HAL
open science

A Dijkstra-type algorithm for dynamic games

Martino Bardi, Juan Pablo Maldonado Lopez

To cite this version:

Martino Bardi, Juan Pablo Maldonado Lopez. A Dijkstra-type algorithm for dynamic games. 2015. hal-00881218v2

HAL Id: hal-00881218

https://hal.science/hal-00881218v2

Preprint submitted on 2 Mar 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journal Name manuscript No.
(will be inserted by the editor)

A Dijkstra-type algorithm for dynamic games

Martino Bardi · Juan Pablo Maldonado López

Received: date / Accepted: date

Abstract We study zero-sum dynamic games with deterministic transitions and alternating moves of the players. Player 1 aims at reaching a terminal set and minimising a running and final cost. We propose and analyse an algorithm that computes the value function of these games extending Dijkstra's algorithm for shortest paths on graphs. We also show the connection of these games with numerical schemes for differential games of pursuit-evasion type, if the grid is adapted to the dynamical system. Under suitable conditions we prove the convergence of the value of the discrete game to the value of the differential game as the step of approximation tends to zero.

Keywords Zero-sum dynamic games · Dijkstra algorithm · Pursuit-evasion games · Discrete time games

Mathematics Subject Classification (2000) 91A25 · 65Kxx · 49N70 · 91A23 · 91A24 · 49N75

1 Introduction

In this paper we study two-person zero-sum dynamic games where the players move a state variable in a finite state space $\mathcal{X}$. Each action has a (possibly discounted) positive cost for player 1, that he pays to player 2, which depends on the position and actions. Player 1 aims at reaching a given terminal set $\mathcal{X}_f$ and once this is done the game is finished and a final cost is incurred. We adopt a rule of alternating moves that gives an informational advantage to one of the players.

Our purpose is to provide an efficient algorithm to compute the value. We follow an approach inspired by the classical Dijkstra algorithm (Dijkstra 1959) for finding shortest paths in finite graphs, which has running time $O(e + v \log v)$ if a suitable data structure is used, where v, e denote respectively the number of vertices and edges, see Fredman and Tarjan (1987). The algorithm we propose updates the approximate value function only in the immediate neighbours of those nodes where the value is already computed, thus reducing the computation time, and converges in a finite number of steps. In particular, if the running

Partially supported by the Fondazione CaRiPaRo Project "Nonlinear Partial Differential Equations: models, analysis, and control-theoretic problems" and the European Project Marie Curie ITN "SADCO - Sensitivity Analysis for Deterministic Controller Design".

M. Bardi
Università degli Studi di Padova
Dipartimento di Matematica
Via Trieste 63-35121
Padova, Italy.
E-mail: bardi@math.unipd.it

Juan Pablo Maldonado López[1,2,3]
[1] Sorbonne Universités, UPMC Univ Paris 06, UMR 7586,
IMJ-PRG, case 247, 4 place Jussieu, F-75005, Paris, France.
[2] CNRS, UMR 7586, F-75005, Paris, France.
[3] Univ Paris-Diderot, Sorbonne Paris Cité, UMR 7586,
F-75013, Paris, France.
E-mail: pablo.maldonado@imj-prg.fr

and terminal costs are constant the algorithm is *single-pass* as Dijkstra's, namely, the value function is computed only once in each node.

Our assumptions are designed to cover discrete approximations of generalised pursuit-evasion differential games (Bardi and Soravia 1991, Bardi, Falcone, and Soravia 1994, Bardi, Bottacin and Falcone 1995; see also the surveys by Bardi, Falcone and Soravia. 1999 and Cardaliaguet, Quincampoix, and Saint-Pierre 1999 and the references therein) if the grid of the numerical scheme is suitably adapted to the controlled dynamical system. We discuss this connection in detail and prove a convergence result of the value of the discrete game to the value of the differential game as the step of approximation tends to zero. Different from the known theory (e.g., Bardi, Falcone and Soravia 1994 and 1999) we do not require the usual condition $\Delta x/\Delta t \to 0$, where Δx is the mesh size and Δt is the time step.

Our motivation comes from the so-called *Fast Marching Methods* (briefly, FMM) for Hamilton-Jacobi equations with convex Hamiltonian arising in deterministic control and front propagation problems, introduced in Tsitsikilis (1995) and Sethian (1996) and developed by Sethian (1999), Sethian and Vladimirsky (2003), Cristiani (2009), see also the references therein and Cacace, Cristiani, and Falcone (2014) for some recent improvements. These numerical methods approximate time-optimal control in continuous time and space with a fully discrete Bellman equation on a grid, and then rely on the classical Dijkstra algorithm for an efficient solution of the discrete approximation. We recall that the methods based on Dynamic Programming have several good properties, especially robustness, but they face the well-known "curse of dimensionality". A large amount of research in the last twenty years was devoted to overcoming this difficulty in some cases and FMM played an important role for problems with positive costs. We refer to McEneaney (2006), Cacace, Cristiani, Falcone, and Picarelli (2012), and the references therein for other approaches.

Recently various forms of FMM were also proposed for solving some Hamilton-Jacobi-Isaacs equations arising from differential games (Cristiani and Falcone 2006, von Lossow 2007, Grüne and Junge 2008, Cristiani 2009), with possible applications to the stabilization of perturbed systems and to front propagation problems with non-convex Hamiltonian. They approximate the continuous problem with a discrete Isaacs equation on a grid. However, so far there is no theoretical justification for using Dijkstra-type algorithms for discrete dynamic games associated to such equations. One of the goals of this paper is contributing to a rigorous foundation of these methods.

Let us mention that some discrete games related to ours are the reachability games studied in Alfaro et al. (2007), where also some numerical algorithms are provided. Moreover, if the players move simultaneously and use random strategies, the game becomes a special case of the zero-sum stochastic games introduced in the seminal paper of Shapley (1953). Several algorithms have also been proposed to compute the value function of stochastic games, starting with the value iteration algorithm in Shapley (1953). Some variants designed to accelerate the convergence can be found in the survey by Filar and Vrieze (1991), the more recent paper by Raghavan and Syed (2003) and the references therein, and in Kushner (2004), where a Gauss-Seidel procedure for value iteration is studied. The extension of Dijkstra-type algorithms to some stochastic games is an interesting open problem.

2 The discrete dynamic game

2.1 The model

Let $\mathcal{X}$ be a finite set belonging to an Euclidean space $\mathbb{R}^d$, representing the state space of a system and whose elements we call *nodes*. Let A, B be finite sets where the players choose their controls. For a function $S : \mathcal{X} \times A \times B \to \mathcal{X}$ define the trajectory $x_\bullet = x_\bullet(x, a_\bullet, b_\bullet)$ recursively by

$$x_{n+1} = S(x_n, a_n, b_n), \quad x_0 = x. \tag{1}$$

Let $\mathcal{X}_f \subset \mathcal{X}$, denote a *terminal set* of nodes (which player 1 wishes to attain) and let $\gamma \in (0, 1]$ be a discount factor. We introduce the *running* and *terminal* cost

$$\ell : \mathcal{X} \times A \times B \to \mathbb{R}, \ 0 < \ell_0 \le \ell(x, a, b) \le L, \ \forall (x, a, b) \in \mathcal{X} \times A \times B \tag{2}$$

$$g : \mathcal{X}_f \to \mathbb{R}, \ g_0 \le g(x) \le g_1, \forall x \in \mathcal{X}_f \tag{3}$$

and additionally define the *arrival time* $\hat{n} : \mathcal{X} \times A^{\mathbb{N}} \times B^{\mathbb{N}} \to \mathbb{R}$ by

$$\hat{n}(x, a_\bullet, b_\bullet) = \begin{cases} \min\{n \in \mathbb{N} : x_n \in \mathcal{X}_f\}, \text{ if } \{n \in \mathbb{N} : x_n \in \mathcal{X}_f\} \neq \emptyset \\ +\infty \qquad\qquad\qquad\qquad \text{else,} \end{cases}$$

where x_n is the trajectory of (1) corresponding to the control sequences $a_\bullet, b_\bullet$. To alleviate the notation, we will often write $\hat{n}$ instead of the more explicit $\hat{n}(x, a_\bullet, b_\bullet)$ when no confusion arises. We have then the following *total cost* functional $J : \mathcal{X} \times A^{\mathbb{N}} \times B^{\mathbb{N}} \to \mathbb{R}$

$$J(x, a_\bullet, b_\bullet) := \sum_{n=0}^{\hat{n}-1} \ell(x_n, a_n, b_n)\gamma^n + \gamma^{\hat{n}} g(x_{\hat{n}}),$$

where the discount factor γ satisfies $0 < \gamma \leq 1$. Observe that if $\hat{n} = +\infty$ the cost is finite for $\gamma < 1$ and $+\infty$ for $\gamma = 1$ (i.e., no discount). Player 1 chooses $a_\bullet \in A^{\mathbb{N}}$ and player 2 chooses $b_\bullet \in B^{\mathbb{N}}$. The aim of player 1 is to minimize the cost functional, whereas player 2 has the opposite goal.

We assume that both players observe each other's actions and the state $x_\bullet$.

We refer to $\mathcal{G} = \mathcal{G}\langle \mathcal{X}, \mathcal{X}_f, S, A, B, \ell, g, \gamma \rangle$ as *the game.*

2.2 The lower value function

We consider an information pattern where player 1 is informed in advance of the action that player 2 will play at each time. Although not realistic in many situations, this is relevant in the discretization of the lower value of a differential game, as we will see in Section 3, and also in the case of discrete robust control problems, where player 2 represents a disturbance.

Definition 1 A map $\alpha : B^{\mathbb{N}} \to A^{\mathbb{N}}$ is a *non anticipating strategy* for player 1 if

$$b_n = \tilde{b}_n, \forall n \leq m \implies \alpha[b_\bullet]_n = \alpha[\tilde{b}_\bullet]_n, \forall n \leq m.$$

Denote with $\mathcal{A}$ the set of non anticipating strategies for player 1. The definition of the set $\mathcal{B}$ of non anticipating strategies for player 2 is completely analogous.

This allows us to introduce the *lower value function*

$$V^-(x) := \inf_{\alpha \in \mathcal{A}} \sup_{b_\bullet \in B^{\mathbb{N}}} J(x, \alpha[b_\bullet], b_\bullet).$$

The following result follows from familiar arguments, see for instance Chapter 8, Theorem 3.18 in Bardi and Capuzzo-Dolcetta 1997.

Proposition 1 *The lower value function satisfies*

$$V^-(x) = \inf_{\alpha \in \mathcal{A}} \sup_{b_\bullet \in B^{\mathbb{N}}} \left\{ \sum_{n=0}^{k \wedge \hat{n}-1} \ell(x_n, \alpha[b_\bullet]_n, b_n)\gamma^n + \gamma^{k \wedge \hat{n}} V^-(x_{k \wedge \hat{n}}) \right\}, \forall k \in \mathbb{N}. \tag{4}$$

$$V^-(x) = \max_{b \in B} \min_{a \in A} \left\{ \ell(x, a, b) + \gamma V^-(S(x, a, b)) \right\}, \ \forall x \notin \mathcal{X}_f \tag{5}$$

$$V^-(x) = g(x), \ \forall x \in \mathcal{X}_f. \tag{6}$$

The first equality (4) is the well known dynamic programming property. By taking $k = 1$ in (4) one can easily prove (5). The last equality (6) follows directly from the definition.

The following form of the dynamic programming property will be useful later.

Proposition 2 *Let $\mathcal{X}_f \subset \tilde{\mathcal{X}} \subset \mathcal{X}$ and let $\tilde{n}$ denote the arrival time to $\tilde{\mathcal{X}}$, i.e. $\tilde{n} = \tilde{n}(x, a_\bullet, b_\bullet) = \inf\{n \in \mathbb{N} : x_n \in \tilde{\mathcal{X}}\}$. Then*

$$V^-(x) = \inf_{\alpha \in \mathcal{A}} \sup_{b_\bullet \in B^{\mathbb{N}}} \left\{ \sum_{n=0}^{\tilde{n}-1} \ell(x_n, \alpha[b_\bullet]_n, b_n)\gamma^n + \gamma^{\tilde{n}} V^-(x_{\tilde{n}}) \right\}.$$

Proof This is a direct consequence of the dynamic programming property (4) since $\tilde{n} \leq \hat{n}$.

2.3 The algorithm

The following algorithm computes the lower value function:

Require: $n = 0, \mathrm{Acc}_0 := \mathcal{X}_f$, $W_0(x) := +\infty, \forall x \in \mathcal{X}$, $V_0^-(x) = g(x), \forall x \in \mathcal{X}_f$
  while $\mathrm{Acc}_n \neq \mathcal{X}$ **do**
    for $x \in \mathcal{X} \setminus \mathrm{Acc}_n, b \in B$ **do**
      $A_n(x,b) := \{a \in A : S(x,a,b) \in \mathrm{Acc}_n\}$
    end for
  end while
  $\mathrm{Cons}_n := \{x \in \mathcal{X} \setminus \mathrm{Acc}_n : A_n(x,b) \neq \emptyset \, \forall b \in B\}$
  while $\mathrm{Cons}_n \neq \emptyset$ **do**
    $W_{n+1}(x) := \max_{b \in B} \min_{a \in A_n(x,b)} \{\ell(x,a,b) + \gamma V_n^-(S(x,a,b))\}, \forall x \in \mathrm{Cons}_n$
    $\mathrm{Acc}_{n+1} := \mathrm{Acc}_n \cup \mathrm{argmin} W_{n+1}$
    $V_{n+1}^-(x) := W_{n+1}(x), \forall x \in \mathrm{argmin} W_{n+1}$
    $V_{n+1}^-(x) := V_n^-(x), \forall x \in \mathrm{Acc}_n$
    $n \leftarrow n+1$
  end while

The notations introduced in the algorithm have the following meaning

- Acc_n is the set of nodes accepted at the step n, at such nodes the approximate value is not re-computed in the next steps;
- $A_n(x,b) \subseteq A$ is the set of controls that take the state x to Acc_n if player 2 uses the control b;
- Cons_n is the set of nodes considered at the step n, they are the nodes from which player 1 can reach Acc_n no matter what player 2 does;
- $x \in \mathrm{argmin} W_{n+1}$ if $W_{n+1}(x) = \min_{\mathcal{X}} W_{n+1}$, such nodes become accepted at step $n+1$.

Note that Acc_n is strictly increasing as long as $\mathrm{Cons}_n \neq \emptyset$, so the algorithm terminates in a finite number N of steps, at most the cardinality of $\mathcal{X} \setminus \mathcal{X}_f$, which we denote with $|\mathcal{X} \setminus \mathcal{X}_f|$. Denote also with $\mathcal{R}$ the set of nodes from which player 1 can reach the terminal set for any behavior of player 2, i.e.,

$$\mathcal{R} := \{x \in \mathcal{X} : \inf_{\alpha \in \mathcal{A}} \sup_{b_\bullet \in B^{\mathbb{N}}} \hat{n}(x, \alpha[b_\bullet], b_\bullet) < +\infty\}.$$

It is easy to see that if N is the terminal step of the algorithm, i.e., $\mathrm{Cons}_N = \emptyset$ or $\mathrm{Acc}_N = \mathcal{X}$, then

$$\mathrm{Acc}_N = \mathcal{R},$$

i.e., the algorithm identifies the set $\mathcal{R}$ reachable by the the first player.

The main result of this section states that the algorithm indeed computes the value function. It requires the following additional assumption in the discounted case $\gamma < 1$ (see Remark 3 below for a discussion about this condition).

Condition 1 If $\gamma < 1$

$$L + \gamma g_1 \leq \frac{\ell_0}{1-\gamma}.$$

Proposition 3 *Assume either $\gamma = 1$, or $\gamma < 1$ and Condition 1. Then, for any $n \leq N$,*

$$V_n^-(x) = V^-(x), \quad \textit{for all } x \in \mathrm{Acc}_n,$$

and the algorithm converges in $N \leq |\mathcal{X} \setminus \mathcal{X}_f|$ steps to the value function V^- on the reachability set $\mathcal{R}$.

Proof Observe that for $n = 0$ the conclusion holds by definition. It suffices to prove that $V_1^-(x) = V^-(x)$ for $x \in \mathrm{Acc}_1$ since by Proposition 2, if we know V^- on $\tilde{\mathcal{X}} = \mathrm{Acc}_1$, then we can obtain V^- as the value of the new problem with $\mathcal{X}_f$ replaced by $\tilde{\mathcal{X}}$ and g by $V^-|_{\tilde{\mathcal{X}}}$ and thus conclude by induction.

Observe first that $V_1^-(x) \geq V^-(x)$ follows easily from the definitions. Now for

$$\bar{x} \in \mathrm{argmin}_{x \in \mathrm{Cons}_1} W_1(x)$$

consider an optimal pair $(\alpha^*, b_\bullet^*) \in \mathcal{A} \times B^{\mathbb{N}}$ and the corresponding optimal trajectory x_n starting from $\bar{x}$, that is,

$$x_{n+1} = S(x_n, \alpha^*[b_\bullet^*]_n, b_n^*), \quad x_0 = \bar{x},$$

$$V^-(\bar{x}) = J(\bar{x}, \alpha^*[b^*_\bullet], b^*_\bullet).$$

If $\hat{n}(\bar{x}, \alpha^*[b^*_\bullet], b^*_\bullet) = 1$ then $V^-(\bar{x}) = W_1(\bar{x}) = V_1^-(\bar{x})$, which is the desired conclusion. If, instead, $\hat{n} := \hat{n}(\bar{x}, \alpha^*[b^*_\bullet], b^*_\bullet) > 1$ we will distinguish two cases.

- Case $\gamma = 1$. From (4) and $\ell > 0$ we have that

$$V^-(\bar{x}) = \sum_{n=0}^{\hat{n}-2} \ell(x_n, \alpha^*[b_\bullet]_n, b^*_n) + V^-(x_{\hat{n}-1}) > V^-(x_{\hat{n}-1}).$$

  On the other hand, we have an optimal pair strategy-control and corresponding optimal trajectory starting from $x_{\hat{n}-1}$ that reaches $\mathcal{X}_f$ in one step. Then $V^-(x_{\hat{n}-1}) = W_1(x_{\hat{n}-1})$ and so

$$V^-(x_{\hat{n}-1}) = W_1(x_{\hat{n}-1}) \geq W_1(\bar{x}) = V_1^-(\bar{x}) \geq V^-(\bar{x})$$

  which is a contradiction.
- Case $\gamma < 1$.
  Here (4) gives

$$\begin{aligned} V^-(\bar{x}) &= \sum_{n=0}^{\hat{n}-2} \ell(x_n, \alpha^*[b_\bullet]_n, b^*_n)\gamma^n + \gamma^{\hat{n}-1} V^-(x_{\hat{n}-1}) \\ &= \sum_{n=0}^{\hat{n}-2} \ell(x_n, \alpha^*[b_\bullet]_n, b^*_n)\gamma^n + \gamma^{\hat{n}-1} W_1(x_{\hat{n}-1}) \\ &\geq \sum_{n=0}^{\hat{n}-2} \ell(x_n, \alpha^*[b_\bullet]_n, b^*_n)\gamma^n + \gamma^{\hat{n}-1} W_1(\bar{x}) \\ &\geq \sum_{n=0}^{\hat{n}-2} \ell(x_n, \alpha^*[b_\bullet]_n, b^*_n)\gamma^n + \gamma^{\hat{n}-1} V^-(\bar{x}). \end{aligned}$$

  Then,

$$V^-(\bar{x})(1-\gamma^{\hat{n}-1}) \geq \ell_0 \frac{1-\gamma^{\hat{n}-1}}{1-\gamma} \implies V^-(\bar{x}) \geq \frac{\ell_0}{1-\gamma}.$$

  On the other hand, since

$$V_1^-(\bar{x}) \leq L + \gamma g_1$$

  we get the inequality $V_1^-(\bar{x}) \leq V^-(\bar{x})$ by Condition 1.

2.4 Remarks and some open problems

Remark 1 The main advantage of our algorithm is that at each step the approximate value function is updated only on some nodes, namely on Cons_n. Moreover, at least one of these nodes becomes accepted and will not be considered in the next steps. In particular, if the costs ℓ and g are constant (generalized pursuit-evasion games), then W_{n+1} is constant on Cons_n and all considered nodes are accepted. In other words, the value function is computed only once on each node, which considerably speeds up the algorithm in comparison to iterative methods. Algorithms with this property are often called *single-pass*.

Remark 2 If we stop the algorithm before it terminates, say at $\tilde{n} < N$, Proposition 3 says that we have anyway computed the value function in the set $\mathrm{Acc}_{\tilde{n}}$, which may be enough for some practical problems with very large grids.

Remark 3 Condition 1 requires that the oscillation of the running cost ℓ and the size of the terminal cost g are not too high compared with $1/(1-\gamma)$. It essentially says that, if player 1 is on a node where he can reach $\mathcal{X}_f$, it is convenient for him to do so even if the cost of this step is high, rather than following forever a trajectory with cheap running costs. For any ℓ and g verifying (2) and (3) the condition is satisfied for γ sufficiently close to 1. It is satisfied also for all $\gamma \in (0,1)$ in discounted pursuit-evasion games, where $\ell \equiv 1$ and $g \equiv 0$.

Remark 4 If $\gamma = 1$, we can add a final step to the algorithm by setting $V^-_{N+1}(x) := W_0(x) = +\infty$ for all $x \in \mathcal{X} \setminus \mathrm{Acc}_N$, so $V^-_{N+1}(x) = V^-(x)$ for $x \in \mathcal{X} \setminus \mathcal{R}$ and we have convergence on the whole state space $\mathcal{X}$. On the other hand, if $\gamma < 1$ the algorithm gives no information on the value V^- outside $\mathcal{R}$.

Remark 5 If $\gamma = 1$, $\ell \equiv 1$, and $g \equiv 0$, the problem for player 1 is finding the path of shortest length that reaches $\mathcal{X}_f$, whereas player 2 seeks to make such length the longest possible (generalized pursuit-evasion). If, in addition, there is no player 2, i.e., B is a singleton, the problem reduces to the shortest path and the algorithm of this section is the classical Dijkstra algorithm.

In reachability games there is no running cost and all the states in $\mathcal{X}_f$ have the same cost. Then the algorithm is essentially the same as Algorithm 2 in Alfaro et al. (2007), where the set Acc_{n+1} is updated by

$$\mathrm{Acc}_{n+1} := \mathrm{Acc}_n \cup \mathrm{Cons}_n.$$

Then the reachability set $\mathcal{R}$ is computed exactly as in the above generalized pursuit-evasion game, although here the length of the path is not of interest. When the moves are alternating, this algorithm runs in a linear time with respect to the size of the game, defined as[1]

$$\|\mathcal{G}\| := |X| + |A| + |B|.$$

Remark 6 All the results of this section can be adapted to the *upper value function*

$$V^+(x) := \sup_{\beta \in \mathcal{B}} \inf_{a_\bullet \in A^{\mathbb{N}}} J(x, a_\bullet, \beta[a_\bullet]),$$

where $\mathcal{B}$ is the set of non-anticipating strategies for the second player, defined in the obvious symmetric way. Although in general $V^-(x) < V^+(x)$, we will see in the next Section that the informational advantage disappears in the continuum limit of discrete approximations of differential games, under suitable assumptions, see Remark 12.

Remark 7 It would be interesting to study the model of Section 2.1 in the case of simultaneous moves, where players select their actions randomly. This falls within the theory of stochastic games initiated by Shapley (1953) (for more recent accounts, see the book by Filar and Vrieze (1996) or Chapter 5 in Sorin (2002) , although the transition map S is deterministic here. Even in the case of deterministic transitions, there are algebraic obstructions that make the extension of our approach difficult, see for instance the first example provided by Parthasarathy and Raghavan (1981). In their example, the data of the problem consist of rational numbers but the value is irrational. The authors suggest that any algorithm that terminates in a finite number of steps, such as ours, would require that the data and the value of the game remain in the same ordered field. See, for instance, Filar and Vrieze (1996) for several classes of games with this so-called ordered field property.

Remark 8 The adaptation of the algorithm to the case of stochastic transitions presents several additional difficulties. For a single player the Dijkstra algorithm was extended to the stochastic case in Bertsekas (2001) if there exists a consistently improving optimal policy. A deeper study of the causality properties needed in stochastic shortest path problems with one controller is in Vladimirsky (2008). As hinted in the preceding remark, the case of zero-sum two-person stochastic games appears completely open.

3 Discretisation of differential games

Consider a continuous-time dynamical system controlled by two players

$$y'(t) = f(y(t), a(t), b(t)), \quad y(0) = x, \tag{7}$$

where $f : \mathbb{R}^d \times A \times B \to \mathbb{R}$ is Lipschitz with respect y, and A, B are given finite sets, as in the previous section. The admissible controls for the players are

$$\tilde{\mathcal{A}} := \{a : [0, +\infty) \to A, \text{ measurable}\}, \quad \tilde{\mathcal{B}} := \{b : [0, +\infty) \to B, \text{ measurable}\}.$$

[1] This definition is slightly different than the one in Alfaro et al (2007), since there only the actions available at each state are considered. Without loss of generality, we assume that all actions are available in all states.

We are given a closed target $\mathcal{T} \subseteq \mathbb{R}^n$, and define the first time the trajectory hits $\mathcal{T}$ as

$$t_x(a,b) := \inf\{t : y_x(t;a,b) \in \mathcal{T}\},$$

where $y_x(\cdot;a,b)$ is the solution of (7) corresponding to $a \in \tilde{\mathcal{A}}$ and $b \in \tilde{\mathcal{B}}$, and $t_x(a.b) := +\infty$ if the target is never reached. Consider also the cost functional

$$\tilde{J}(x,a,b) := \int_0^{t_x} l(y(t),a(t),b(t))e^{-\lambda t}dt + e^{-\lambda t_x} g(y_x(t_x;a,b)), \quad t_x = t_x(a,b), \quad a \in \tilde{\mathcal{A}},\ b \in \tilde{\mathcal{B}};$$

where $\lambda \geq 0$ is the discount rate, and the running and terminal costs $l : \mathbb{R}^d \times A \times B \to \mathbb{R}$ and $g : \mathcal{T} \to \mathbb{R}$ are continuous and bounded, with $l > 0$.

In differential games *non-anticipating strategies* for the first player are maps $\alpha : \tilde{\mathcal{B}} \to \tilde{\mathcal{A}}$ such that, for all $t > 0$, $b(s) = \tilde{b}(s)$ for all $s \leq t$ implies $\alpha[b](s) = \alpha[\tilde{b}](s)$ for all $s \leq t$. We denote with Γ the set of such strategies. The lower value function of the game, in the sense of Varaiya, Roxin, Elliott and Kalton, is defined as

$$v^-(x) := \inf_{\alpha \in \Gamma} \sup_{b \in \tilde{\mathcal{B}}} \tilde{J}(x, \alpha[b], b).$$

It is well known that under suitable conditions v^- coincides with the upper value function and it is therefore the value of the game, see Remark 12 for more details. It is also known (see, e.g., Chapter VIII in Bardi and Capuzzo-Dolcetta 1997) that it is a viscosity solution of the (lower) Hamilton-Jacobi-Isaacs equation

$$\lambda v^- - \max_{b \in B} \min_{a \in A} \left\{ f(x,a,b) \cdot Dv^- + l(x,a,b) \right\} = 0 \ \text{ in } \Omega := \mathbb{R}^d \setminus \mathcal{T}. \tag{8}$$

Now take a finite grid $\mathcal{X}$ and set $\mathcal{X}_f := \mathcal{X} \cap \mathcal{T}$, i.e., the final nodes are those in the target. Next fix $h > 0$ and set

$$S(x,a,b) = x + hf(x,a,b), \qquad \ell(x,a,b) = hl(x,a,b), \qquad \gamma = e^{-\lambda h}.$$

Then the dynamic game of Section 2.1 is a natural discretisation of the differential game, with the control system replaced by its Euler scheme with step h and the integral cost functional replaced by a Riemann sum. This discretisation is the basis of some semi-Lagrangian approximation schemes for the Hamilton-Jacobi-Isaacs equation, see, e.g., Bardi, Falcone and Soravia (1999) and Falcone and Ferretti (2014). In such schemes one solves the finite difference equation

$$W(x) = \max_{b \in B} \min_{a \in A} \left\{ \ell(x,a,b) + \gamma W(S(x,a,b)) \right\}, \quad \forall\, x \in \mathcal{X} \setminus \mathcal{X}_f, \tag{9}$$

with the boundary condition

$$W(x) = g(x), \qquad \forall\, x \in \mathcal{X}_f. \tag{10}$$

Note, however, that in general $S(x,a,b)$ might not be in $\mathcal{X}$, so in the right hand side W is extended by interpolation among the neighbouring nodes. The main assumption of this section is that *the grid is adapted to the dynamics* (with time step h), that is,

$$S(x,a,b) \in \mathcal{X} \quad \forall\, x \in \mathcal{X} \setminus \mathcal{X}_f,\ a \in A,\ b \in B. \tag{11}$$

Then W can be computed only on the nodes, without any interpolation procedure.

Proposition 4 *Under the assumption* (11) *the solution W of the discrete Isaacs equation* (9) *coincides with the lower value function V^- defined in Section 2.2. Thus it can be computed by the algorithm of Section 2.3.*

Proof Under the assumption (11) the discrete Isaacs equation (9) coincides with the one-step Dynamic Programming Principle (5) satisfied by V^-. On the other hand, from (5) one gets the general Dynamic Programming Principle (4) by induction, and therefore the equality $W = V^-$.

Note that, given a finite set $\mathcal{X}_f$, the dynamics f, and h, a grid adapted to the dynamics can be easily constructed, at least in principle, by setting

$$\mathcal{X}_0 := \mathcal{X}_f, \qquad \mathcal{X}_{n+1} := \{x \in \mathbb{R}^d : S(x,a,b) \in \mathcal{X}_n, \text{ for some } a \in A,\ b \in B\}$$

and iterating a finite number of times, where the set $\mathcal{X}_{n+1}$ is truncated to a finite set in case it is not finite.

Example 1 Assume that $f = f(a,b)$ does not depend on x. Then the previous construction of a grid adapted to the dynamics becomes explicit

$$\mathcal{X}_0 := \mathcal{X}_f, \qquad \mathcal{X}_{n+1} := \{x - hf(a,b) \text{ for some } x \in \mathcal{X}_n,\ a \in A,\ b \in B\},$$

and after a finite number of iterations it produces a finite grid $\mathcal{X}$, since $\mathcal{X}_f, A$ and B are finite.

Example 2 Consider the *convex-concave eikonal equation* in $\mathbb{R}^2$ with Dirichlet boundary conditions

$$|u_x| - \delta|u_y| = l(x,y) \ \text{ in } \Omega, \qquad u(x,y) = g(x,y) \ \text{ on } \partial\Omega,$$

with $\delta, l > 0$ and Ω a rectangle, say $\Omega = (0,c) \times (0,d)$ (see Bardi and Terrone 2013 for other results on such equation). An associated differential game can be chosen with dynamics

$$x' = a,\ y' = b, \quad a \in \{-1,0,1\},\ b \in \{-\delta,0,\delta\}.$$

A rectangular grid $\mathcal{X} = \{(jh, k\delta h) : j = 1,\dots,\frac{c}{h},\ k = 1,\dots,\frac{d}{\delta h}\}$ is adapted to the dynamics if $\frac{c}{h}, \frac{d}{\delta h} \in \mathbb{N}$, and this occurs for a suitable sequence of steps h going to 0 if $c\delta/d = m/n$ for some $m,n \in \mathbb{N}$. In fact we can take $h = \frac{d}{\delta n K}$ for any $K \in \mathbb{N}$, because $\frac{c}{h} = mK$ and $\frac{d}{\delta h} = nK$.

The locally uniform convergence of the solution W of the discrete Isaacs equation (9) to the lower value function v^- as h and the mesh size k of the grid $\mathcal{X}$ tend to zero was proved in Bardi, Falcone and Soravia (1994) in the case v^- is continuous, $\mathcal{T}$ has a Lipschitz boundary, and $k/h \to 0$, see also Cristiani and Falcone (2009), the survey by Bardi, Falcone, and Soravia (1999), and the recent book by Falcone and Ferretti (2014). In the case of discontinuous v^- the convergence is weaker and the precise statements are more technical, see Bardi, Bottacin, and Falcone (1995). For grids adapted to the dynamics it is unlikely that $k/h \to 0$ because the space and time step are of the same order, as it can be seen in the preceding examples. However, by taking advantage of the property (11) we can remove such restriction, as we show now.

We consider a sequence $h_n \to 0$ and sequences $\mathcal{X}^n$, $\mathcal{X}^n_f = \mathcal{X}^n \cap \mathcal{T}$, such that $\mathcal{X}^n$ is a grid adapted to the dynamics with time step h_n and

$$\forall\, x \in \mathbb{R}^d \quad \exists\, x^{(n)} \in \mathcal{X}^n \text{ such that } \lim_n x^{(n)} = x. \tag{12}$$

We call this an *admissible sequence of grids*. For each pair $h_n, \mathcal{X}^n$ we solve the discrete Isaacs equation (9) with boundary conditions (10) and call W_n its solution. Then we define for $x \in \overline{\Omega}$ the viscosity semi-limits

$$\overline{W}(x) := \sup\left\{\limsup_n W_n(x^{(n)}) : \mathcal{X}^n \ni x^{(n)} \to x\right\}, \quad \underline{W}(x) := \inf\left\{\liminf_n W_n(x^{(n)}) : \mathcal{X}^n \ni x^{(n)} \to x\right\}. \tag{13}$$

Note that they are finite at x if W_n are equibounded in a neighborhood of x.

The following lemma about the semi-limits is a straightforward extension of a standard result in viscosity theory, see Lemma V.1.6 in Bardi and Capuzzo-Dolcetta (1997).

Lemma 1 *Assume $W_n : \mathcal{X}^n \to \mathbb{R}$ are locally equibounded, $\mathcal{X}^n$ satisfies* (12)*, $\phi \in C(\overline{\Omega})$, and $\overline{W} - \phi$ attains at y a strict maximum relative to the ball $\mathbb{B}$ centered at y with radius $r > 0$. Choose $x_n \in$* $\operatorname{argmax}_{\mathcal{X}^N \cap \mathbb{B}}(W_n - \phi)$*. Then there is a subsequence $x_{n_k} \to y$ and such that $W_{n_k}(x_{n_k}) \to \overline{W}(y)$ as $k \to \infty$.*

The next proposition is the main result of this section.

Proposition 5 *Assume $h_n \to 0$, $\mathcal{X}^n$ is an associated admissible sequence of grids, W_n solves* (9) *on $\mathcal{X} = \mathcal{X}^n$, and the sequence W_n is locally equibounded. Then $\overline{W}$ and $\underline{W}$ are, respectively, a viscosity sub- and supersolution of the Isaacs equation* (8)*.*

Proof We prove that $\overline{W}$ is a subsolution, the statement about $\underline{W}$ is obtained in a similar way. Take $\phi \in C^1(\overline{\Omega})$ such that $\overline{W} - \phi$ attains at $y \in \Omega$ a strict maximum relative to the ball $\mathbb{B}$ centered at y with radius $r > 0$. We must prove that

$$\lambda\overline{W}(y) - \max_{b\in B}\min_{a\in A}\{D\phi(y)\cdot f(y,a,b)) + l(y,a,b)\} \le 0. \tag{14}$$

Consider the subsequence of maximum points of $W_n - \phi$ produced in Lemma 1 and relabel it so that $x_n \to y$ and $W_n(x_n) \to \overline{W}(y)$. Since $x_n \in \mathcal{X}^n \setminus \mathcal{X}^n_f$, the discrete Isaacs equation (9) gives

$$W_n(x_n) = \max_{b\in B}\min_{a\in A}\left\{h_n l(x_n,a,b) + \gamma_n W_n(x_n + h_n f(x_n,a,b))\right\}, \quad \gamma_n := e^{-\lambda h_n}.$$

Then for all n there exists $b_n \in B$ such that

$$\gamma_n\left[W_n(x_n) - W_n(x_n + h_n f(x_n,a,b_n))\right] \le h_n l(x_n,a,b_n) + (\gamma_n - 1)W_n(x_n) \quad \forall a \in A.$$

Since x_n is a local maximum point of $W_n - \phi$ we have, for h_n small enough,

$$\phi(x_n) - \phi(x_n + h_n f(x_n,a,b_n)) \le W_n(x_n) - W_n(x_n + h_n f(x_n,a,b_n)).$$

Combining the last two inequalities gives

$$\min_{b\in B}\max_{a\in A}\left\{\gamma_n \frac{\phi(x_n) - \phi(x_n + h_n f(x_n,a,b))}{h_n} - l(x_n,a,b)\right\} \le \frac{\gamma_n - 1}{h_n} W_n(x_n)$$

and letting $n \to \infty$ we get

$$\min_{b\in B}\max_{a\in A}\left\{-D\phi(y)\cdot f(y,a,b)) - l(y,a,b)\right\} \le -\lambda\overline{W}(y),$$

which implies the desired inequality (14).

Now we can get the convergence of the scheme under suitable assumptions.

Corollary 1 *Assume $\lambda > 0$, $\partial\mathcal{T}$ is a Lipschitz hypersurface, f is bounded on $\partial\mathcal{T} \times A \times B$, $h_n \to 0$, $\mathcal{X}^n$ is an associated admissible sequence of grids, and the lower value $v^- \in C(\overline{\Omega})$. Then the semilimits* (13) *of the solutions W_n of* (9) *and* (10) *satisfy*

$$\overline{W}(x) = \underline{W}(x) = v^-(x) \qquad \forall\, x \in \overline{\Omega}. \tag{15}$$

Proof It is easy to see that for $\lambda > 0$ the sequence W_n is uniformly bounded by $\sup_{\mathbb{R}^d\times A\times B}|l|/\lambda + \sup_{\mathbb{R}^d} g$. The rest of the proof is similar to that of Theorem 1 in Bardi, Falcone and Soravia (1994) or Theorem 2.3 in Bardi, Falcone and Soravia (1999). In addition to Proposition 5 we use the following boundary conditions in the viscosity sense:

$$\overline{W} \le g \quad \text{or} \quad \lambda\overline{W} - \max_{b\in B}\min_{a\in A}\left\{f(x,a,b)\cdot\overline{W} + l(x,a,b)\right\} \le 0 \quad \text{on } \partial\Omega, \tag{16}$$

$$\underline{W} \ge g \quad \text{or} \quad \lambda\underline{W} - \max_{b\in B}\min_{a\in A}\left\{f(x,a,b)\cdot\underline{W} + l(x,a,b)\right\} \ge 0 \quad \text{on } \partial\Omega.$$

To prove, for instance, (16), we proceed as in Proposition 5. We take $\phi \in C^1(\overline{\Omega})$ such that $\overline{W} - \phi$ attains a strict local maximum at $y \in \partial\Omega$. By extracting a subsequence from the sequence of maximum points of $W_n - \phi$ produced in Lemma 1 we can assume that either $x_n \in \mathcal{X}^n \setminus \mathcal{X}^n_f$ for all n or $x_n \in \mathcal{X}^n_f$ or all n. In the former case the argument of Proposition 5 gives the inequality (14). In the latter case $g(x_n) = W_n(x_n) \to \overline{W}(y)$, and the continuity of g implies $\overline{W}(y) = g(y)$.

Next we use a comparison principle between a continuous semi-solution of (8) attaining the boundary data g and a semicontinuous semi-solution of (8) with boundary condition in viscosity sense. It can be found in Bardi and Capuzzo-Dolcetta (1997) for Ω bounded and in Bardi, Falcone and Soravia (1994) for Ω unbounded. Here we need the Lipschitz regularity of $\partial\Omega$, the boundedness of f on $\partial\Omega$, and the fact that, if $v^- \in C(\overline{\Omega})$, it solves (8) and $v^- = g$ on $\partial\Omega$. Then we get the inequalities

$$\overline{W} \le v^- \le \underline{W} \qquad \text{in } \overline{\Omega}.$$

Then $\overline{W} = v^- = \underline{W}$.

Remark 9 The equality (15) implies the following form of *uniform convergence* of W_n to v^-:
for all $\epsilon > 0$ and compact set K there exists $\overline{n}$ and $\delta > 0$ such that

$$|W_n(x^{(n)}) - v^-(x)| < \epsilon \quad \forall\, x \in K,\ x^{(n)} \in \mathcal{X}^n,\ n \ge \overline{n},\ |x^{(n)} - x| < \delta,$$

as it is easy to prove by contradiction.

Remark 10 Proposition 5 and Corollary 1 appear to be new even in the case of a single player, although the convergence of FMM schemes for Hamilton-Jacobi equations with convex Hamiltonian has been studied, e.g., in (Sethian and Vladimirsky 2003 and Cristiani 2009).

Remark 11 The case of discontinuous v^-, which may occur in pursuit-evasion games, could also be treated in the framework of Bardi, Bottacin, and Falcone (1995).

Remark 12 All the results of this section can be adapted to the upper value of the differential game

$$v^+(x) := \sup_{\beta\in\Delta}\inf_{a\in\tilde{\mathcal{A}}} \tilde{J}(x,a,\beta[a]),$$

where Δ is the set of non-anticipating strategies for the second player. It is known that v^+ satisfies the upper Hamilton-Jacobi-Isaacs equation

$$\lambda v^+ - \min_{a\in A}\max_{b\in B}\left\{f(x,a,b)\cdot Dv^+ + l(x,a,b)\right\} = 0 \ \text{ in } \Omega := \mathbb{R}^d \setminus \mathcal{T}.$$

If the Isaacs condition holds, namely,

$$\min_{a\in A}\max_{b\in B}\{f(x,a,b)\cdot p + l(x,a,b)\} = \max_{b\in B}\min_{a\in A}\{f(x,a,b)\cdot p + l(x,a,b)\} \qquad \forall\, p\in\mathbb{R}^d,\ x\in\overline{\Omega},$$

the upper H-J-I equation coincides with (8). By the uniqueness of viscosity solutions to the Dirichlet problem, if either v^- or v^+ is known to be continuous, we get that $v^- = v^+$ and the differential game has a value. In this case the informational advantage of player 1 in the discrete game of Section 2 vanishes in the continuum limit.

References

1. Alfaro T, Henzinger T, and Kupferman O, (2007) Concurrent reachability games, Theoretical Computer Science 386:188–217.
2. Bardi M and Capuzzo-Dolcetta I (1997) Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations Birkhäuser Boston Inc.
3. Bardi M, Bottacin S, Falcone M (1995) Convergence of discrete schemes for discontinuous value functions of pursuit-evasion games. New trends in dynamic games and applications, 273–304, Ann. Internat. Soc. Dynam. Games, 3, Birkhäuser Boston.
4. Bardi M, Falcone M and Soravia P (1994) Fully discrete schemes for the value function of pursuit-evasion games. Advances in dynamic games and applications, pp 89–105, Ann. Internat. Soc. Dynam. Games, 1, Birkhäuser Boston.
5. Bardi M, Falcone M. and Soravia P (1999) Numerical methods for pursuit-evasion games via viscosity solutions, in "Stochastic and differential games: theory and numerical methods", M. Bardi, T. Parthasarathy e T.E.S. Raghavan eds., pp. 105-175, Ann. Internat. Soc. Dynam. Games, 4, Birkhäuser, Boston.
6. Bardi M and Soravia P (1991) Approximation of differential games of pursuit-evasion by discrete-time games. Differential games–developments in modelling and computation, Lecture Notes in Control and Inform. Sci., 156, Springer, pp 131–143.
7. Bardi M and Soravia P (1994) A comparison result for Hamilton-Jacobi equations and applications to some differential games lacking controllability. Funkcial. Ekvac. 37:19–43.
8. Bardi M and Terrone G (2013) On the Homogenization of some Non-coercive Hamilton-Jacobi-Isaacs Equations, Commun. Pure Appl. Anal. 12:207–236.
9. Bertsekas D (2001) Dynamic Programming and Optimal Control, 2nd ed., Vol. II. Athena Scientific, Boston.
10. Cacace S, Cristiani E and Falcone M (2014) Can local single-pass methods solve any stationary Hamilton-Jacobi equation?, SIAM J. Sci. Comput. 36:A570–A587.
11. Cacace S, Cristiani E, Falcone M and Picarelli A (2012), A patchy dynamic programming scheme for a class of Hamilton-Jacobi-Bellman equations. SIAM J. Sci. Comput. 34:A2625–A2649.
12. Cardaliaguet P, Quincampoix M and Saint-Pierre P (1999) Set-valued numerical analysis for optimal control and differential games, in "Stochastic and differential games: theory and numerical methods", M. Bardi, T. Parthasarathy e T.E.S. Raghavan eds. , Ann. Internat. Soc. Dynam. Games, 4:177–247, Birkhäuser Boston.
13. Cristiani E (2009) A fast marching method for Hamilton-Jacobi equations modeling monotone front propagations, J. Sci. Comput. 39:189–205.
14. Cristiani E and Falcone M (2006) A Fast Marching Method for Pursuit-Evasion Games. Communications to SIMAI Congress, Vol. 1, DOI: 10.1685/CSC06059
15. Cristiani E and Falcone M (2009) Fully-discrete schemes for the value function of pursuit-evasion games with state constraints. Advances in dynamic games and their applications, Ann. Internat. Soc. Dynam. Games, 10:177–206, Birkhäuser Boston.
16. Dijkstra E W (1959) *A note on two problems in connection with graphs*, Numer. Math., 1:269–271.
17. Falcone M and Ferretti R (2014) Semi-Lagrangian approximations schemes for linear and Hamilton-Jacobi equations, SIAM, Philadelphia.

18. Filar J A and Raghavan T E S (1991) Algorithms for stochastic games: A survey, Zeischrift fuer Operations Research **35** 6:437–472.
19. Filar J A and Vrieze O J (1996) Competitive Markov Decision Processes, Springer-Verlag, Berlin.
20. Fredman M L and Tarjan R E (1987), Fibonacci heaps and their uses in improved network optimization algorithms, J. Assoc. Comput. Mach. 34.
21. Grüne L and Junge O (2008) *Global optimal control of perturbed systems.* J. Optim. Theory Appl. 136 3:411–429.
22. Kushner H J (2004), The Gauss-Seidel numerical procedure for Markov stochastic games, IEEE Transactions on Automatic Control **49** 10:1779 – 1784.
23. McEneaney W M (2006) Max-plus methods for nonlinear control and estimation. Birkhäuser Boston, Inc., Boston.
24. Parthasarathy T and Raghavan T E S (1981) An orderfield property for stochastic games when one player controls transition probabilities, Journal of Optimization Theory and Applications, 33:375-392.
25. Raghavan T E S and Syed Z (2003) A policy-improvement type algorithm for solving zero-sum two-person stochastic games of perfect information. Math. Program. 95 Ser. A, 3:513–532.
26. Sethian J A (1996) A fast marching level set method for monotonically advancing fronts, Proc. Nat. Acad. Sci. U.S.A. 93:1591–1595.
27. Sethian J A (1999) Level set methods and fast marching methods. Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Second edition. Cambridge University Press.
28. Sethian J A and Vladimirsky A (2003) Ordered upwind methods for static Hamilton-Jacobi equations: theory and algorithms, SIAM J. Numer. Anal. 41 1:325–363.
29. Shapley L S (1953) Stochastic games, Proc. Nat. Acad. Sci. U. S. A. 39:1095–1100.
30. Sorin S (2002) A first course on zero-sum repeated games, Mathématiques & Applications vol. 37, Springer-Verlag.
31. Tsitsiklis J N (1995) Efficient algorithms for globally optimal trajectories, IEEE Transactions on Automatic Control 40:1528–1538.
32. Vladimirsky A (2008) Label-setting methods for multimode stochastic shortest path problems on graphs, Math. Oper. Res. 33 4:821–838.
33. von Lossow M (2007) A min-max version of Dijkstra's algorithm with application to perturbed optimal control problems, in Proceedings of the 6th International Congress on Industrial and Applied Mathematics (ICIAM07), Proc. Applied Math. Mechanics 7:4130027–4130028.