

A Dynamic Programming Approach to Viability Problems

Pierre-Arnaud Coquelin, Sophie Martin, Rémi Munos

▶ **To cite this version:**

Pierre-Arnaud Coquelin, Sophie Martin, Rémi Munos. A Dynamic Programming Approach to Viability Problems. IEEE ADPRL, IEEE Computational Intelligence Society, Apr 2007, Hawai, United States. pp.178-184. inria-00125423

HAL Id: inria-00125423
https://inria.hal.science/inria-00125423

Submitted on 19 Jan 2007

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.

A Dynamic Programming Approach to Viability Problems

Pierre-Arnaud Coquelin
Centre de Mathématiques Appliquées
Ecole Polytechnique,
1128 Palaiseau cedex, France
Email: coquelin@cmapx.polytecnique.fr

Sophie Martin
Laboratoire d'Ingénierie pour les Systémes Complexes,
Cemagref de Clermont-Ferrand
63172 Aubière cedex, France
Email: sophie.martin@clermont.cemagref.fr

Rémi Munos
INRIA Futurs
Université Lille 3, France
Email: remi.munos@inria.fr

***Abstract*— Viability theory considers the problem of maintaining a system under a set of viability constraints. The main tool for solving viability problems lies in the construction of the *viability kernel*, defined as the set of initial states from which there exists a trajectory that remains in the set of constraints indefinitely. The theory is very elegant and appears naturally in many applications. Unfortunately, the current numerical approaches suffer from low computational efficiency, which limits the potential range of applications of this domain.**

In this paper we show that the viability kernel is the zero-level set of a related dynamic programming problem, which opens promising research directions for numerical approximation of the viability kernel using tools from approximate dynamic programming. We illustrate the approach using $k-$nearest neighbors on a toy problem in two dimensions and on a complex dynamical model for anaerobic digestion process in four dimensions.

I. Introduction

In a typical viability problem, the state of a controlled dynamical system is submitted to viability constraints. Solving the problem means discriminating the states from which there exists at least one control law such that the system remains in the constraint set indefinitely (these state define the *viability kernel*), from the states from which the system is doomed to go beyond the viability constraints in finite time.

To solve such problems, viability theory has been designed and developed for about twenty years [1]. It encompasses a great deal of theoretical work and provides efficient mathematical tools for analyzing the local and asymptotic behavior of control systems. It is also used to study the evolutions of uncertain systems subjected to viability constraints arising in socio-economic and biological sciences, as well as in control theory, since their expression is often natural under this formalism (see for instance [4] for the management of a renewable resource, [14] for the analysis of fishing quotas consequences, and [13] for a lake eutrophication problem in ecology).

Formally, a typical **viability problem** is defined as follows. Consider a control system, defined by the differential equation:

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)),\ u(t) \in U \text{ for almost every } t \geq 0 \\ x(0) = x_0 \end{cases} \quad (1)$$

where $f : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^d$ is the *state dynamics* and U a compact subset of $\mathbb{R}^p$, the *control space*. We denote by F the set-valued map defined by $F(x) := \{f(x,u), u \in U\}$, for every $x \in \mathbb{R}^d$.

A viability problem considers evolutions $x(\cdot)$ that are viable in a subset $K \subset \mathbb{R}^d$ representing an environment in which the trajectory of the evolution must remain forever: $\forall t \geq 0, \quad x(t) \in K$.

The **Viability Kernel** of the subset K under the dynamics F, denoted by $\text{Viab}_F(K)$, is defined as the set of points x_0 from which can start a viable solution, i.e.,

$$\begin{aligned} \text{Viab}_F(K) \quad := \quad & \{x_0 \in K,\ \exists u(\cdot) \in L^\infty((0,+\infty);U),\ \text{such} \\ & \text{that the solution of (1) associated to } u(\cdot) \\ & \text{satisfies: } \forall t \geq 0, x(t) \in K\}\,. \end{aligned} \quad (2)$$

The viability kernel may be characterized in diverse ways through tangential conditions thanks to the viability theorems [1]. The computation of the viability kernel also serves for designing interesting feed-back policies, such as the *heavy controller* which remains constant as long as possible until the boundary of the viability kernel is reached.

There are several numerical approaches for approximating the viability kernel. The *viability kernel algorithm*, proposed by Saint-Pierre [15], computes, for a given discretization step Δh of the space and time, a grid-based representation of the viability kernel $\text{Viab}_F^{\Delta h}(K)$ such that: $\lim_{\Delta h \to 0} \text{Viab}_F^{\Delta h}(K) = \text{Viab}_F(K)$. Bokanowski et al. [6] characterized the viability kernel by the infinite time limit of the value function of an evolutionary control problem and this value function being discontinuous, used the anti-diffusive *Ultra-Bee* numerical scheme extended to the resolution of Hamilton-Jacobi-Bellman equations to reduce numerical diffusion.

These approaches share common features that limit their applicability to real world problems: the discretization in space is grid-based and therefore subject to the curse of dimensionality: the computational time and memory needed to find and store $\text{Viab}'(K)$ such that $d(\text{Viab}'(K), \text{Viab}(K)) \leq \epsilon$ grow exponentially with the dimension of underlying state space. Consequently these algorithms do not work in dimensions higher than 3. More efficient subset representation is considered by Chapel et al. [7] where Support Vector Machines summarize the information contained in the grid and provide an analytical description of the viability kernel.

In this paper we address the issue of efficiently computing the viability kernel using approximate dynamic programming (see eg. [8]) methods. We propose an original approach that proceeds in three steps. First, we characterize $\text{Viab}_F(K)$ as the zeros of a $[0,1]$-valued map $V^*(x)$ geometrically decreasing with the maximal exit time of K when starting at $x \in K$. Second, we show that V^* solves the Hamilton-Jacobi-Bellman equation, that we discretize in time using a finite time-step Δt, and use an approximate dynamic programming algorithm to approximate the solution $V^{\Delta t}$ of the resulting Bellman equation. Third, we define the approximate viability kernel associated to the function $V^{\Delta t}$ by the under-level set $K_\epsilon := \{x \in K, V^{\Delta t}(x) \leq \epsilon\}$ with $\epsilon > 0$ a given threshold.

A contribution of this work is to establish a link between the viability kernel and the under-level set of an approximate solution of a related dynamic programming (DP) problem. Our purpose in this paper is to describe how one may use approximate dynamic programming for numerically solving viability problems. In particular, we believe this would enable to introduce machine learning tools such as statistical learning (see e.g. [12]) and function approximation (see e.g. [9]) into the viability community. Another contribution is the viability study of the anaerobic digestion process model described in [11]. We illustrate the fact that the computation of the viability kernel provides a very natural yet powerful tool to analyze, understand, and control the complex behavior of this chemical process.

The paper is organized as follows. Section II establishes the connection between the viability kernel and the zero-level set of the value function for a related DP problem. We also mention a convergence result for the time-discretized problem. In Section III, we describe the numerical method for solving the discrete-time DP problem and illustrate the method using Approximate Value Iteration combined with $k-$nearest neighbors. We recall an error bound for this ADP method in terms of L^∞ norm. In Section IV we bound the approximation error of the viability kernel in terms of the approximation power of the function space. In Section V we present the numerical examples in two and four dimensions that illustrate the approach.

II. A Dynamic Programming approach to solving Viability problems

In this section we establish the link between the viability kernel and the zero level set of the value function of a related Dynamic Programming (DP) problem. We then introduce a consistent time-discretization procedure whose solution will be numerically approximated in the next section.

First, let us define $\tau : \mathbb{R}^n \times L^\infty(\mathbb{R}_+; U) \to \overline{\mathbb{R}}_+$ as the exit time from K of the trajectory $x(\cdot)$ (defined by (1)), starting from x_0 under the control law $u(\cdot)$: $\tau(x_0; u(\cdot)) := \inf\{t \in \mathbb{R}_+ | x(t) \notin K\}$ with the convention that $\tau = \infty$ if the trajectory stays forever in K. Then, the maximum value $\tau^* : \mathbb{R}^n \to \overline{\mathbb{R}}_+$ of the exit time, for all possible controls, is

$$\tau^*(x_0) = \sup_{u(\cdot) \in L^\infty(\mathbb{R}_+; U)} \tau(x_0; u(\cdot)) \, .$$

From its definition, the viability kernel is the set of points with infinite exit-time value:

$$\text{Viab}_F(K) = \{x \in K, \tau^*(x) = +\infty\}.$$

We now introduce the related discounted optimal exit time control problem. The discount factor is denoted by $\gamma \in (0,1)$ and the optimal value function $V^* : \mathbb{R}^n \to \mathbb{R}$ is defined by:

$$V^*(x) := \inf_{u(\cdot) \in L^\infty(\mathbb{R}_+; U)} [\gamma^{\tau(x, u(\cdot))}] = \gamma^{\tau^*(x)},$$

which is 1 discounted by γ to the power the maximum exit time.

It is straightforward to check that the viability kernel is the zero level set of V^*: $\text{Viab}_F(K) = \{x \in K, V^*(x) = 0\}$. Consequently, computing the value function V^* enables to deduce the viability kernel. This is the basic idea of the dynamic programming approach to compute viability kernels.

The optimal value function solves a non-linear partial differential equation, called the Hamilton-Jacobi-Bellman (HJB) equation in the sense of viscosity solutions (see eg. [10]): if V^* is differentiable at $x \in K$, then

$$V^*(x) \log(\gamma) + \min_{u \in U} \left[\nabla V^*(x) \cdot f(x,u)\right] = 0, \qquad (3)$$

with the (boundary) condition $V^*(x) = 1$ for $x \notin K$.

The analysis of such equations and the formalism of viscosity solutions is beyond the scope of this paper; we refer the interested reader to the references [10], [2].

Now, we discretize in time the HJB-equation using some consistent and monotonic numerical scheme. Let Δt be the time-step, and write $V^{\Delta t}$ the approximate value function, which is the unique solution of the following dynamic programming equation: for $x \in K$,

$$V^{\Delta t}(x) = \gamma^{\Delta t} \min_{u \in U} V^{\Delta t}(g^{\Delta t}(x,u)) \qquad (4)$$

where $g^{\Delta t}(x,u) := x + f(x,u)\Delta t$ is the (Euler-based) discretized state-dynamics. The boundary condition is the same: $V^{\Delta t}(x) = 1$ for $x \notin K$.

The consistency of this numerical scheme holds since a regular solution V of (3) plugged in the scheme (4) would yield a truncation error of order 1 in time, i.e.

$$\begin{aligned} & \min_{u \in U} \frac{\gamma^{\Delta t} V(x + f(x,u)\Delta t) - V(x)}{\Delta t} \\ = \; & \min_{u \in U} \left[\nabla V(x) \cdot f(x,u)\right] + V(x) \log(\gamma) + O(\Delta t) \\ = \; & O(\Delta t) \end{aligned}$$

Now, in order to guarantee the convergence of the discrete-time value function $V^{\Delta t}$ to the optimal value function V^*, when Δt tends to zero, we make use of the general results of [3], where the comparison principle (between lower and upper viscosity solutions) is a consequence of the following assumption on the state dynamics at the boundary:

Assumption 1: Note $\overrightarrow{n}(x)$ the outward normal of K at $x \in \partial K$. We assume that for all $x \in \partial K$,

if $\exists u \in U$, s.t. $f(x,u) \cdot \overrightarrow{n}(x) \leq 0$, then $\exists u \in U$, s.t. $f(x,u) \cdot \overrightarrow{n}(x) < 0$,

if $\exists u \in U$, s.t. $f(x,u) \cdot \overrightarrow{n}(x) \geq 0$, then $\exists u \in U$, s.t. $f(x,u) \cdot \overrightarrow{n}(x) > 0$.

Under this assumption, we have the convergence result (see [3] for a proof): $\lim_{\Delta t \to 0} V^{\Delta t} = V^*$.

It is worth noting that the technical assumption 1 guarantees that the optimal value function V^* is continuous, which simplifies the convergence proof of numerical schemes such as (4). However, in general this assumption is by no means necessary for convergence, but without it, the proof become more tedious and requires the theory of non-continuous viscosity solutions (see eg. [2]).

Let us introduce the viability kernel $\text{Viab}_g^{\Delta t}(K)$ of K under the discrete dynamics $g^{\Delta t}$. It is also straightforward to check that $\text{Viab}_g^{\Delta t}(K)$ is the zero-level set of $V^{\Delta t}$ that is to say: $\text{Viab}_g^{\Delta t}(K) = (V^{\Delta t})^{-1}(0)$. Thus in the sequel, we aim at computing an approximation of the discrete-time solution $V^{\Delta t}$ for a given (small) Δt.

For notation clarity, we omit in the remainder of the paper the Δt exponent, simply writing V for $V^{\Delta t}$, g for $g^{\Delta t}$, $\text{Viab}_g(K)$ for $\text{Viab}_g^{\Delta t}(K)$ and γ for $\gamma^{\Delta t}$.

III. Numerical methods and error bounds for the discrete-time DP problem

We may write the DP equation (4) as a fixed-point equation $V = TV$ where T is the DP operator, defined, for any bounded function W over K and for all $x \in K$, by :

$$TW(x) := \gamma \min_{u \in U} \begin{cases} W(g(x,u)) & \text{if} \quad g(x,u) \in K \\ 1 & \text{if} \quad g(x,u) \notin K \end{cases}$$

Since T is a contraction mapping in $\| \cdot \|_\infty$ norm, it admits a unique fixed-point that can be computed by regular DP methods (see eg. [8]) such as the **value iteration algorithm**:

- Initialization: start with any $V_0 \in \mathbb{R}^K$
- Iteration: repeat $V_{n+1} = TV_n$ until convergence occurs (see above references)

Unfortunately, in continuous state spaces or in large discrete spaces, the value function cannot be computed exactly, and has to be represented using a finite number of parameters. In this paper we investigate function approximators where the values are represented using a finite set of states $X_N := \{x_i\}_{1 \leq i \leq N} \subset K$.

Let us define an **approximation operator** $A_N : \mathbb{R}^K \to \mathbb{R}^K$, based on X_N, such that for any function $W \in \mathbb{R}^K$, $A_N(W)$ is defined only from the N values $\{W(x_i)\}_{1 \leq i \leq N}$. We write $\mathcal{F}_N$ the image of $\mathbb{R}^K$ by A_N, ie: $\mathcal{F}_N := A_N(\mathbb{R}^K)$.

Examples of such operators include $k-$nearest neighbors, locally weighted regression, linear approximation, and kernel methods in reproducing kernel Hilbert spaces, such as SVMs (see eg. [12]). The one-nearest neighbor approximator, defined by $A_N(W)(x) := W(\arg\min_{y \in X_N} \|x - y\|)$ satisfies the uniform approximation bound

$$\|W - A_N(W)\|_\infty \leq \epsilon(\mathcal{F}_N) \tag{5}$$

in the class of $L-$Lipschitz functions on K (ie. functions W such that $\forall x, y \in K, |W(x) - W(y)| \leq L\|x - y\|$). Indeed, for such functions, we can take $\epsilon(\mathcal{F}_N) = Ld(X_N, K)$ where $d(X_N, K) := \sup_{x \in K} \min_{1 \leq i \leq N} \|x - x_i\|$ is the density of X_N in K.

The **Approximate Value Iteration** (AVI) algorithm is defined by (see e.g. [8]):

- Initialization: Sample N states $X_N = \{x_i\}_{1 \leq i \leq N}$ and choose any function $V_0 \in \mathbb{R}^K$.
- Iteration: repeat $V_{n+1} := A_N(TV_n)$ until some stopping criterion is met (see [8]).

Now, assuming that the approximation bound (5) holds for functions in $T\mathcal{F}_N$ (thus, at each iteration, the approximation error satisfies $\|V_{n+1} - TV_n\|_\infty \leq \epsilon(\mathcal{F}_N)$), then a bound on the approximation error of the value function V, in terms of the approximation capacity of $\mathcal{F}_N$ is:

$$\|V - V_n\|_\infty \leq \frac{1}{1-\gamma}\epsilon(\mathcal{F}_N) + \gamma^n \|V - V_0\|_\infty, \tag{6}$$

(which simply results from an inductive argument based on the inequality $\|V - V_n\|_\infty \leq \|TV - TV_{n-1}\|_\infty + \|TV_{n-1} - A_N(TV_{n-1})\|_\infty \leq \gamma\|V - V_{n-1}\|_\infty + \epsilon(\mathcal{F}_N)$).

IV. Approximation of the viability kernel

In this section we explain how to derive an approximate viability kernel by selecting the states whose approximate value function is below a given positive threshold. This thresholding procedure is reminiscent of the two-classes classification problem where a potential function h is trained based on labeled data, and then used for the prediction at a new state x based on $\text{sign}(h(x) - b)$ (where b is the threshold). Viability kernel approximation using sequences of SVM classification problems has been undertaken in [7].

In some way, this paper illustrates that a good choice of potential function h for viability kernel classification is the value function of the related dynamic programming problem introduced in the previous section.

For any function h and real number $\eta > 0$, let us defined the **under-level set** $K_\eta(h) = \{x \in K, h(x) \leq \eta\}$. We have the following result concerning the approximation of the viability kernel K by $K_\eta(\hat{V})$, where $\hat{V}$ is an approximation of the value function.

Theorem 1: Assume that there exists two constants $C > 0$ and $\alpha > 0$ such that for all x in a small neighborhood of $\text{Viab}_g(K)$, $V(x) \geq Cd(x, \text{Viab}_g(K))^\alpha$. Assume that the approximate value function $\hat{V}$ is $\eta-$close to V, i.e. $\|V - \hat{V}\|_\infty \leq \eta$. Then the approximate viability kernel $K_\eta(\hat{V})$ satisfies, for small $\eta > 0$,

$$\text{Viab}_g(K) \subset K_\eta(\hat{V}) \subset K_{2\eta}(V) \subset \text{Viab}_g(K) + \mathcal{B}\big(0, (2\eta/C)^{1/\alpha}\big).$$

where $\mathcal{B}(0, \delta)$ is the (euclidian) ball centered at 0 with radius δ.

The assumption in Theorem 1 expresses a lower bound on the increase of V in the direction of the outward normal of the viability kernel, and is not very stringent in general. A typical value $\alpha = 2$ is usual, although one may build artificial problems for which this increase is exponentially slow (all derivatives being equal to zero).

Proof: This result derives from the definition of the under-level sets. In short, if $x \in \text{Viab}_g(K)$, then $V(x) = 0$ thus $\hat{V}(x) \leq \eta$ thus $x \in K_\eta(\hat{V})$. Now if $x \in K_\eta(\hat{V})$, then $V(x) \leq \hat{V}(x) + \eta \leq 2\eta$, thus $x \in K_{2\eta}(V)$. Finally, if $x \in K_{2\eta}(V)$ then $d(x, \text{Viab}_g(K)) \leq (\frac{2\eta}{C})^{1/\alpha}$ thus (for small η) $x \in \mathcal{B}\big(0, (\frac{2\eta}{C})^{1/\alpha}\big)$. □

As a consequence, one may derive a convergence result to the viability kernel of the continuous time problem $\text{Viab}_F(K)$ (defined by (2)).

Corollary *Let V_N be a sequence of approximate value functions computed by the method of Section III, using a decreasing time-step $\Delta t(N)$, such that the approximation error satisfies $\epsilon(\mathcal{F}_N)/\Delta t(N) \to 0$ when $N \to \infty$ (which is the case for $k-$nearest neighbors with uniformly sampled states X_N with a $\Delta t(N)$ decreasing strictly more slowly than $N^{-1/d}$, where d is the space dimension), then*

$$\lim_{N\to\infty} K_{1/N}(V_N) = \mathit{Viab}_F(K).$$

This consistency result is a direct consequence of Theorem 1 and guarantees that for a rich enough class of function approximators, the approximate viability kernel defined by the under-level set of the resulting approximate value function is close to the true viability kernel.

Proof: From the hypothesis, for any N, there exists $\epsilon(\mathcal{F}_N)$ and $\Delta t(N)$ such that $\epsilon(\mathcal{F}_N)/\Delta t(N) \leq 1/N$. Now, from (6), $\|V^* - V_N\|_\infty \leq 2\epsilon(\mathcal{F}_N)/(1-\gamma^{\Delta t(N)})$ for large enough N, thus $\|V^* - V_N\|_\infty \leq 4\epsilon(\mathcal{F}_N)/\Delta t(N) \leq 4/N$, and the convergence follows as a direct consequence of Theorem 1. In the case of $k-$nearest neighbors with uniformly sampled states, the density of points ($\epsilon(\mathcal{F}_N)$ as mentioned in the previous section) is of order $1/N^{1/d}$. Thus if $\Delta t(N)$ decreases strictly more slowly than $N^{-1/d}$, then $\epsilon(\mathcal{F}_N)/\Delta t(N) \to 0$ for $N \to \infty$. □

V. Simulations

A. A two-dimensional consumption problem

We illustrate the method on this two-dimensional consumption problem (see [1], [15]), defined by the state dynamics

$$\left\{ \begin{array}{lcl} \dot{x}(t) & = & x(t) - y(t) \\ \dot{y}(t) & = & u \end{array} \right. \tag{7}$$

with the control $u \in U := \{-\frac{1}{2}, \frac{1}{2}\}$, and the constraints $x(t) \in [0, 2]$ and $y(t) \in [0, 3]$.

Figure 1 represents (left) the approximate value function V_n (here $n = 200$) obtained for a set X_N of $N = 10^5$ points sampled uniformly over the domain $K = [0, 2] \times [0, 3]$. The discount factor $\gamma^{\Delta t} = 0.95$ (with $\Delta t = 0.05$). The approximator used here is the $3-$nearest neighbors. The right figure shows the set of states $\{x_i, V_n(x_i) \leq 0.01\}$ being an approximation of the viability kernel.

Viability kernels make it possible to define interesting control policies. Let us simply illustrate the notion of *timorous controller*, which is a secure variant of the so-called *heavy controller* (see eg. [1]) where the control is kept constant until the state is below a fixed (security) distance to the boundary of the viability kernel. When this happens, the chosen control is the one maximizing the distance from the next resulting states to the boundary. This timorous controller provides a robust, secure, and piecewise constant controller. A piece of trajectory following a timorous controller (with a security distance of 0.05) is shown in black line in Figure 1 (right).

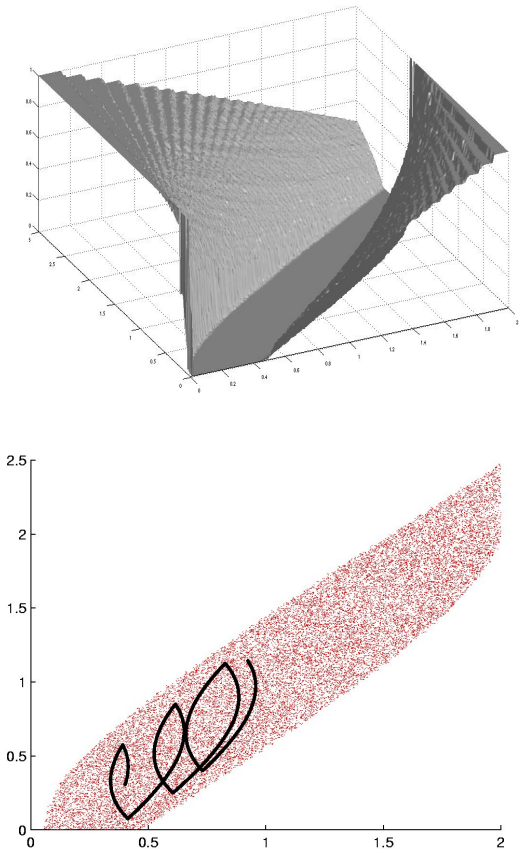


Fig. 1. The consumption problem. Top: value function of the related DP problem. The variables x and y lie in the (x, y)-plane, and the the value of the value function for this problem is indicted in the z-axis. Bottom: set of points $\{x_i\}$ in the approximate kernel (such that $V_n(x_i) \leq 0.01$) and a timorous trajectory starting from $x_0 = 0.4$, $y_0 = 0.3$ and using a security distance of 0.05.)

B. A four dimensional anaerobic digestion model

Anaerobic wastewater treatment process is a powerful and ecological process to degrade organic substrate. However, its instability prevents it to be used on a wide scale industrially. Consequently, during the last decades there has been a growing interest in modeling and studying its dynamical behavior [11], [5], aiming at building stable controllers [16].

Anaerobic digestion can be viewed as a two stage chemical process:

$$\begin{array}{rcl} \text{Acidogenesis: } k_1 S_1 & \stackrel{\mu_1(S_1)X_1}{\longrightarrow} & X_1 + k_2 S_2 + k_4 CO_2 \\ \text{Methanogenesis: } k_3 S_2 & \stackrel{\mu_2(S_2)X_2}{\longrightarrow} & k_5 CO_2 + k_6 CH_4 \end{array}$$

where $k_1, \ldots, k_6$ are the stoechiometric constants and S_1, S_2, X_1, X_2 are respectively the concentration of organic substrate, of volatile fatty acid, of acidogenesis bacteria, of methanogenesis bacteria in the tank. The reaction rates $\mu_1(\cdot)$ and $\mu_2(\cdot)$ are given by:

$$\begin{aligned}\mu_1(S_1) &= \mu_{1max}\frac{S_1}{S_1+K_{S1}}\\ \mu_2(S_2) &= \mu_{2max}\frac{S_2}{S_2+K_{S2}+S_2^2/K_{I2}}.\end{aligned}$$

where μ_{1max}, μ_{2max}, K_{S1}, K_{S2}, and K_{I2} are constants. By a mass balance calculation, one may derive the following 4-dimensional dynamical model:

$$\begin{aligned}\dot{X}_1 &= (\mu_1(S_1)-\alpha D)X_1,\\ \dot{S}_1 &= D(S_{1in}-S_1)-k_1\mu_1(S_1)X_1,\\ \dot{X}_2 &= (\mu_2(S_2)-\alpha D)X_2,\\ \dot{S}_2 &= D(S_{2in}-S_2)-k_2\mu_1(S_1)X_1-k_3\mu_2(S_2)X_2,\end{aligned}$$

where α is the rate of bacteria in suspension in the liquid phase, S_{1in}, S_{2in} are the effluent characteristics. $D \in \{0, 0.5\}$ is the liquid flow per volume unit and represents the control parameter (which takes two possible values). Since those dynamics modelize an unstable phenomenon, the knowledge of viability states is of particular interest. The main cause of instability is the accumulation of volatile fatty acid S_2 in the tank which inhibits the methanogenesis phase, and then produces even more S_2 which eventually acidifies the tank and kills all the bacteriological species. Thus, the natural constraint set is given by all variables being non-negative and S_2 is additionally required to be upper bounded, thus: $(X_1, X_2, S_1, S_2) \in (\mathbb{R}_+)^3 \times [0, 10]$.

We compute the four dimensional viability kernel where we chose the same numerical values as in [5]. For the algorithm parameters, we took $\Delta_t = 0.005$, $\gamma^{\Delta t} = 0.999$. We run $n = 500$ value iterations using a $3-$nearest neighbor approximator using 2.10^5 points uniformly sampled in the domain $K = \{(X_1, X_2, S_1, S_2) \in [0, 2] \times [0, 0.2] \times [0, 3] \times [0, 10]\}$. The full computation of the viability kernel takes 162.52 sec. with Matlab on a 1.8 Ghz computer. Projections of the viability kernel onto two dimensional spaces are represented in Figures 2,3, and 4. For clarity, we plotted the set of points located on a regular grid whose nearest neighbor belongs to the under-level set of the approximate value function (with a threshold of 0.001).

We believe that the results are very interesting and provide an original visual representation of the stability problem for an anaerobic digestion. We observe three kinds of dependencies between variables directly from the geometry of the viability kernel boundary: convex dependency (Figure 2), linear dependency (Figure 3), and independency (Figure 4). The results are in accordance with the qualitative interpretation of the chemical phenomenon. For example, Figure 2 illustrates the fact that we cannot have at the same time a large amount of organic substrate S_1 and acidogenesis bacteria X_1. Indeed this would induce a fast production of volatile fatty acid S_2 which would lead to the reaction death. The other figures also illustrate in a simple and intuitive way similar relationships between variables.

Moreover, the viability kernel contains quantitative relevant knowledge that can be exploited, for example, to design a timorous controller (as defined in previous section) which is particularly adapted here since it provides a safe controller with infrequent switches.

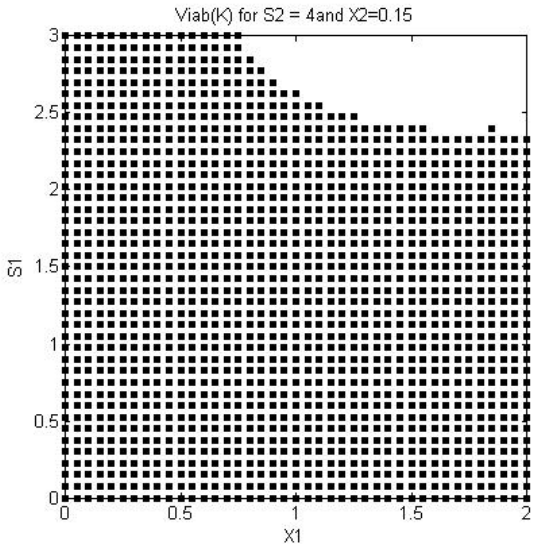
VI. Conclusion

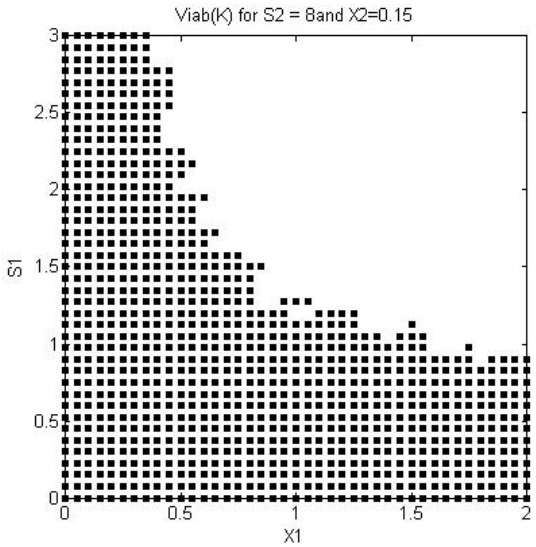
This paper describes an alternative approach for solving viability problems by introducing a related exit-time discounted optimal control problem. We showed how the viability kernel is approximated by the under-level set of an approximate value function for some small positive threshold, and provide approximation error based on the capacity of the considered function space. To illustrate, we used a simple $k-$nearest neighbors approximator with states sampled uniformly randomly inside the set of constraints K.

Of course, more sophisticated function approximators, and other approximate DP algorithms (such as policy iteration based) could be used to improve performance. Besides, adaptive methods could easily be designed from this work. Indeed, once a rough approximate kernel $K_\eta(V_N)$ has been obtained, one may build a new problem by redefining the constraint set to be $K_\eta(V_N)$ instead of K. As long as $\text{Viab}_g(K) \subset K_\eta(V_N)$ (which is the case if $\eta \geq \epsilon(\mathcal{F}_N)$), then we still have $\text{Viab}_g(K) = \text{Viab}_g(K_\eta(V_N))$. The benefit being that since $K_\eta(V_N)$ is smaller than K, sampling N states in this subset would yield a higher density of points and thus would decrease $\epsilon(\mathcal{F}_N)$.

Another way of decreasing $\epsilon(\mathcal{F}_N)$ is to remove the states that are strictly inside the current approximation of the viability kernel (e.g. that lie at a lower bounded security distance from the boundary). The main idea being that we only need to sample points around the boundary of the viability kernel.

Extension of this approach to other viability problems, such as approximating the capture basin is straightforward.





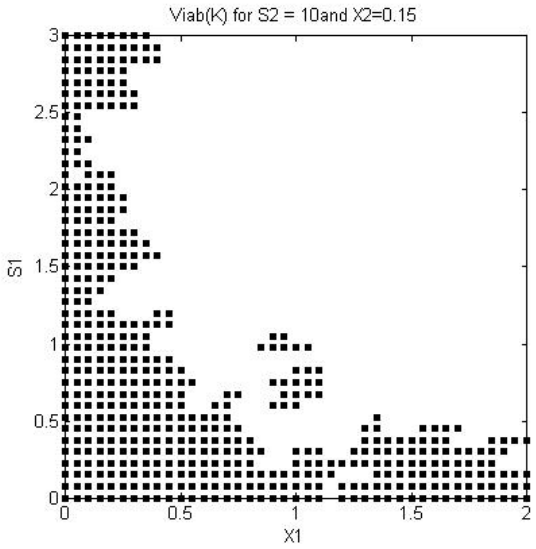
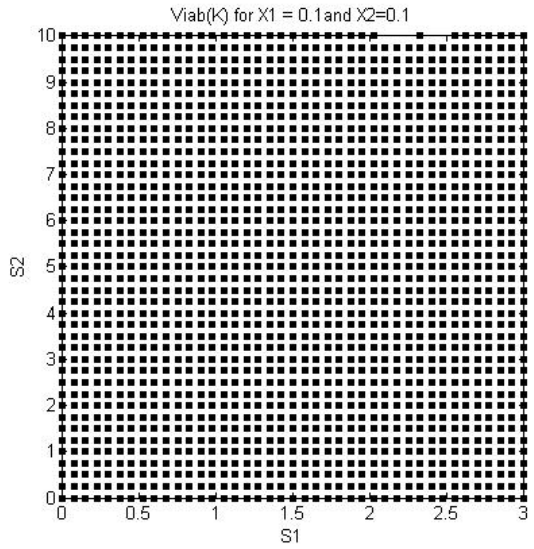
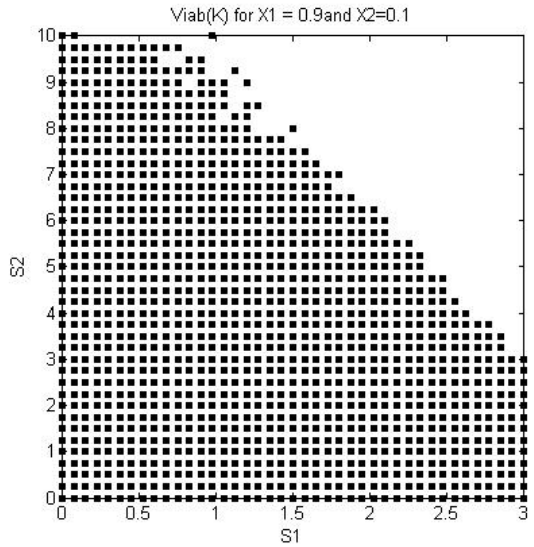


Fig. 2. Representation of S_1 versus X_1 for different values of S_2. Notice the convex dependency between the variables.





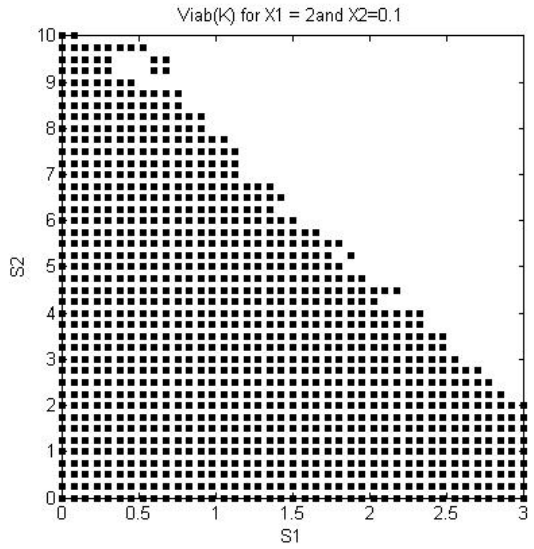


Fig. 3. Representation of S_2 versus S_1 for different values of X_1. Notice the linear dependency between the variables.

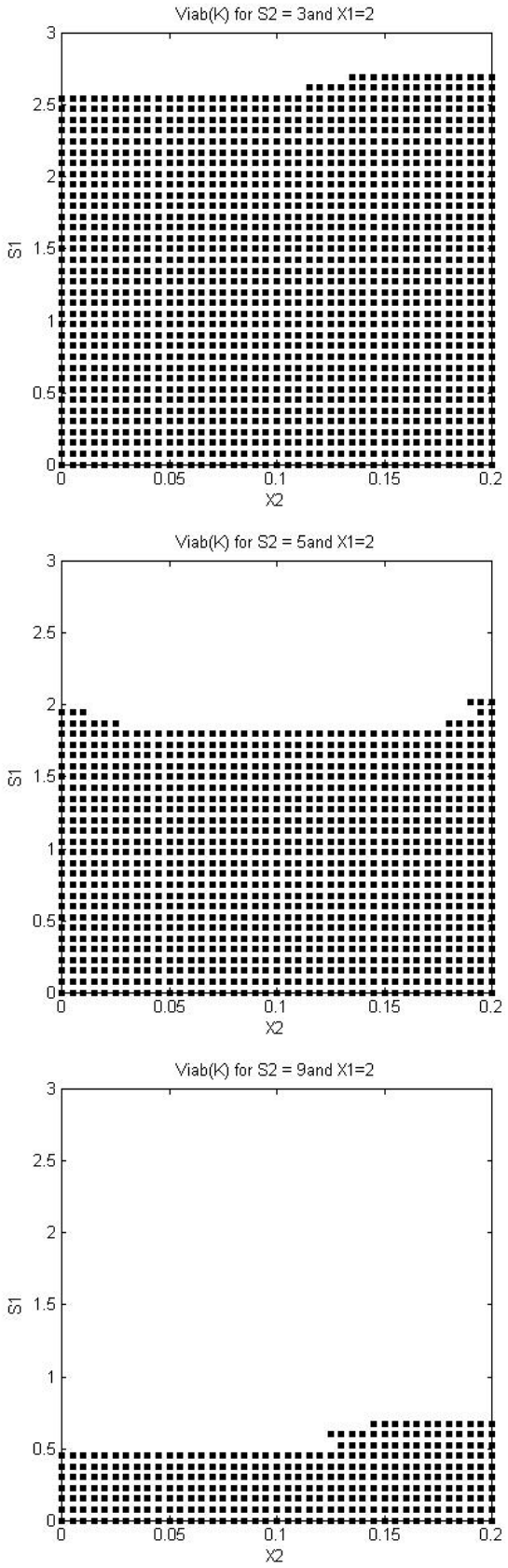


Fig. 4. Representation of S_1 versus X_2 for different values of S_2. Notice the independency of S_1 w.r.t. X_2.

REFERENCES

[1] J.P. Aubin. *Viability theory*. Birkhäuser, 1991.

[2] Martino Bardi and Italo Capuzzo-Dolcetta. *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. Birkhauser Boston, 1997.

[3] Guy Barles and P.E. Souganidis. Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Analysis*, 4:271–283, 1991.

[4] C. Béné, L. Doyen, and D. Gabay. A viability analysis for a bio-economic model. *Ecological Economics*, 36:385–396, 2001.

[5] O. Bernard, Z. Hadj-Sadok, D. Dochain, A. Genovesi, and J.-P. Steyer. Dynamical model development and parameter identification for an anaerobic wastewater treatment process. *Biotech.Bioeng.*, vol. 75:424–438, 2001.

[6] O. Bokanowski, S. Martin, R. Munos, and H. Zidani. An anti-diffusive scheme for viability problems. *Applied Mathematics and Numerics*, 2006.

[7] L. Chapel, G. Deffuant, S. Martin, and C. Mullon. Defining yield policies in a viability approach. *In Proceedings of the V th European Conference on Ecological Modelling (ECEM-2005)*, Pushshino, Russia, 2005.

[8] D.Bertsekas. *Dynamic Programming and Optimal Control, Vol. 1 and 2*. Athena Scientific, Nashua, 2005.

[9] R. DeVore. *Nonlinear Approximation*. Acta Numerica, 1997.

[10] Wendell H. Fleming and H. Mete Soner. *Controlled Markov Processes and Viscosity Solutions*. Applications of Mathematics. Springer-Verlag, 1993.

[11] IWA Task Group for Mathematical Modelling of Anaerobic Digestion Processe. *Anaerobic Digestion Model No.1 (ADM1)*. IWA, 2002.

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics, 2001.

[13] S. Martin. The cost of restoration as a way of defining resilience: a viability approach applied to a model of lake eutrophication. *Ecology and Society*, 9(2):URL: http://www.ecologyandsociety.org/vol8/iss2/art8, 2004.

[14] C. Mullon, P. Curry, and L. Shannon. Viability model of trophic interctions in marine ecosystems. *Natural Ressource Modelling*, 17:2758, 2004.

[15] P. Saint-Pierre. Approximation of viability kernel. *Appl. Math. Optim.*, 29:187–209, 1994.

[16] J-P. Steyer, J-C. Bouvier, T. Conte, P. Gras, and P. Sousbie. Evaluation of a four year experience with a fully instrumented anaerobic digestion process. *Water Science and Technology*, 45:495–502, 2002.