

Global Optimal Control of Perturbed Systems

L. Grüne · O. Junge

Published online: 16 November 2007
© Springer Science+Business Media, LLC 2007

Abstract We propose a new numerical method for the computation of the optimal value function of perturbed control systems and associated globally stabilizing optimal feedback controllers. The method is based on a set-oriented discretization of the state space in combination with a new algorithm for the computation of shortest paths in weighted directed hypergraphs. Using the concept of multivalued game, we prove the convergence of the scheme as the discretization parameter goes to zero.

Keywords Optimal control · Dynamic games · Set-oriented numerics · Graph theory

1 Introduction

Global infinite-horizon optimal control methods for the solution of general nonlinear stabilization problems are attractive for their flexibility and theoretical properties, because they are applicable to virtually all types of nonlinear dynamical systems, their optimal value functions can typically be identified as Lyapunov functions and they allow for a rigorous treatment of perturbations in a game theoretical setting. However, these methods have the drawback that their numerical solution requires the discretization of the state space which results in huge numerical problems both in terms of computational cost and in terms of memory requirements. Hence, in order to make these methods applicable to a broader range of systems, advanced numerical techniques are needed in order to reduce the computational effort as much as possible.

Communicated by H.J. Pesch.

L. Grüne
Mathematisches Institut, Universität Bayreuth, Bayreuth, Germany

O. Junge (✉)
Zentrum Mathematik, Technische Universität München, Garching, Germany
e-mail: junge@ma.tum.de

A novel approach to such problems was presented in the recent paper [1], where a set oriented numerical method for the approximate computation of the optimal value function of certain nonlinear optimal control problems has been developed. The approach relies on a division of state space into boxes that constitute the nodes of a directed weighted graph on which standard shortest path algorithms can directly be applied, yielding an approximate value function which is piecewise constant on the state space.

In [2], it was subsequently shown that the approximate optimal value function can in fact be used in order to construct a stabilizing feedback controller. Based on concepts from dynamic programming [3] and Lyapunov based approximate stability analysis [4], a statement about its optimality properties was given and a local a posteriori error estimate derived that enables an adaptive construction of the division of state space. However, due to the fact that the approximate optimal value function is not continuous, the constructed feedback law is in general not robust with respect to perturbations of the system.

In the present paper, we show how to incorporate arbitrary perturbations into the framework sketched above. These perturbations can be either inherently contained in the underlying model, describing, e.g., external disturbances or the effect of unmodelled dynamics, or they could be added on top of the original model to account, e.g., for discretization errors.

Our goal in this paper is to construct a feedback law which is robust in the sense that on a certain subset of state space it stabilizes the system regardless on how the perturbation acts. Conceptually, this problem leads to a dynamic game, where the controls and the perturbations are associated to two “players” that try to minimize and to maximize a given cost functional, respectively. We show how the discretization of state space in a natural way leads to a multivalued dynamic game (i.e. a discrete inclusion) and prove convergence of the associated value function when the images of the inclusion shrink to the original single-valued map. From this multivalued game we derive a directed weighted hypergraph that gives a finite state model of the original game. We formulate a minmax-version of Dijkstra’s algorithm in order to compute the associated approximate value function and prove convergence when the box-diameter of the state space division goes to zero.

Compared to other dynamic programming approaches to the stabilization of perturbed nonlinear systems (see, e.g., [5] and the references therein), the main advantages of our method are these general and rigorously provable convergence properties and the low computational cost of our minmax-Dijkstra algorithm, cf. Sect. 6.1. However, our new algorithm is also advantageous for unperturbed problems when treating the spatial discretization errors as perturbation (cf. example (16)).

The paper is organized as follows. In the ensuing Sect. 2 we describe the problem formulation and the associated game theoretic interpretation. In Sect. 3 we introduce the concept of a multivalued game and an enclosure and prove a statement about the convergence of the value function of a sequence of enclosures of a multivalued game. These result are extended to systems with state constraints in Sect. 4. In Sect. 5 we show how via the division of state space one obtains a multivalued game from the original system, construct the corresponding hypergraph and introduce an associated shortest path algorithm. Some hints on its implementation, complexity issues as well

as two numerical examples are addressed in Sect. 6. Convergence of the numerical approximation to the optimal value function and the construction of approximately optimal feedback laws are discussed in Sects. 7 and 8, respectively.

The present paper is a shortened version of the manuscript [6].

2 Problem Formulation

We consider the problem of optimally stabilizing the discrete-time perturbed control system

$$x_{k+1} = f(x_k, u_k, w_k), \quad k = 0, 1, \dots, \tag{1}$$

where $f : X \times U \times W \rightarrow X$ is continuous, $x_k \in X$ is the state of the system, $u_k \in U$ is the control input and $w_k \in W$ is a perturbation parameter, chosen from sets $X \subset \mathbb{R}^d$, $U \subset \mathbb{R}^m$ and $W \subset \mathbb{R}^\ell$. In addition to the evolution law, we are given a continuous cost function $g : X \times U \rightarrow [0, \infty)$, that assigns the cost $g(x_k, u_k)$ to any transition $x_{k+1} = f(x_k, u_k, w_k)$, $w_k \in W$.

Our goal is to derive an (optimal) *feedback law* $u : X \rightarrow U$ that *stabilizes* the system in the sense that, for a certain subset $S \subset X$, any trajectory starting in S tends to some prescribed set $O \subset X$, while the *worst case accumulated cost* is minimized.

Let us be more precise. For a given initial point $x \in X$, a control sequence $\bar{u} = (u_k)_{k \in \mathbb{N}} \in U^{\mathbb{N}}$ and a perturbation sequence $\bar{w} = (w_k)_{k \in \mathbb{N}} \in W^{\mathbb{N}}$ yield the *trajectory* $\bar{x}(x, \bar{u}, \bar{w}) = (x_k(x, \bar{u}, \bar{w}))_{k \in \mathbb{N}}$, defined by $x_0 = x$ and

$$x_{k+1} = f(x_k(x, \bar{u}, \bar{w}), u_k, w_k), \quad k = 0, 1, \dots, \tag{2}$$

while the associated accumulated cost is given by

$$J(x, \bar{u}, \bar{w}) = \sum_{k=0}^{\infty} g(x_k(x, \bar{u}, \bar{w}), u_k).$$

In order to formalize the interplay between the control and the perturbation, we employ a *game theoretic viewpoint* which we describe next. The problem formulation actually already describes a game (see e.g. [7]), where at each step of the iteration (1) two “players” choose a control value u_k and a perturbation value w_k , respectively. The goal of the controlling player is to minimize J , while the perturbing player tries to maximize this quantity.

We assume that the controlling player has to choose the value u_k first and that the perturbing player has the advantage of knowing u_k when choosing the perturbation value w_k . However, the perturbing player is not able to foresee future choices of the controlling one. More formally, we restrict the choice of perturbation sequences $\bar{w} \in W^{\mathbb{N}}$ to those that result from applying a *nonanticipating strategy* $\beta : U^{\mathbb{N}} \rightarrow W^{\mathbb{N}}$ to a given control sequence $\bar{u} \in U^{\mathbb{N}}$, i.e. we have $\bar{w} = \beta(\bar{u})$, with β satisfying

$$u_k = u'_k \quad \forall k \leq K \quad \Rightarrow \quad \beta(\bar{u})_k = \beta(\bar{u}')_k \quad \forall k \leq K,$$

for any two control sequences $\bar{u} = (u_k)_k, \bar{u}' = (u'_k)_k \in U^{\mathbb{N}}$. Let \mathcal{B} denote the set of all nonanticipating strategies $\beta : U^{\mathbb{N}} \rightarrow W^{\mathbb{N}}$.

As mentioned, our goal is to find a feedback law $u : X \rightarrow U$ such that with controls $u_k = u(x_k)$, x_k approaches a given set $O \subset X$, regardless of how the perturbation sequence \bar{w} is chosen. Accordingly, we assume that we know a compact robust forward invariant set $O \subset X$, i.e. for all $x \in O$ there is a control $u \in U$ such that $f(x, u, W) \subset O$. Since we are done with controlling the system once we are on O , we assume that $g(x, u) = 0$ for all $x \in O$ and all $u \in U$ and $g(x, u) > 0$ for all $x \notin O$ and all $u \in U$. Further assumptions on g and on the dynamics in a neighborhood of O will be specified later.

Our construction of the feedback law will be based on the *upper value function* $V : X \rightarrow [0, \infty]$,

$$V(x) = \sup_{\beta \in \mathcal{B}} \inf_{\bar{u} \in U^{\mathbb{N}}} J(x, \bar{u}, \beta(\bar{u})), \tag{3}$$

of the game (1), which fulfills the *optimality principle*

$$V(x) = \inf_{u \in U} \left[g(x, u) + \sup_{w \in W} V(f(x, u, w)) \right]. \tag{4}$$

3 Multivalued Games

As we will see in the next section, our set oriented approach to the discretization of state space of the perturbed control system (1) leads to a finite state multivalued system. For the convergence analysis of this discretization it turns out to be useful to introduce as an intermediate object an infinite state multivalued game defined by a discrete inclusion. This is given by a multivalued map $F : X \times U \times W \rightrightarrows X$, where $X \subset \mathbb{R}^d$ is a closed set and $U \subset \mathbb{R}^m$, $W \subset \mathbb{R}^\ell$ and the images of F are compact sets, together with a cost function $G : X \times X \times U \times W \rightarrow [0, \infty)$.

In order to simplify our presentation we first assume that $F(x, u, w) \neq \emptyset$ for all $x \in X$, $u \in U$, $w \in W$, which will be relaxed later, cf. Sect. 4. Further regularity assumptions on these maps will be imposed when needed. Note that we have introduced a second state argument in G , which allows to associate different costs to the trajectories of the associated discrete inclusion.

For a given initial state $x \in X$, a given control sequence $\bar{u} = (u_k)_{k \in \mathbb{N}} \in U^{\mathbb{N}}$ and a given perturbation sequence $\bar{w} = (w_k)_{k \in \mathbb{N}} \in W^{\mathbb{N}}$, a *trajectory* of the game is given by any sequence $\bar{x} = (x_k)_{k \in \mathbb{N}} \in X^{\mathbb{N}}$ such that $x_0 = x$ and $x_{k+1} \in F(x_k, u_k, w_k)$, $k = 0, 1, 2, \dots$. We denote by

$$\mathcal{X}_F(x, \bar{u}, \bar{w}) = \{ (x_k)_{k \in \mathbb{N}} \in X^{\mathbb{N}} \mid x_0 = x, x_{k+1} \in F(x_k, u_k, w_k) \ \forall k \in \mathbb{N} \}$$

the set of all trajectories of F associated to x , \bar{u} and \bar{w} . The *accumulated cost* is given by

$$J_{(F,G)}(x, \bar{u}, \bar{w}) = \inf_{(x_k)_{k \in \mathbb{N}} \in \mathcal{X}_F(x, \bar{u}, \bar{w})} \sum_{k=0}^{\infty} G(x_k, x_{k+1}, u_k, w_k).$$

As in the previous section, we are interested in computing the upper value function

$$V_{(F,G)}(x) = \sup_{\beta \in \mathcal{B}} \inf_{\bar{u} \in U^{\mathbb{N}}} J_{(F,G)}(x, \bar{u}, \beta(\bar{u})), \quad x \in X, \tag{5}$$

of this game. By standard dynamic programming arguments [8], one sees that this function fulfills the optimality principle

$$V_{(F,G)}(x) = \inf_{u \in U} \sup_{w \in W} \inf_{x_1 \in F(x,u,w)} \{G(x, x_1, u, w) + V_{(F,G)}(x_1)\}. \tag{6}$$

Observe that our original single-valued game (2), (3) can be recast in this multi-valued setting by defining $F(x, u, w) := \{f(x, u, w)\}$ and $G(x, x_1, u, w) := g(x, u)$. We will now investigate the relation of the value functions of different multivalued games. For this purpose, we first introduce the concept of an *enclosure*.

Definition 3.1 If (F_1, G_1) and (F_2, G_2) are two multivalued games such that $F_2(x, u, w) \subset F_1(x, u, w)$ for all x, u and w and $G_1(x, x', u, w) \leq G_2(x, x', u, w)$ for all $x, x' \in F_2(x, u, w)$ and all u and w , then (F_1, G_1) is called an *enclosure* of (F_2, G_2) .

Proposition 3.1 Let the game (F_1, G_1) be an enclosure of the game (F_2, G_2) . Then, $V_{(F_1,G_1)} \leq V_{(F_2,G_2)}$.

Proposition 3.2 Let the sequence of games (F_i, G_i) , $i \in \mathbb{N}$, be enclosures of the game (F, G) and assume

$$\sup_{x \in X, u \in U, w \in W} H(F_i(x, u, w), F(x, u, w)) \rightarrow 0, \quad \text{as } i \rightarrow \infty \tag{7}$$

(where H denotes the Hausdorff distance) and

$$\sup_{x, x_1 \in X, u \in U, w \in W} |G_i(x, x_1, u, w) - G(x, x_1, u, w)| \rightarrow 0, \quad \text{as } i \rightarrow \infty. \tag{8}$$

Assume furthermore that F is upper semicontinuous in x and that G is continuous in x and x_1 , both uniformly in u and w and on compact subsets of X . In addition, we assume that there exists $\alpha \in \mathcal{K}_\infty$ ¹ with $G(x, x_1, u, w) \geq \alpha(d(x, O))$ and $G_i(x, x_1, u, w) \geq \alpha(d(x, O))$, for all $i \in \mathbb{N}$, $u \in U$, $w \in W$, and that $V_{(F,G)}$ is continuous on ∂O . Then, for each compact set $K \subset X$ for which $\sup_{x \in K} V_{(F,G)}(x) < \infty$, we have

$$\sup_{x \in K} |V_{(F_i,G_i)}(x) - V_{(F,G)}(x)| \rightarrow 0, \quad \text{as } i \rightarrow \infty,$$

i.e., the uniform convergence on compact sets in the domain of $V_{(F,G)}$.

Proof Let $k^* : X^{\mathbb{N}} \rightarrow \mathbb{N}$ be a bounded map. Then, from the optimality principle (6), we obtain by induction

¹A function $\gamma : [0, \infty) \rightarrow [0, \infty)$ is of class \mathcal{K} if it is continuous, zero at zero and strictly increasing. It is of class \mathcal{K}_∞ , if, in addition, it is unbounded.

$$V_{(F,G)}(x) = \sup_{\beta \in \mathcal{B}} \inf_{\bar{u} \in U^{\mathbb{N}}} \inf_{\bar{x} \in \mathcal{X}_F(x, \bar{u}, \beta(\bar{u}))} \left\{ \sum_{k=0}^{k^*(\bar{x})-1} G(x_k, x_{k+1}, u_k, \beta(\bar{u})_k) + V_{(F,G)}(x_{k^*(\bar{x})}) \right\}.$$

Now let $\gamma := \sup_{x \in K} V_{(F,G)}(x)$. Due to the lower bound α on G , for every $\delta > 0$ there exists a time $k_{\gamma, \delta} \in \mathbb{N}$ such that for each trajectory $\bar{x} \in \mathcal{X}_F(x, \bar{u}, \beta(\bar{u}))$ with cost bounded by γ there exists a time $k^*(\bar{x}) \leq k_{\gamma, \delta}$ such that $x_{k^*(\bar{x})} \in B_{\delta}(O)$. We fix $\varepsilon > 0$ and $x \in K$ and choose $\delta > 0$ such that $V_{(F,G)}(x) \leq \varepsilon$ for all $x \in B_{\delta}(O)$ (δ exists because of the continuity of $V_{(F,G)}$ on ∂O). Then, using an ε -optimal perturbation strategy $\beta^* \in \mathcal{B}$ and an arbitrary $\bar{u}^* \in U^{\mathbb{N}}$, from the above optimality principle we obtain

$$\begin{aligned} V_{(F,G)}(x) &\leq \inf_{\bar{u} \in U^{\mathbb{N}}} \inf_{\bar{x} \in \mathcal{X}_F(x, \bar{u}, \beta^*(\bar{u}))} \left\{ \sum_{k=0}^{k^*(\bar{x})-1} G(x_k, x_{k+1}, u_k, \beta^*(\bar{u})_k) + V_{(F,G)}(x_{k^*(\bar{x})}) \right\} \\ &\quad + \varepsilon \\ &\leq \inf_{\bar{u} \in U^{\mathbb{N}}} \inf_{\bar{x} \in \mathcal{X}_F(x, \bar{u}, \beta^*(\bar{u}))} \left\{ \sum_{k=0}^{k^*(\bar{x})-1} G(x_k, x_{k+1}, u_k, \beta^*(\bar{u})_k) \right\} + 2\varepsilon \\ &\leq \inf_{\bar{x} \in \mathcal{X}_F(x, \bar{u}^*, \beta^*(\bar{u}^*))} \left\{ \sum_{k=0}^{k^*(\bar{x})-1} G(x_k, x_{k+1}, u_k^*, \beta^*(\bar{u}^*)_k) \right\} + 2\varepsilon. \end{aligned}$$

Now, fixing β^* , for any $i \in \mathbb{N}$ we can pick an ε -optimal control \bar{u}_i^* , yielding

$$\begin{aligned} \gamma &\geq V_{(F_i, G_i)}(x) \\ &\geq \inf_{\bar{x} \in \mathcal{X}_{F_i}(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))} \left\{ \sum_{k=0}^{\infty} G_i(x_k, x_{k+1}, (\bar{u}_i^*)_k, \beta^*(\bar{u}_i^*)_k) \right\} - \varepsilon \\ &\geq \inf_{\bar{x} \in \mathcal{X}_{F_i}(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))} \left\{ \sum_{k=0}^{k^*(\bar{x})} G_i(x_k, x_{k+1}, (\bar{u}_i^*)_k, \beta^*(\bar{u}_i^*)_k) \right\} - \varepsilon. \end{aligned}$$

In particular, this last expression is bounded by γ and hence the lower bound α for G_i implies that there exists a compact set K_1 such that each ε -optimal trajectory $(x_k)_k \in \mathcal{X}_{F_i}(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))$ lies in K_1 for all $i \in \mathbb{N}$.

Now assumption (7) and the upper semicontinuity of F imply that, for each $\varepsilon_1 > 0$, there exists an $i_0 \in \mathbb{N}$ such that, for $i \geq i_0$ and each such ε -optimal trajectory $(x_k)_k \in \mathcal{X}_{F_i}(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))$, there exists a trajectory $(\tilde{x}_k)_k \in \mathcal{X}_F(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))$ with $\|x_k - \tilde{x}_k\| \leq \varepsilon_1$ for all $k = 1, \dots, k_{\gamma, \delta}$. Hence (8) and the continuity of G imply that we can find $i_1 \in \mathbb{N}$ such that

$$\begin{aligned} &\left| \inf_{(x_k)_k \in \mathcal{X}_F(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))} \left\{ \sum_{k=0}^{k^*} G(x_k, x_{k+1}, (\bar{u}_i^*)_k, \beta^*(\bar{u}_i^*)_k) \right\} \right. \\ &\quad \left. - \inf_{(x_k)_k \in \mathcal{X}_{F_i}(x, \bar{u}_i^*, \beta^*(\bar{u}_i^*))} \left\{ \sum_{k=0}^{k^*} G_i(x_k, x_{k+1}, (\bar{u}_i^*)_k, \beta^*(\bar{u}_i^*)_k) \right\} \right| \leq \varepsilon, \end{aligned}$$

for all $i \geq i_1$ and all $k^* \in \{1, \dots, k_{\gamma, \delta}\}$. Combining this inequality with the estimates for $V_{(F,G)}$ and $V_{(F_i,G_i)}$ using $\bar{u}^* = \bar{u}_i^*$ in the former, we obtain $V_{(F,G)}(x) \leq V_{(F_i,G_i)}(x) + 5\varepsilon$ for all $i \geq i_1$. Since i_1 depends only on $k_{\gamma, \delta}$ and ε , hence only on the set K and not on the individual x , we obtain the desired uniform convergence. \square

Remark 3.1 Note that we have obtained our result under very weak assumptions on F and G using, however, the crucial continuity assumption of $V_{(F,G)}$ on ∂O . This implicit assumption is difficult to check directly but can be ensured by an asymptotic controllability assumption, see [6]. We emphasize that we only need continuity at the boundary of O and that the value function may be discontinuous elsewhere.

4 State Space Constraints

So far we have assumed $F(x, u, w) \neq \emptyset$ for all $x \in X, u \in U, w \in W$ which guarantees that for, each initial value x and each pair of control and perturbation sequences \bar{u} and \bar{w} , we obtain at least one trajectory $(x_k)_k$ which is defined for all $k \in \mathbb{N}_0$. However, in practice, it will often be necessary to relax this assumption.

In order to motivate this relaxation, assume that we are given a multivalued game (\tilde{F}, G) on a state space $\tilde{X} \subseteq \mathbb{R}^d$. In our numerical approach, the state space set X on which we can solve the problem will be a compact set while the state space \tilde{X} of the given problem is often unbounded. In addition, from a modeling point of view it might be desirable to introduce state constraints, e.g., in order to avoid certain critical regions of the state space. In both cases, it will be necessary to restrict the state space of the original problem defining $F(x, u, w) := \tilde{F}(x, u, w) \cap X, x \in X, u \in U, w \in W$. This construction may result in $F(x, u, w) = \emptyset$ for certain $x \in X, u \in U, w \in W$ and consequently it may happen that a solution trajectory will only exist for finite time. More precisely, for given F , given $\bar{u} = (u_k)_k \in U^{\mathbb{N}}$, given $\bar{w} = (w_k)_k \in W^{\mathbb{N}}$ and any sequence $\bar{x} = (x_k)_k \in X^{\mathbb{N}}$, let

$$k_F^{\max}(\bar{x}, \bar{u}, \bar{w}) = \max\{\hat{k} \in \mathbb{N} : x_{k+1} \in F(x_k, u_k, w_k), k = 0, \dots, \hat{k} - 1\}$$

be the maximal index up to which the sequence \bar{x} constitutes a trajectory of F . Since a trajectory with $k_F^{\max}(\bar{x}, \bar{u}, \bar{w}) < \infty$ cannot converge to the set O , we set

$$J_{(F,G)}(x, \bar{u}, \bar{w}) := \infty \quad \text{if } k_F^{\max}(\bar{x}, \bar{u}, \bar{w}) < \infty \text{ for each } \bar{x} \in X^{\mathbb{N}} \text{ with } x = x_0 \notin O.$$

It is easy to see that Proposition 3.1 remains valid in this case, while Proposition 3.2 is more difficult to recover in this setting. The reason lies in the fact that any enclosure will necessarily enlarge the set of possible trajectories, even if we apply the same state space constraints to F and F_i . In the presence of state space constraints this means that for any i there may exist a trajectory \bar{x}_i of F_i for which all nearby trajectories \bar{x} of F violate the space constraints. In other words, unless very specific knowledge about the dynamics F is available and used for the construction of the enclosure F_i , the enlargement of the dynamics has the implicit effect of relaxing the state space constraints.

However, if we assume that the optimal value function is continuous with respect to relaxations of the state space constraints, then we can recover Proposition 3.2. In order to formalize this relaxation, for $\varepsilon > 0$ we define the space $X_\varepsilon := \{x \in \tilde{X} \mid d(x, X) \leq \varepsilon\}$, the multivalued dynamics $F_\varepsilon(x, u, w) := \tilde{F}(x, u, w) \cap X_\varepsilon$ and the related optimal value function $V_{(F_\varepsilon, G)}$. Using this notation we can prove the following variant of Proposition 3.2.

Proposition 4.1 *Consider the state space constrained dynamics F of \tilde{F} and consider a sequence of enclosures (F_i, G_i) of F on X . Let the assumptions of Proposition 3.2 hold for F and F_i , where (7) in the case of $F(x, u, w) = \emptyset$ is to be understood as*

$$F_i(x, u, w) = \emptyset, \quad \text{for all } i \in \mathbb{N} \text{ and all } x, u, w \text{ with } F(x, u, w) = \emptyset.$$

Assume, furthermore, that \tilde{F} is upper semi-continuous in x uniformly in u and w on compact subsets of \tilde{X} and let $\|\cdot\|_p$ be the usual p -norm for real valued functions on X for some $p \in \{1, \dots, \infty\}$.

Then for each compact set $K \subset X$ for which $\sup_{x \in K} V_{(F, G)}(x) < \infty$ and on which the continuity assumption

$$\|V_{(F_\varepsilon, G)}|_K - V_{(F, G)}|_K\|_p \rightarrow 0, \quad \text{as } \varepsilon \rightarrow 0 \tag{9}$$

holds, we have $\|V_{(F_i, G_i)}|_K - V_{(F, G)}|_K\|_p \rightarrow 0$ as $i \rightarrow \infty$.

Proof see [6]. □

Remark 4.1

We would like to emphasize that our result allows for a rigorous convergence proof of the approximating multivalued game in the presence of discontinuities, a feature which is rarely found in other approximation techniques.

5 Discretization of the Game

In this section, we describe the set-oriented discretization technique which transforms our problem into a graph theoretic problem. In order to introduce our method, we first recall the corresponding procedure for unperturbed systems developed in [1] before we turn to the general setting.

5.1 Discretizing the Unperturbed System

If X is finite and there are no perturbations, then one can use a shortest path algorithm like Dijkstra’s method [9], see also Sect. 9, in order to compute the value function, see e.g. [8]. In [1], it has been shown how to discretize general optimal control problems with continuous state space such that this approach can be applied. We review this method here in a different formulation that carries over directly to the case of a perturbed control system in the next section.

We consider a single-valued control system $f : X \times U \rightarrow X$ (f continuous, $X \subset \mathbb{R}^d$ and $U \subset \mathbb{R}^m$ compact, $0 \in X, 0 \in U, f(0, 0) = 0$), together with a continuous cost function $g : X \times U \rightarrow [0, \infty)$ with $g(x, u) > 0$ for $x \neq 0$ and $g(0, 0) = 0$. Let \mathcal{P} be a finite partition of X , i.e. \mathcal{P} is a finite set of mutually disjoint subsets $P \subset X$. Define the map $\pi : X \rightarrow \mathcal{P}, \pi(x) = P, x \in P$, as well as $\rho : X \rightrightarrows X, \rho = \pi^{-1} \circ \pi$ (i.e. to each x, ρ associates the set of the partition \mathcal{P} which contains x).

Box-Enclosure of the System. Consider the multivalued game (which is actually a multivalued control system, since there are no perturbations here) (F, G) with

$$F(x, u, w) = F(x, u) := \rho(f(x, u)) \quad \text{and} \quad G(x, x_1, u, w) = g(x, u).$$

The optimality principle (6) in this case reads

$$V_{(F,G)}(x) = \inf_{u \in U} \left\{ g(x, u) + \inf_{x_1 \in F(x,u)} V_{(F,G)}(x_1) \right\}. \tag{10}$$

Projection onto Piecewise Constant Functions. The right-hand side of (10) defines an operator on real-valued functions on X , the *dynamic programming operator* $L : \mathbb{R}^X \rightarrow \mathbb{R}^X$,

$$L[v](x) = \inf_{u \in U} \left\{ g(x, u) + \inf_{x_1 \in F(x,u)} v(x_1) \right\}.$$

Note that the optimal value function $V_{(F,G)}$ is, by definition of L , a fixed point of L , i.e. $L[V_{(F,G)}] = V_{(F,G)}$. Abusing notation, we identify the space $\mathbb{R}^{\mathcal{P}}$ with the subspace of real valued functions on X that are piecewise constant on the elements of the partition \mathcal{P} (in fact, we view $v \in \mathbb{R}^{\mathcal{P}}$ as the function $v \circ \pi \in \mathbb{R}^X$). We define the projection $\varphi : \mathbb{R}^X \rightarrow \mathbb{R}^{\mathcal{P}} \subset \mathbb{R}^X, \varphi[v](x) = \inf_{x' \in \rho(x)} v(x')$, and the corresponding *discretized dynamic programming operator* $L_{\mathcal{P}} : \mathbb{R}^{\mathcal{P}} \rightarrow \mathbb{R}^{\mathcal{P}}, L_{\mathcal{P}} = \varphi \circ L$. Explicitly, the discretized operator reads

$$\begin{aligned} L_{\mathcal{P}}[v](x) &= \inf_{x' \in \rho(x)} \left\{ \inf_{u \in U} \left\{ g(x', u) + \inf_{x_1 \in F(x',u)} v(x_1) \right\} \right\} \\ &= \inf_{x' \in \rho(x), u \in U} \left\{ g(x', u) + v(f(x', u)) \right\}, \end{aligned}$$

since $v \in \mathbb{R}^{\mathcal{P}}$ is constant on each element of \mathcal{P} , i.e. on each set $F(x', u)$.

We define the discretized optimal value function $V_{\mathcal{P}} \in \mathbb{R}^{\mathcal{P}}$ as the unique fixed point of $L_{\mathcal{P}}$ with $V_{\mathcal{P}}(0) = 0$. Then, $V_{\mathcal{P}}$ satisfies the optimality principle

$$V_{\mathcal{P}}(x) = \inf_{x' \in \rho(x), u \in U} \left\{ g(x', u) + V_{\mathcal{P}}(f(x', u)) \right\}. \tag{11}$$

Graph Theoretic Formulation. Note that, since \mathcal{P} is finite, $V_{\mathcal{P}}(f(x', u))$ in (11) can only take finitely many values. We can therefore rewrite (11) as

$$V_{\mathcal{P}}(x) = \min_{P \in \pi(f(\rho(x), U))} \inf_{x' \in \rho(x), u \in U: f(x', u) \in P} \left\{ g(x', u) + V_{\mathcal{P}}(P) \right\}, \tag{12}$$

where $V_{\mathcal{P}}(P) = V_{\mathcal{P}}(x)$ for any $x \in P \in \mathcal{P}$. If we define the multivalued map (or, equivalently, the directed graph) $\mathcal{F} : \mathcal{P} \rightrightarrows \mathcal{P}$, $\mathcal{F}(P) = \pi(f(\pi^{-1}(P), U))$, $P \in \mathcal{P}$, and the cost function $\mathcal{G}(P', P) = \inf\{g(x, u) \mid x \in P', f(x, u) \in P, u \in U\}$, we can rewrite (12) as the *discrete optimality principle*

$$V_{\mathcal{P}}(P) = \min_{P_1 \in \mathcal{F}(P)} \{ \mathcal{G}(P, P_1) + V_{\mathcal{P}}(P_1) \}.$$

5.2 Discretization of the Perturbed System

Now, we want to carry over the discretization procedure from the last section to our game setting. We proceed in a completely analogous way, additionally incorporating the perturbations. This will ultimately lead to a directed hypergraph (a *forward hypergraph* or *F-graph* in the terminology of [10]), instead of an ordinary graph for which we formulate the associated shortest path algorithm at the end of the section.

Box-Enclosure of the System. Consider the multivalued game (F, G) with

$$F(x, u, w) = \rho(f(x, u, w)) \quad \text{and} \quad G(x, x_1, u, w) = g(x, u), \tag{13}$$

(where f and g are the control system and cost function introduced in Sect. 2). From the optimality principle (6), we obtain

$$\begin{aligned} V_{(F,G)}(x) &= \inf_{u \in U} \sup_{w \in W} \inf_{x_1 \in F(x,u,w)} \{ g(x, u) + V_{(F,G)}(x_1) \} \\ &= \inf_{u \in U} \left\{ g(x, u) + \sup_{w \in W} \inf_{x_1 \in F(x,u,w)} V_{(F,G)}(x_1) \right\}. \end{aligned}$$

Projection onto Piecewise Constant Functions. The dynamic programming operator $L : \mathbb{R}^X \rightarrow \mathbb{R}^X$ here reads

$$L[v](x) = \inf_{u \in U} \left\{ g(x, u) + \sup_{w \in W} \inf_{x_1 \in F(x,u,w)} v(x_1) \right\}.$$

Correspondingly, the discretized operator $L_{\mathcal{P}} : \mathbb{R}^{\mathcal{P}} \rightarrow \mathbb{R}^{\mathcal{P}}$ is given by

$$\begin{aligned} L_{\mathcal{P}}[v](x) &= \inf_{x' \in \rho(x)} \left\{ \inf_{u \in U} \left\{ g(x', u) + \sup_{w \in W} \inf_{x_1 \in F(x',u,w)} v(x_1) \right\} \right\} \\ &= \inf_{x' \in \rho(x), u \in U} \left\{ g(x', u) + \sup_{x_1 \in F(x',u,W)} v(x_1) \right\}, \end{aligned}$$

since $v \in \mathbb{R}^{\mathcal{P}}$ is constant on each element of \mathcal{P} , i.e. on each set $F(x', u, w)$.

We define the discretized optimal value function $V_{\mathcal{P}} \in \mathbb{R}^{\mathcal{P}}$ as the unique fixed point of $L_{\mathcal{P}}$ with $V_{\mathcal{P}}(P) = 0$ for all partition elements $P \in \mathcal{P}$ with $\pi^{-1}(P) \cap O \neq \emptyset$. Then, $V_{\mathcal{P}}$ satisfies the optimality principle

$$V_{\mathcal{P}}(x) = \inf_{x' \in \rho(x), u \in U} \left\{ g(x', u) + \sup_{x_1 \in F(x',u,W)} V_{\mathcal{P}}(x_1) \right\}. \tag{14}$$

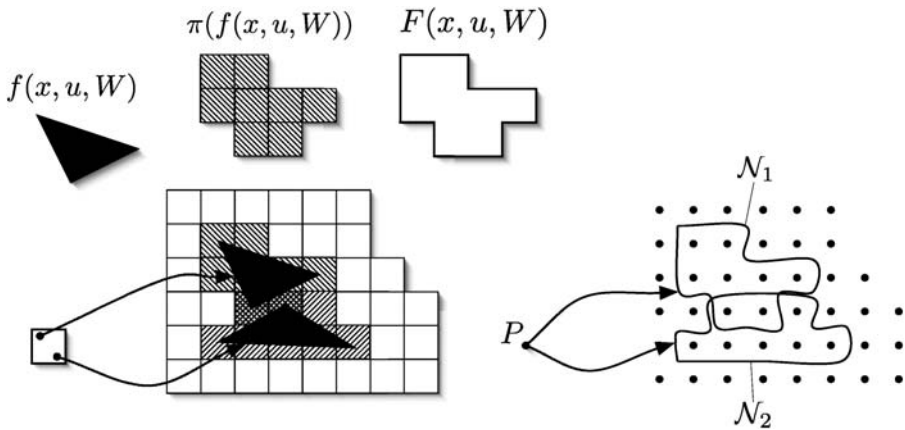


Fig. 1 Illustration of the construction of the hypergraph

Graph Theoretic Formulation. In order to derive the corresponding shortest path algorithm, it is useful to formulate (14) equivalently in terms of an associated graph. To this end note that for any pair $(x, u) \in X \times U$, the set $F(x, u, W) \subset X$ is the union of a finite set of elements from the partition \mathcal{P} . In particular, the family $\{F(x', u, W) : (x', u) \in \rho(x) \times U\}$ of subsets of X is finite for any $x \in X$. Putting this in terms of a corresponding map on \mathcal{P} : each partition element P is mapped to a finite family $\{\mathcal{N}_i\}_{i=1, \dots, i(P)}, \mathcal{N}_i \subset \mathcal{P}$, of subsets of \mathcal{P} under all perturbations. Formally, we have a directed hypergraph (\mathcal{P}, E) with the set $E \subset \mathcal{P} \times 2^{\mathcal{P}}$ of hyperedges given by $E = \{(P, \mathcal{N}) \mid \pi(F(x, u, W)) = \mathcal{N} \text{ for some } (x, u) \in P \times U\}$, or equivalently, the multivalued map $\mathcal{F} : \mathcal{P} \rightrightarrows 2^{\mathcal{P}}, \mathcal{F}(P) = \{\pi(F(x, u, W)) : (x, u) \in P \times U\}$, cf. Fig. 1.

If we define weights on the edges of this hypergraph by $\mathcal{G}(P, \mathcal{N}) = \inf\{g(x, u) : (x, u) \in P \times U, \pi(F(x, u, W)) = \mathcal{N}\}$, then we can write (14) equivalently as

$$V_{\mathcal{P}}(P) = \inf_{\mathcal{N} \in \mathcal{F}(P)} \left\{ \mathcal{G}(P, \mathcal{N}) + \sup_{N \in \mathcal{N}} V_{\mathcal{P}}(N) \right\}. \tag{15}$$

Dijkstra’s Method for the Perturbed System. We are now going to generalize Dijkstra’s algorithm (see Sect. 9) such that it computes the value function of a weighted directed hypergraph (i.e. the function defined by the optimality principle (15)).

Let $(\mathcal{P}, E), E \subset \mathcal{P} \times 2^{\mathcal{P}}$, be a hypergraph with weights $\mathcal{G} : E \rightarrow [0, \infty)$. In order to adapt Algorithm 9.1, we need to modify the relaxing step in lines 7–9, such that the maximization over all perturbations (i.e. over $N \in \mathcal{N}$) in (15) is taken into account. The modified version of lines 7–9 reads:

For each $(Q, \mathcal{N}) \in E$ with $P \in \mathcal{N}$ do:
 if $V(Q) > \mathcal{G}(Q, \mathcal{N}) + \max_{N \in \mathcal{N}} V(N)$, then
 set $V(Q) := \mathcal{G}(Q, \mathcal{N}) + \max_{N \in \mathcal{N}} V(N)$.

As justified by Proposition 9.1 (see Sect. 9), if $\mathcal{N} \subset \mathcal{P} \setminus Q$, then $\max_{N \in \mathcal{N}} V(N) = V(P)$, and the node Q will never be relaxed again. On the other hand, if $\mathcal{N} \not\subset \mathcal{P} \setminus Q$,

then Q will be relaxed at a later time again and we do not need to relax it in this iteration of the while-loop. These considerations lead to the following further modification of lines 7–9:

For each $(Q, \mathcal{N}) \in E$ with $P \in \mathcal{N}$, do:
 If $\mathcal{N} \subset \mathcal{P} \setminus Q$, then
 if $V(Q) > \mathcal{G}(Q, \mathcal{N}) + V(P)$, then
 set $V(Q) := \mathcal{G}(Q, \mathcal{N}) + V(P)$.

Including the adapted initialization, the overall algorithm for the case of a perturbed system reads as follows. Here, $\mathcal{D} \subset \mathcal{P}$ is the set of destination nodes which typically will be chosen as $\mathcal{D} = \{P \in \mathcal{P} : P \cap O \neq \emptyset\}$ (with the robust forward invariant set O from Sect. 2).

Algorithm 5.1 MINMAX-DIJKSTRA($(\mathcal{P}, E), \mathcal{G}, \mathcal{D}$)

1 for each $P \in \mathcal{P}$, set $V(P) := \infty$
 2 for each $P \in \mathcal{D}$, set $V(P) := 0$
 3 $Q := \mathcal{P}$
 4 while $Q \neq \emptyset$
 5 $P := \operatorname{argmin}_{P' \in Q} V(P')$
 6 $Q := Q \setminus \{P\}$
 7 for each $(Q, \mathcal{N}) \in E$ with $P \in \mathcal{N}$
 8 if $\mathcal{N} \subset \mathcal{P} \setminus Q$, then
 9 if $V(Q) > \mathcal{G}(Q, \mathcal{N}) + V(P)$, then
 10 $V(Q) := \mathcal{G}(Q, \mathcal{N}) + V(P)$

We note that this algorithm bears similarities with the SBT-algorithm in [10]. However, in our case the graph has a special structure (namely, the heads of the hyperedges consist of only a single node, i.e. we have an F -graph as defined in [10]). This yields the subquadratic complexity in the number of nodes as derived in Sect. 6.1 and thus gives an improvement over SBT.

6 Implementation and Numerical Examples

6.1 Implementation

In the numerical realization we always let the state space X be a box in \mathbb{R}^d and construct a partition \mathcal{P} of it by dividing X uniformly into smaller boxes. In fact, we realize this division by repeatedly bisecting the current division (changing the coordinate direction after each bisection). The resulting sequence of partitions can efficiently be stored as a binary tree—see [11] for more details.

In order to compute (or rather approximate) the set $E \subset \mathcal{P} \times 2^{\mathcal{P}}$ of hyperedges, we choose finite sets $\tilde{P} \subset P$, $\tilde{U} \subset U$ and $\tilde{W} \subset W$ of *test points*—typically on an equidistant grid in each of these sets. We then compute $\tilde{\mathcal{F}}(P) := \{\pi(F(x, u, \tilde{W})) : (x, u) \in \tilde{P} \times \tilde{U}\} \subset 2^{\mathcal{P}}$ as an approximation to $\mathcal{F}(P)$ and correspondingly approximate the weights by $\tilde{\mathcal{G}}(P, \mathcal{N}) = \min\{g(x, u) : (x, u) \in \tilde{P} \times \tilde{U}, \pi(F(x, u, \tilde{W})) = \mathcal{N}\}$.

Time and Space Complexity. The time complexity of the standard Dijkstra algorithm (Algorithm 9.1 in the appendix) strongly depends on the data structure which is used in order to store the set \mathcal{Q} . In particular, the complexity of the operations in lines 5 (extracting the node with minimal V -value) and line 9 (decreasing the V -value and the associated reorganization of the data structure) have a crucial influence. In our implementation we are using a binary heap in order to store \mathcal{Q} which leads to a complexity of $\mathcal{O}((|\mathcal{P}| + |E|) \log |\mathcal{P}|)$.

In the perturbed case (Algorithm 5.1), each hyperedge is considered at most N times in line 7, with N being a bound on the cardinality of the hypernodes \mathcal{N} . Additionally, we need to perform the check in line 8, which has linear complexity in N . Furthermore, line 10 of the algorithm is executed at most once per hyperedge and thus the overall complexity of the perturbed Dijkstra algorithm is $\mathcal{O}(|\mathcal{P}| \log |\mathcal{P}| + |E|N^2 + |E| \log |\mathcal{P}|)$, see also [12].

The space requirements grow linearly with the number of partition elements. Since typically the whole state space has to be covered, this number grows exponentially with the dimension of phase space (assuming a uniform partitioning). The concrete storage consumption strongly depends on the properties of the underlying control system. While the number of hyperedges is essentially determined by the Lipschitz constant of f , the size of the hypernodes \mathcal{N} will crucially be influenced by the size of the perturbation. In the applications that we have in mind in this paper, these numbers are of moderate size.

As a rule of thumb, the main computational effort in our approach goes into the construction of the hypergraph via the mapping of test points—in particular, if the system is given by a short-time integration of a continuous time system. Note that this “sampling” of the system will be required in any method that computes the value function. Typically however, in standard methods like value iteration, certain points are sampled multiple times which leads to a higher computational effort in comparison to our approach.

6.2 Numerical Examples

A Simple 1D System. We start by looking at an additively perturbed version of a simple 1D map from [2],

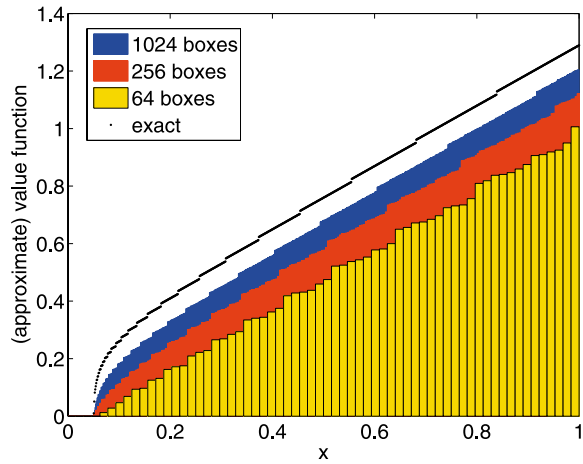
$$x_{k+1} = x_k + (1 - a)u_k x_k + w_k, \quad k = 0, 1, \dots,$$

with $x_k \in [0, 1]$, $u_k \in [-1, 1]$, $w_k \in [-\varepsilon, \varepsilon]$ for some $\varepsilon > 0$ and the fixed parameter $a \in (0, 1)$. The cost function is $g(x, u) = (1 - a)x$ so that (regardless of how the perturbation sequence is chosen) the optimal control policy is to steer to the origin as fast as possible, i.e. to choose $u_k = -1$ for all k . Similarly, the optimal strategy for the “perturbing player” is to slow down the dynamics as much as possible, corresponding to $w_k = \varepsilon$ for all k . The resulting dynamical system is the affine linear map

$$x_{k+1} = ax_k + \varepsilon, \quad k = 0, 1, \dots,$$

which has a fixed point at $x = \varepsilon/(1 - a)$, i.e. under worst-case conditions (assuming $w_k = \varepsilon$ for all k), it will be impossible to get any closer than $\alpha_0 := \varepsilon/(1 - a)$ to the

Fig. 2 Perturbed simple 1D map: Upper value function and its approximations on various partitions



origin. Correspondingly, we choose a neighborhood $O = [0, \alpha]$ with $\alpha > \alpha_0$ as our target region. With $k(x) = \left\lceil \frac{\log \frac{\alpha - \alpha_0}{x - \alpha_0}}{\log a} \right\rceil + 1$, the exact optimal value function is

$$V(x) = (x - \alpha_0)(1 - a^{k(x)}) + \varepsilon k(x),$$

as shown in Fig. 2 for $a = 0.8$, $\varepsilon = 0.01$ and $\alpha = 1.1\alpha_0$. In that figure, we also show the approximate optimal value functions on partitions of 64, 256 and 1024 intervals, respectively. In the construction of the hypergraph, we used an equidistant grid of ten points in each partition interval, in the control space and in the perturbation space.

The Inverted Pendulum—Reloaded. As a more challenging test case, we reconsider the problem of designing an optimal globally stabilizing controller for an inverted pendulum on a cart (see [1, 2]):

$$\left(\frac{4}{3} - m_r \cos^2 \varphi\right) \ddot{\varphi} + \frac{1}{2} m_r \dot{\varphi}^2 \sin 2\varphi - \frac{g}{\ell} \sin \varphi = -u \frac{m_r}{m\ell} \cos \varphi. \tag{16}$$

The equation models the (planar) motion of an inverted pendulum with mass $m = 2$ on a cart with mass $M = 8$ which moves under an applied horizontal force u . The angle φ measures the offset angle from the vertical up position. The parameter $m_r = m/(m + M)$ is the mass ratio and $\ell = 0.5$ the distance of the pendulum mass from the pivot. We use $g = 9.8$ for the gravitational constant. The instantaneous cost is

$$q(\varphi, \dot{\varphi}, u) = \frac{1}{2}(0.1\varphi^2 + 0.05\dot{\varphi}^2 + 0.01u^2). \tag{17}$$

Denoting the evolution operator of the control system (16) for constant control functions u by $\Phi^t(t, u)$, we consider the time- T -map $\Phi^T(x, u)$ of this system as our discrete-time system with $T = 0.1$. The map Φ^T is approximated via the classical Runge-Kutta scheme of order 4 with stepsize 0.02. Thus, we arrive at the cost function

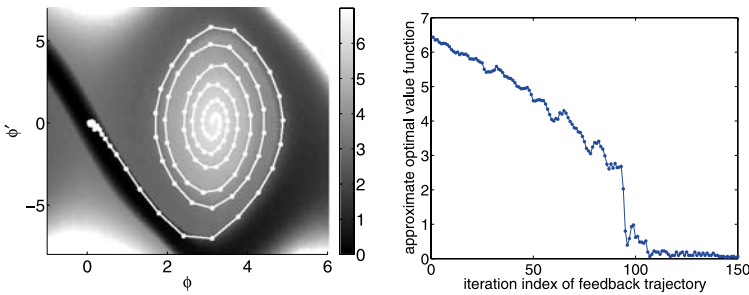


Fig. 3 Approximate optimal value function and feedback trajectory (left) and the approximate optimal value function along the feedback trajectory (right) for the inverted pendulum on a 2^{18} box partition

$$g(\varphi, \dot{\varphi}, u) = \int_0^T q(\Phi^t((\varphi, \dot{\varphi}), u), u) dt.$$

We choose $X = [-8, 8] \times [-10, 10]$ as the region of interest.

In [2], a feedback trajectory with initial value (3.1, 0.1) was computed that was based on an approximate optimal value function on a partition of 2^{18} boxes (cf. Fig. 3 (left)). In contrast to what one might expect, the approximate optimal value function does actually not decrease monotonically along this trajectory (cf. Fig. 3). This effect is due to the fact that the discretization method used in [2] allows for jumps in the trajectories which cannot be reproduced by the real system. The fact that the approximate optimal value function is not always decreasing indicates that the approximation accuracy in this example is just fine enough to allow for stabilization, and in fact, on a coarser partition of 2^{14} boxes, the associated feedback is not stabilizing this initial condition any more.

We are now going to use the approach developed in this paper in order to design a stabilizing feedback controller on basis of the coarser partition (2^{14} boxes). To this end, we imagine the perturbation of our system being given as “for a given state $(\varphi, \dot{\varphi})$, be prepared to start anywhere in the box that contains $(\varphi, \dot{\varphi})$ ”, i.e. we define our game by $F((\varphi, \dot{\varphi}), u, W) := \Phi^T(B, u)$, where $B \in \mathcal{P}$ is the box in the partition \mathcal{P} under consideration which contains the point $(\varphi, \dot{\varphi})$. Note that we do not need to parameterize the points in $\Phi^T(B, u)$ with $w \in W$ for the construction of the hypergraph.

Figure 4 shows the approximate upper value function on a partition of 2^{14} boxes with target region $O = [-0.1, 0.1]^2$ as well as the trajectory generated by the associated feedback for the initial value (3.1, 0.1). As expected, the approximate value function is decreasing monotonically along this trajectory. Furthermore, despite the fact that we used considerably fewer boxes as for Fig. 3, the resulting trajectory is obviously closer to the optimal one because it converges to the origin much faster.

7 Convergence Analysis

In this section, we show that (and in which sense) the approximate optimal value function constructed in the preceding section converges to the true one as the underlying

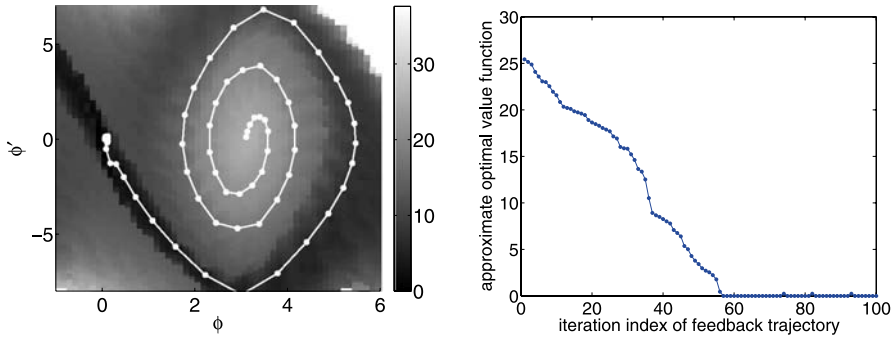


Fig. 4 Approximate upper value function and feedback trajectory (left) and the approximate upper value function along the feedback trajectory (right) for the inverted pendulum on a 2^{14} box partition using the robust feedback construction

partitions are refined, using the abstract results for multivalued games developed in the Sects. 3 and 4. We begin with the following observation on the relation between $V_{\mathcal{P}}$ and $V_{(F,G)}$ with F, G from (13).

Proposition 7.1 Consider the discretized optimal value function $V_{\mathcal{P}}$ and the optimal value function $V_{(F,G)}$ from (5) corresponding to the game (13). If $V_{(F,G)}$ is continuous on ∂O , then these functions are related by $V_{\mathcal{P}}(x) = \inf_{x' \in \rho(x)} V_{(F,G)}(x')$.

Proof See [6]. □

We consider now a sequence of increasingly finer partitions of X and ask under which conditions the corresponding approximate optimal value functions converge to the value function of the game (f, g) . In a nested sequence of partitions, each element of a partition is contained in an element of the preceding partition.

The following theorem states our main convergence result. It shows that we obtain L^∞ convergence on compact sets on which $V_{(f,g)}$ is continuous and—under a mild regularity condition on the set of discontinuities— L^1 convergence on every compact set on which $V_{(f,g)}$ is bounded. We consider first problems without state space constraints and address the constrained case in Remark 7.1, below.

Theorem 7.1 Let $(\mathcal{P}_i)_{i \in \mathbb{N}}$ be a nested sequence of partitions of X such that

$$\sup_{x \in X} H(\rho_i(x), \{x\}) \rightarrow 0 \quad \text{as } i \rightarrow \infty.$$

Assume that $g(x, u)$ is continuous, that $g(x, u) > 0$ for $x \notin O$ and that $V_{(f,g)}$ is continuous on ∂O . Then $\|V_{\mathcal{P}_i}|_{K_i} - V_{(f,g)}|_{K_i}\|_\infty \rightarrow 0$ as $i \rightarrow \infty$ for every compact set $K \subseteq X$ on which $V_{(f,g)}$ is continuous and $K_i = \bigcup_{P \in \mathcal{P}_i, \pi^{-1}(P) \subset K} \pi^{-1}(P)$ being the largest subset of K which is a union of partition elements $P \in \mathcal{P}_i$.

If we assume furthermore that the set of discontinuities of $V_{(f,g)}$ has zero Lebesgue measure, then $\|V_{\mathcal{P}_i}|_K - V_{(f,g)}|_K\|_{L^1} \rightarrow 0$ as $i \rightarrow \infty$ on every compact set $K \subseteq X$ with $\sup_{x \in K} V_{(f,g)}(x) < \infty$.

Proof We use Proposition 3.2 with $(F, G) = (f, g)$ (f interpreted as a set valued map) and Proposition 7.1. Note that since $F_i(x, u, w) = \rho_i(f(x, u, w))$ and $G_i(x, u, w) = g(x, u)$, the games (F_i, G_i) are enclosures of (f, g) (in fact, since the sequence of partitions is nested, for every i , (F_i, G_i) is an enclosure of (F_{i+1}, G_{i+1})). Under the assumptions of the theorem, all assumptions of Proposition 3.2 are satisfied. In particular, by the assumptions on g and since X and U are compact, we know that there exists a function $\alpha \in \mathcal{K}_\infty$ such that $G_i(x, x_1, u, w) = g(x, u) \geq \alpha(d(x, O) + d(x_1, O))$ for all i . Thus, $V_{(F_i, G_i)}$ converges uniformly to $V_{(f, g)}$ on K . In order to show the L^∞ convergence on K_i observe that if $V_{(f, g)}$ is continuous on K then it is also uniformly continuous on K which implies $\sup_{P \in \mathcal{P}_i, \pi^{-1}(P) \subset K} |\inf_{x \in P} V_{(f, g)}(x) - \sup_{x \in P} V_{(f, g)}(x)| \rightarrow 0$ as $i \rightarrow \infty$. Thus, we can use Proposition 7.1 in order to conclude that

$$\begin{aligned} \|V_{\mathcal{P}_i}|_{K_i} - V_{(f, g)}|_{K_i}\|_\infty &\leq \sup_{P \in \mathcal{P}_i, \pi^{-1}(P) \subset K} \left| V_{\mathcal{P}_i}|_P - \sup_{x \in P} V_{(f, g)}(x) \right| \\ &= \sup_{P \in \mathcal{P}_i, \pi^{-1}(P) \subset K} \left| \inf_{y \in P} V_{(F_i, G_i)}(y) - \sup_{x \in P} V_{(f, g)}(x) \right| \\ &\leq \sup_{P \in \mathcal{P}_i, \pi^{-1}(P) \subset K} \left\{ \left| \inf_{y \in P} V_{(F_i, G_i)}(y) - \inf_{x \in P} V_{(f, g)}(x) \right| \right. \\ &\quad \left. + \left| \inf_{x \in P} V_{(f, g)}(x) - \sup_{x \in P} V_{(f, g)}(x) \right| \right\} \rightarrow 0, \end{aligned}$$

as $i \rightarrow \infty$. In order to show the L^1 convergence, observe that the uniform convergence $V_{(F_i, G_i)} \rightarrow V_{(f, g)}$ on K implies $\|V_{(F_i, G_i)}|_K - V_{(f, g)}|_K\|_{L^1} \rightarrow 0$ as $i \rightarrow \infty$. It thus remains to show that $V_{(F_i, G_i)}|_K - V_{\mathcal{P}_i}|_K \rightarrow 0$ in L^1 . Let D be the set of discontinuities of $V_{(f, g)}$ and $\mathcal{D}_i = \{P \in \mathcal{P}_i, P \cap D \neq \emptyset\}$. We write $\int_K V_{(F_i, G_i)} - V_{\mathcal{P}_i} \, dm = I_{i,1} + I_{i,2}$, with

$$I_{i,1} = \sum_{P \in \mathcal{D}_i} \int_{P \cap K} V_{(F_i, G_i)} - V_{\mathcal{P}_i} \, dm, \tag{18}$$

$$I_{i,2} = \sum_{P \in \mathcal{P}_i \setminus \mathcal{D}_i} \int_{P \cap K} V_{(F_i, G_i)} - V_{\mathcal{P}_i} \, dm. \tag{19}$$

Because of $V_{(f, g)} \geq V_{(F_i, G_i)}$, the assumption that D has zero Lebesgue measure and $H(\rho_i(x), \{x\}) \rightarrow 0$, we have that $I_{i,1} \rightarrow 0$ for $i \rightarrow \infty$. Using Proposition 7.1, the compactness of K , and the fact that $V_{(F_i, G_i)}|_K \rightarrow V_{(f, g)}|_K$ uniformly, we also obtain that $I_{i,2} \rightarrow 0$ as $i \rightarrow \infty$, i.e. $V_{(F_i, G_i)}|_K - V_{\mathcal{P}_i}|_K \rightarrow 0$ in L^1 and thus the assertion of the theorem. □

Corollary 7.1 *Under the assumptions of Theorem 7.1, we have $V_{\mathcal{P}_i}(x) \rightarrow V_{(f, g)}(x)$ as $i \rightarrow \infty$ for Lebesgue-almost all $x \in K$, where K is any compact subset of the domain of $V_{(f, g)}$.*

Proof By standard arguments, there exists a subsequence $(i(j))_j$ such that $V_{\mathcal{P}_{i(j)}}(x) \rightarrow V_{(f,g)}(x)$ as $j \rightarrow \infty$ for Lebesgue-almost all $x \in K$. Since $(V_{\mathcal{P}_i(x)})_i$ is monotone, we obtain the assertion. \square

Remark 7.1 Using Proposition 4.1 instead of Proposition 3.2, it is easily seen that our convergence results remain valid in case of state space constraints if we assume condition (9) for $\bar{F}(x, u, w) = \{f(x, u, w)\}$. In this case, the first assertion of Theorem 7.1 will hold for the p -norm from (9) instead of the ∞ -norm.

8 Feedback Construction

As usual, we use the approximate optimal value function $V_{\mathcal{P}}$ and the optimality principle (4) in order to construct an approximate optimal feedback. More precisely, for any point $x \in S_0$, $S_0 := \{x \in X : V_{(f,g)}(x) < \infty\}$, we define

$$u_{\mathcal{P}}(x) = \operatorname{argmin}_{u \in U} \max_{w \in W} \{g(x, u) + V_{\mathcal{P}}(f(x, u, w))\}.$$

We can adapt immediately Theorem 3 from [2] in order to obtain a statement about the performance of this feedback. In particular, one can show that the feedback is robust with respect to arbitrary perturbations of the system. For details, we refer to [6] again.

Remark 8.1 A particular application of our result is to robustify the feedback construction from [2] with respect to small perturbations. For this purpose, a particularly convenient way is to consider an “ ε -inflated” system related to the original unperturbed system. More precisely, given an unperturbed control system $f : X \times U \rightarrow X$, one considers the perturbed system $x_{k+1} = f(x_k, u_k) + \varepsilon w_k, k = 0, 1, \dots$, with $w_k \in [-1, 1]^d$ for some (small) $\varepsilon > 0$. In the numerical realization, the sets $F(x, u, W) = f(x, u) + \varepsilon[-1, 1]^d$ are easy to construct using ideas from rigorous discretization, see [13, 14].

9 Dijkstra’s Method

Let (\mathcal{P}, E) be a finite directed graph with edge weights $g : E \rightarrow [0, \infty)$. Let $D \in \mathcal{P}$ be the *destination node*. The following algorithm [9] computes the length $V(P) \in [0, \infty)$ of the shortest path from P to D for all nodes $P \in \mathcal{P}$.

Algorithm 9.1 DIJKSTRA $((\mathcal{P}, E), g, D)$

For each $P \in \mathcal{P}$, set $V(P) := \infty$.

Set $V(D) := 0$.

Set $\mathcal{Q} := \mathcal{P}$.

While $\mathcal{Q} \neq \emptyset$, do:

Set $P := \operatorname{argmin}_{P' \in \mathcal{Q}} V(P')$.

Set $\mathcal{Q} := \mathcal{Q} \setminus \{P\}$.

For each $Q \in \mathcal{P}$ with $(Q, P) \in E$, do:
 if $V(Q) > g(Q, P) + V(P)$, then
 set $V(Q) := g(Q, P) + V(P)$.

The following proposition follows immediately from the construction of the algorithm and the fact that the edge weights are nonnegative.

Proposition 9.1 *During the while-loop in lines 4–9 of Algorithm 9.1 it holds that*

$$V(P) \geq V(P'), \quad \text{for all } P' \in \mathcal{P} \setminus \mathcal{Q}.$$

Acknowledgements We thank Marcus von Lossow for helpful comments on the complexity analysis.

References

1. Junge, O., Osinga, H.M.: A set oriented approach to global optimal control. *ESAIM: Control Optim. Calc. Var.* **10**(2), 259–270 (2004)
2. Grüne, L., Junge, O.: A set oriented approach to optimal feedback stabilization. *Syst. Control Lett.* **54**(2), 169–180 (2005)
3. Grüne, L.: An Adaptive Grid scheme for the discrete Hamilton-Jacobi-Bellman equation. *Numer. Math.* **75**(3), 319–337 (1997)
4. Nešić, D., Teel, A.R.: A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models. *IEEE Trans. Autom. Control* **49**(7), 1103–1122 (2004)
5. Huang, S., James, M.R., Nešić, D., Dower, P.M.: A unified approach to controller design for achieving ISS and related properties. *IEEE Trans. Autom. Control* **50**(11), 1681–1697 (2005)
6. Grüne, L., Junge, O.: Global optimal control of perturbed systems. Technical Report, arXived at math.OC/0703874 (2006)
7. Fleming, W.: The convergence problem for differential games. *J. Math. Anal. Appl.* **3**, 102–116 (1961)
8. Bertsekas, D.P.: *Dynamic Programming and Optimal Control*, vol. 2. Athena Scientific, Belmont (1995)
9. Dijkstra, E.: A note on two problems in connection with graphs. *Numer. Math.* **5**, 269–271 (1959)
10. Gallo, G., Longo, G., Nguyen, S., Pallottino, S.: Directed hypergraphs and applications. *Discrete Appl. Math.* **40**, 177–201 (1992)
11. Dellnitz, M., Hohmann, A.: A subdivision algorithm for the computation of unstable manifolds and global attractors. *Numer. Math.* **75**(3), 293–317 (1997)
12. von Lossow, M.: A min-man version of Dijkstra’s algorithm with application to perturbed optimal control problems. In: *Proceedings of the GAMM Annual Meeting, Zürich, Switzerland, 2007* (to appear)
13. Junge, O.: Rigorous discretization of subdivision techniques. In: Fiedler, B., Gröger, K., Sprekels, J. (eds.) *Proceedings of EQUADIFF 99*, pp. 916–918. World Scientific, Singapore (2000)
14. Grüne, L.: *Asymptotic Behavior of Dynamical and Control Systems under Perturbation and Discretization*. Lecture Notes in Mathematics, vol. 1783. Springer, New York (2002)