > 0\} \tag{4}$$

The most common choice for the function $g(x)$ and $h(x)$ is to use the distance between the state x and the set of interested, namely:

$$
g(x) = \begin{cases} -d(x, R^c) & \text{if } x \in R \\ d(x, R) & \text{if } x \in R^c \end{cases} \tag{5}
$$

$$
h(x) = \begin{cases} d(x, A^c) & \text{if } x \in A \\ -d(x, A) & \text{if } x \in A^c \end{cases} \tag{6}
$$

Since $g(x)$, $h(x)$ must be bounded, we will see later why, we can introduce two constants C_g C_h to impose a saturation to the distance functions, or alternatively, we can use the arctangent of the signed distance, in this way the resulting functions are bounded and also globally Lipschtz [2]. In the next section we will see how the value function $V(\cdot)$ is formulated when we have both a reach R and an avoid A set, and most importantly how it can be solved in order to calculate the BRS. In the following sections we will refer to the BRS as a reach-avoid set (RAS) to highlight the fact that there is both a set to reach and one to avoid.

III. PROBLEM FORMULATION

Consider the system $\dot{x} = f(x, u, d)$ with $x \in \mathbb{R}^n$, $u \in U \subseteq \mathbb{R}^m$, $d \in D \subseteq \mathbb{R}^p$, $f : \mathbb{R}^n \times U \times D \to \mathbb{R}^n$ and $t \in [\tau_i, \tau_f]$. The inputs $u(\cdot)$, $d(\cdot)$ represent Player 1 (control) and Player 2 (disturbance) respectively, we assume that they are drawn from the set of measurable functions: (WHY??)

$$u(\cdot) \in \mathcal{U}_{[\tau_i, \tau_f]} \triangleq \{\sigma : [\tau_i, \tau_f] \to U | \sigma(\cdot) \text{ is measurable}\}$$

$$d(\cdot) \in \mathcal{D}_{[\tau_i, \tau_f]} \triangleq \{\sigma : [\tau_i, \tau_f] \to D | \sigma(\cdot) \text{ is measurable}\}$$

Consider also two functions $g : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}$ used to represent the target set R and the avoid set A respectively (5) (6). Assume U, D are compact, $f(\cdot)$, $g(\cdot)$, $h(\cdot)$ are bounded and Lipschitz continuous in x and continuous in u and d, therefore, the system dynamics admits a unique trajectory $x(t)$ from an initial state x_i at time τ_i under input $u(\cdot)$ and $d(\cdot)$. We denote this solutions as:

$$\phi(t; x_i, \tau_i, u(\cdot), d(\cdot)) : [\tau_i, \tau_f] \to \mathbb{R}^n$$

In order to define the information patterns of the differential game is necessary to compute the RAS, we assume that the control input u (Player 1), is restricted to use a non-anticipative strategies $\gamma(\cdot)$ as before. Given the previous sets R and A (4), whereby for technical reasons we assume R closed and A open, we can define two different kinds of reachability problems. In the first one we are interested in reaching safely R exactly at the end of the game ($t = \tau_f$), in the second instead, the system can reach safely R at any t inside the time horizon $[\tau_i, \tau_f]$. In the following we will formulate and solve both of them.

A. Reach-avoid at the terminal time

In this first type of reachability problem, we are interested in characterizing the RAS as the set of initial states from which the system trajectory $\sigma(\cdot)$ can start and reach the target set R at the terminal time τ_f, without passing through the avoid set A over the time interval $[\tau_i, \tau_f]$. Formally the RAS contains all the initial states x_i for which there exists an optimal strategy

$\gamma[d](t) \in \Gamma[\tau_i, \tau_f]$ such that for all $d(\cdot) \in \mathcal{D}_{[\tau_i,\tau_f]}$, the system trajectory satisfies $x(\tau_f) \in R$ and $x(t) \in A^c$ for all $t \in [\tau_i, \tau_f]$:

$$RAS_{\tau_f}(t) = \Big\{ x_i \in \mathbb{R}^n | \exists \gamma(\cdot) \in \Gamma_{[t,\tau_f]}, \forall d(\cdot) \in \mathcal{D}_{[t,\tau_f]}$$
$$(\phi(\tau_f; x_i, t, \gamma(\cdot), d(\cdot)) \in R) \quad \wedge$$
$$(\forall \tau \in [t, \tau_f], \phi(\tau; x_i, t, \gamma(\cdot), d(\cdot)) \notin A) \Big\} \quad (7)$$

The cost function $J(\cdot)$ of the game now must take into account also the presence of obstacles along the trajectory, therefore we define it as:

$$J(x, t, u(\cdot), d(\cdot)) = \max \Big\{ g(\mathcal{X}(\tau_f)), \max_{\tau \in [\tau_i, \tau_f]} h(\mathcal{X}(\tau)) \Big\} \quad (8)$$

Where:

$$\mathcal{X}(\tau) = \phi(\tau; x, t, u(\cdot), d(\cdot))$$

Then the value function $V : \mathbb{R}^n \times [\tau_i, \tau_f] \to R$ is given by:

$$V(x, t) = \inf_{\gamma(\cdot) \in \Gamma_{[t,\tau_f]}} \sup_{d(\cdot) \in \mathcal{D}_{[t,\tau_f]}} J(x, t, \gamma(\cdot), d(\cdot))$$

The RAS is linked to the level set of the value function $V(\cdot)$ through the following proposition proved in [3] :

Proposition 1. $RAS_{\tau_f}(t) = \{x \in \mathbb{R}^n | V(x, t) \le 0\}$

We are finally ready to introduce the theorem that allows us to compute $V(\cdot)$ and then thanks to Prop.1, to calculate RAS_{term}. The proof can be found in [3].

Theorem 1. $V(\cdot)$ *is the unique viscosity solution over* $(x, t) \in \mathbb{R}^n \times [\tau_i, \tau_f]$ *of the variational inequality:*

$$\max \Big\{ h(x) - V(x, t), \frac{\partial V}{\partial t}(x, t) + H(x, t) \Big\} = 0$$

$$H(x, t) = \sup_{d \in D} \inf_{u \in U} \frac{\partial V}{\partial t}(x, t) f(x, u, d)$$

with terminal condition:

$$V(x, \tau_f) = \max \{g(x), h(x)\}$$

B. Reach-avoid at any time

The second type of reachability problem is similar to the previous one, however, in this case we are not interested to reach the target set exactly at $t = \tau_f$ but at any time $t \in [\tau_i, \tau_f]$, therefore, the RAS now contains the set of initial states x_i from which the system trajectory can start and, using an optimal control input u, reaches the target set R at some time t without passing through the set A until it hits R.

$$\widetilde{RAS}_t(t) = \Big\{ x_i \in \mathbb{R}^n | \exists \gamma(\cdot) \in \Gamma_{[t,\tau_f]}, \forall d(\cdot) \in \mathcal{D}_{[t,\tau_f]}$$
$$\exists \tau_1 \in [t, \tau_f], (\phi(\tau_1; x_i, t, \gamma(\cdot), d(\cdot)) \in R) \quad \wedge$$
$$(\forall \tau_2 \in [t, \tau_f], \phi(\tau_2; x_i, t, \gamma(\cdot), d(\cdot)) \notin A) \Big\} \quad (9)$$

For technical reasons related to the current differential game [3] [5], it is necessary to define an augmented system dynamics in which Player 1 uses an augmented input $\widetilde{u} = [u, \overline{u}] \in U \times [0, 1]$

$$\widetilde{f}(x, \widetilde{u}, d) = \overline{u} f(x, u, d)$$

Assume $\widetilde{U}, \widetilde{\mathcal{U}}, \widetilde{\Gamma}$ defined similarly to the previous case, we denote the augmented system trajectory as $\widetilde{\phi}(\tau; x_i, t, \widetilde{u}(\cdot), d(\cdot))$. The value function is then similar to the previous case:

$$\widetilde{V}(x, t) = \inf_{\widetilde{\gamma}(\cdot) \in \widetilde{\Gamma}_{[t,\tau_f]}} \sup_{d(\cdot) \in \mathcal{D}_{[t,\tau_f]}} \widetilde{J}(x, t, \widetilde{\gamma}(\cdot), d(\cdot)) \quad (10)$$

Where

$$\widetilde{J}(x, t, \widetilde{u}(\cdot), d(\cdot)) = \max \Big\{ g(\widetilde{\mathcal{X}}(\tau_f)), \max_{\tau \in [\tau_i, \tau_f]} h(\widetilde{\mathcal{X}}(\tau)) \Big\}$$

$$\widetilde{\mathcal{X}}(\tau) = \widetilde{\phi}(\tau; x, t, \widetilde{u}(\cdot), d(\cdot))$$

Also in this case the RAS is linked to the value function $\widetilde{V}(x, t)$, proof in [3].

Proposition 2. *For any* $t \in [\tau_i, \tau_f]$:

$$\widetilde{RAS}_t(t) = \Big\{ x \in \mathbb{R}^n | \widetilde{V}(x, t) \le 0 \Big\}$$

Finally, the following theorem allows us to compute the RAS also in this case. Proof in [3].

Theorem 2. $\widetilde{V}(\cdot)$ *is the unique viscosity solution over* $(x, t) \in \mathbb{R}^n \times [\tau_i, \tau_f]$ *of the variational inequality:*

$$\max \Big\{ h(x) - V(x, t), \frac{\partial \widetilde{V}}{\partial t}(x, t) + \min \Big\{ 0, \widetilde{H}(x, t) \Big\} \Big\} = 0$$

$$\widetilde{H}(x, t) = \sup_{d \in D} \inf_{u \in U} \frac{\partial \widetilde{V}}{\partial t}(x, t) f(x, u, d)$$

with terminal condition:

$$\widetilde{V}(x, \tau_f) = \max \{g(x), h(x)\}$$

IV. Case of Study

case of study test

V. Simulation

simulation

VI. Conclusion

conclusion

References

[1] S. Bansal, Mo Chen, S. Herbert and C. J. Tomlin, "Hamilton-Jacobi Reachability: A Brief Overview and Recent Advances", *arxiv.org/abs/1709.07523*.

[2] K. Margellos and J. Lygeros, "Hamilton-Jacobi formulation for reach-avoid problems with an application to air traffic management", *Proceedings of the 2010 American Control Conference*, 2010, pp. 3045-3050, doi: 10.1109/ACC.2010.5530514.

[3] K. Margellos and J. Lygeros, "Hamilton–Jacobi Formulation for Reach–Avoid Differential Games," *in IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1849-1861, Aug. 2011, doi: 10.1109/TAC.2011.2105730.

[4] Datar, Mandar & Ketkar, Vishwanath & Mejari, Manas & Gupta, Ankit & Singh, N.M. & Kazi, Faruk, "Motion Planning for the Three Wheel Mobile Robot using the Reachable Set Computation under Constraints, (2014)" *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 3. 10.3182/20140313-3-IN-3024.00178.

[5] I. M. Mitchell, A. M. Bayen and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *in IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947-957, July 2005, doi: 10.1109/TAC.2005.851439.