Scientific Research

Computing Reachable Sets as Capture-Viability Kernels in Reverse Time

Noël Bonneuil[1,2]
[1]National Institute of Demographic Studies, Paris, France
[2]Advanced Studies in Social Sciences, Paris, France
Email: bonneuil@ined.fr

ABSTRACT

The set $S_F(x_0;T)$ of states y reachable from a given state x_0 at time T under a set-valued dynamic $x'(t) \in F(x(t))$ and under constraints $x(t) \in K$ where K is a closed set, is also the capture-viability kernel $\mathrm{Capt}^T_{-F}(K, \{x_0\})$ of x_0 at T in reverse time of the target $\{x_0\}$ while remaining in K. In dimension up to three, Saint-Pierre's viability algorithm is well-adapted; for higher dimensions, Bonneuil's viability algorithm is better suited. It is used on a large-dimensional example.

Keywords: Set-Valued Analysis; Reachable Set

1. Introduction

Reachable sets gather the possible future states that a controlled set-valued system can take. The usual method for computing relies on solving HJB. For example, "project grid points from an equidistant grid onto the reachable set;" and "each projection requires to solve an optimal control problem" [1]. This method requires a huge computation, and is often limited to three dimensions. References of other clever methods are in [2,3].

I shall compute reachable sets in identifying them to capture basins. As such, they can be computed by viability algorithms, which do not need to solve HJB. Beside this advantage of avoiding the roundabout of HJB, the viability approach to reachable sets, from the very start, deals with non linear dynamics. Reachable sets appear as a straightforward consequence of set-valued analysis. At last, reachable sets can have a large state dimension.

Viability algorithms are normally used to compute the viability kernel of a closed set K under a dynamic F. The viability kernel is the largest set of initial states x, from which at least one solution to the dynamic F remains in a closed set K. Moreover, the computation of reachable sets impinges on the dimension of the dynamical system: after the dimension three, the computation becomes rapidly intractable. I shall show that Bonneuil's viability algorithm [4], which neither requires solving HJB nor relies on a grid, but uses stochastic optimization, overcomes the curse of dimensionality and provides sets of reachable states. To do this, I shall high-light the fact that reachable sets are also capture basins of a convenient augmented dynamic. After defining reachable sets and capture-viability kernels, I shall identify reachable sets to capture-viability kernels under the dynamic in reverse time. Then, after presenting viability algorithms, I shall compute the reachable sets to the three noteworthy non-linear cases of [1] and to a 10-dimensional case.

2. The Reachable Set as a Capture Basin

Consider a finite-dimensional vector space X and the dynamic:

$$x'(t) \in F(x(t)) = \left\{ f(x(t), u(t)), u(t) \in U \right\} \forall t \geq 0, \quad (1)$$

where F is a Marchaud map. A map $F: X \to K$ is Marchaud if and only if:

Assumption 2.1 (i) the graph and the domain of F are closed and not empty,

(ii) the values $F(x)$ are convex,

(iii) $\exists c \in \mathrm{R}$ such that $\sup_{y \in F(x)} \|y\| < c(\|x\| + 1)$,

Assumptions (i) and (ii) hold true if the continuous-time control set U is a nonempty convex compact subset of a metric space and the function $f: R^m \times U \to R^m$ is Lipschitz with respect to each variable, and affine with respect to u.

The state space is $X = R^p$, the control space R^q, and $T > 0$ the time horizon. I denote $S(x)$ the set of all solutions to Equation (1) starting from a given state x.

Definition 2.1 *A reachable set from a set D at date T under the dynamic F within a closed set K is the set of the states* $x_T \in K$ *for which there exists an initial state* $x_0 \in D$ *and a solution* $x(.) \in S_F(x_0;T)$ *such that* $x(T) = x_T$. *It is denoted*

$$\text{Reach}_F(D, K, T)$$
$$:= \{x_T \in X / \exists x(.) \in S_F(x_0) \cap K : x(T) = x_T\} \quad (2)$$

The reachable set in the interval [0; T] is defined as:

$$\text{Reach}_F(D, K, [0, T])$$
$$:= \cup_{t \in [0,T]} \text{Reach}_F(D, K, t) \quad (3)$$

A state x_0 is said to be viable in K under F if there exists at least one solution $x(t)$ of Equation (1) starting from $x(0) = x_0$ and remaining in K until T. A set of viable states is called a viability domain, and [5] showed that there exists a maximal viability domain including all others. This set is the viability kernel denoted $\text{Viab}_F(K)$ (which is then a set of initial conditions):

$$\text{Viab}_F^T(K) := \{x_0 / \exists x(.), x(0) = x_0 \text{ and} $$
$$\forall t \in [0, T], x'(t) \in F(x(t)), x(t) \in K\}. \quad (4)$$

Definition 2.2 *A capture-viability domain of a set C viable in the set K under the dynamic F is a subset of initial states* $x \in K$ *from which at least one solution viable in K starts until it reaches the target C at time T.*

When F is Marchaud, there exists one largest capture-viability domain with target C including all others [5]. It is denoted:

$$\text{Capt}_F^T(K, C) := \{x_0 / \exists x(.), x(0) = x_0 \text{ and} $$
$$\forall t \in [0, T], x'(t) \in F(x(t)), x(t) \in K, x(T) \in C\} \quad (5)$$

and called capture-viability kernel of the target C in K under F for time horizon T.

From definitions 2.1 and 2.2, it follows

Theorem 2.3 *The reachable set at time T from a subset C in the closed set K under the Marchaud dynamic F is also the capture-viability kernel of C in reverse time*:

$$\text{Reach}_F(C, K, T) = \text{Capt}_{-F}^T(K, C). \quad (6)$$

The proof is a consequence of articulating Definitions 2.1 and 2.2. The consequence of Theorem 2.3 is that the reachable set can be computed by a viability algorithm.

3. Algorithms

Saint-Pierre [6] devised an algorithm to compute capture-viability kernels when F is Marchaud and Lipschitz. First, he showed that the capture-viability kernel of C under F in a closed set K is another set, the viability kernel of K under $\tilde{F}$, defined as:

$$\forall x \in X, \tilde{F}(x) = F(x)$$
$$\forall x \in C, \tilde{F}(x) = \overline{\text{co}}(F(x) \cup \{0\}), \quad (7)$$

where $\overline{\text{co}}(A)$ is the adherence of the convexified of a set A.

Second, he applied his viability algorithm to $\text{Viab}_F(K)$. The principle of this algorithm is to discretize Equation (1) so that the sequence of subsets K_j starting at $K_0 = K$ and defined recursively by

$$K_{j+1} = K_j \cap F(K_j) \quad (8)$$

converges to a subset contained in the viability kernel of K under F. Saint-Pierre [6] showed that this sequence converges to the viability kernel if F is also Lipschitz:

$$\text{Viab}_F(K) = \bigcap_{j=0}^{\infty} K_j \quad (9)$$

Although this algorithm is theoretically valid in any dimension, in practice, as K is reduced to a discrete grid, the algorithm must be able to update every cell of the grid at any time, which is a formidable task. The algorithm is then limited to three state dimensions.

Bonneuil [4] addressed the computation of viable states and of the viability kernel in large state dimension, using a different procedure, based on stochastic optimization. The idea is to minimize the distance to the set of constraints of solutions starting from a given state, and to assess the viability status of this state whether or not the distance minimization leads to at least one trajectory remaining in the set of constraints.

The set of constraints K is represented by a constraint on state $x \in X$:

$$K = \{x / h(x) \leq 0\}. \quad (10)$$

This constraint on $h(x)$ can be either explicit, such as:

$$h(x) = \max_{i=1,\cdots,m} g_i(x) \quad (11)$$

where $g_i : R^n \to R, i = 1, \cdots, m$; or implicit:

$$h(x) = d(x, K). \quad (12)$$

Define a cost $c(T; x)$ at state x for a given time horizon T as:

$$c(T, x) = \text{Inf}_{x(.) \in S(x)} \text{Sup}_{t \in 0, T} h(x(t)). \quad (13)$$

Bonneuil [4] showed the theorem:

$$x \in \text{Viab}_F^T(K) \iff c(T, x) \leq 0. \quad (14)$$

The implementation of Equation (13) in T dimensions requires a minimization routine in large dimension, such as stochastic optimization. I found exact consistency with the results obtained from Saint-Pierre's algorithm for the

three examples of [1]. For the cases involving a higher state dimension, I checked that the same result is obtained with Saint-Pierre's algorithm when fixing all but three of the variables. To summarize the presentation of the large state dimension procedure:

- If x belongs to the interior of $\text{Viab}_F(K)$, then there is no need to go as far as the minimum of $\text{Sup}_{t \in 0,T} h(x(t))$: the optimization stops as soon as one solution remaining in K is found.
- If $c(T,x) > 0$, the solution starting from x leaves K, and simulated annealing runs its course. The search for viable states is also achieved by the minimization of a distance to the set of constraints, so that the procedure relies on a double stochastic optimization: one where the initial state under examination is fixed, so as to decide whether it is viable or not, and one where this initial state is varied. There is no longer a need to memorize the state of every cell in a grid.

In order to compute reachable sets, I apply one of these two algorithms to the viability-capture basins identified with the reachable sets through Theorem 2.3. In the section below, I treat three noteworthy examples, moreover with time t varying.

4. Computation of Sets of Reachable States

I suggest to compute the three nonconvex reachable sets taken as examples by [1]. The image $F(x)$ is convex in the assumption of Marchaud, but the reachable set can be nonconvex.

The results by Saint-Pierre's [6] algorithm, as I explained, are obtained by up-dating a 3-dimensional grid, so that the display can be given with shaded facets. Bonneuil's algorithm [4] works with points, so that, unless applying a vizualisation software, the display is made of points. Baier and Gerdts [1] study the reachable set at fixed T, and in their conclusion, they call for the computation of the reachable set on an interval $[t_0, T]$. I consider T as an additional dimension, so that I compute the capture-viability kernel of target C within K under the augmented dynamics:

$$\begin{cases} x'(t) \in -F(x(t)) \\ t' = -1 \end{cases} \quad (16)$$

and search for the initial states (x_T, T) for which there exists a solution $x(.)$ remaining in K and such that:

$$\begin{cases} x(t) \in K, \forall t \in [0,T], \\ x(0) \in C, x(T) = x_T. \end{cases} \quad (17)$$

The representation is then done in $X \times R$.

Example 1: *the brachistochrone* corresponds to the control problem:

$$\begin{cases} x_1'(t) = \sqrt{2gx_2(t)} \cos(u(t)) \\ x_2'(t) = \sqrt{2gx_2(t)} \sin(u(t)) \\ x_1(0) = 0 \\ x_2(0) = 1 \\ u(t) \in [-\pi, \pi] \\ t \in [0,1]. \end{cases} \quad (18)$$

I rewrite this problem as a target problem in reverse time, for $x(t) = (x_1(t), x_2(t))$:

$$\begin{cases} x'(t) \in -F(x(t)) \\ t' = -1 \\ C = \{(0,1,0)\} \\ K = X \\ U = [-\pi, \pi] \\ t \in [0,1], \end{cases} \quad (19)$$

with

$$F(x) = \left\{ \left(\sqrt{2gx_2} \cos(u), \sqrt{2gx_2} \sin(u) \right), u \in U \right\}. \quad (20)$$

Figure 1 represents the capture-viability kernel in $X \times R$, with X equal to R. To get the reachable set at a given time T, one has to take a section of this set at T constant; to get the reachable set in $[0;T]$, one has to take the projection of the capture-viability up to T onto the $T = 0$ plane. In viability algorithms, the time horizon T is taken as an additional variable, so that the reachable sets for all T are computed at once, contrary to other procedures.
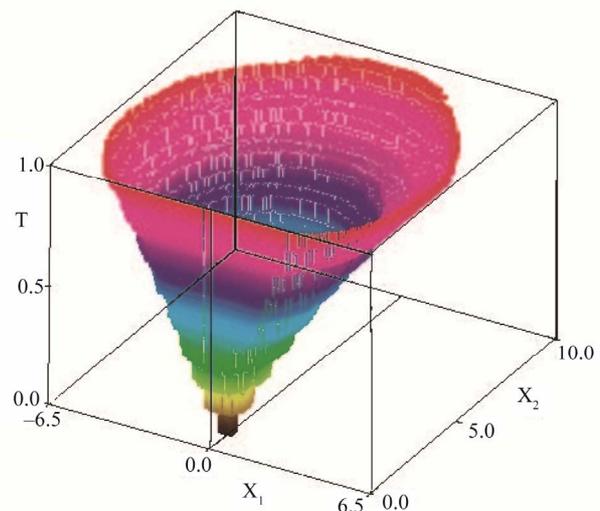
Figure 1. Reachable sets at time T varying: brachistochrone.

Example 2: *Rayleigh problem*
The control problem [1] is:

$$\begin{cases} x_1'(t) = x_2(t) \\ x_2'(t) = -x_1(t) + x_2(t)\left(1.4 - 0.1x_2^2(t)\right) + 4u(t) \\ x_1(0) = -5 \\ x_2(0) = -5 \\ u(t) \in [-1,1] \\ t \in [0,2.5]. \end{cases} \qquad (21)$$

I rewrite this problem as a target problem in reverse time, for $x(t) = \left(x_1(t), x_2(t)\right)$:

$$\begin{cases} x'(t) \in -F(x(t)) \\ t' = -1 \\ C = \{(-5,5,0)\} \\ K = X \\ U = [-\pi,\pi] \\ t \in [0,2.5], \end{cases} \qquad (22)$$

with

$$F(x) = \left\{ \left(x_2, -x_1 + x_2\left(1.4 - 0.1x_2^2\right) + 4u\right), u \in U \right\} \quad (23)$$

The reachable sets at *T* varying are represented on **Figure 2**.

Example 3: *Kenderov*
The control problem [1] is:

$$\begin{cases} x_1'(t) = 8\left(a_{11}x_1(t) + a_{12}x_2(t) - 2a_{12}x_2(t)u(t)\right) \\ x_2'(t) = 8\left(a_{12}x_1(t) + a_{11}x_2(t) - 2a_{12}x_1(t)u(t)\right) \\ x_1(0) = 2 \\ x_2(0) = 2 \\ u(t) \in [-1,1] \\ t \in [0,1]. \end{cases} \qquad (24)$$

Baier and Gerdts [1] consider $a_{11} = \sigma^2 - 1$, $a_{12} = \sigma\sqrt{1 - \sigma^2}$, and $\sigma = 0.9$. I rewrite this problem as a target problem in reverse time, for $x(t) = \left(x_1(t), x_2(t)\right)$:

$$\begin{cases} x'(t) \in -F(x(t)) \\ t' = -1 \\ C = \{(-2,2,0)\} \\ K = X \\ U = [-\pi,\pi] \\ t \in [0,2.5], \end{cases} \qquad (25)$$
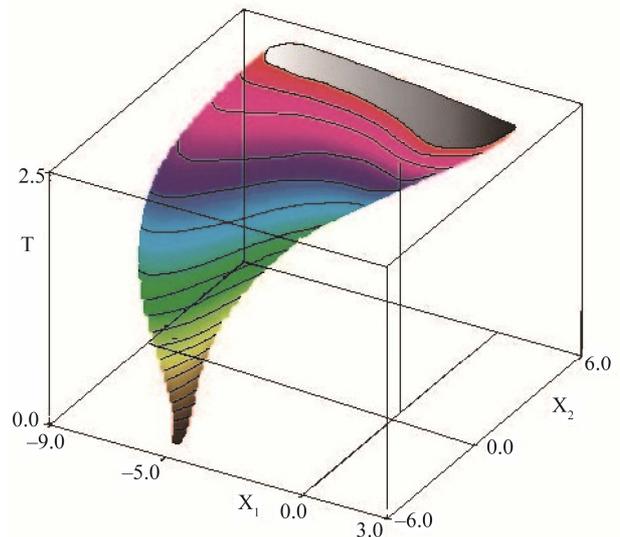
Figure 2. Reachable sets at time *T* varying: Rayleigh.

with

$$F(x) = \left\{ \left(8\left(a_{11}x_1 + a_{12}x_2 - 2a_{12}x_2u\right), \right. \right.$$
$$\left. \left. 8\left(a_{12}x_1 + a_{11}x_2 - 2a_{12}x_1u\right)\right), u \in U \right\} \qquad (26)$$

The reachable sets at *T* varying are represented on **Figure 3**.

Example 4: *in dimension p* + 1
I suggest the control problem:

$$\begin{cases} x_i'(t) = x_p(t), \quad i = 1, \cdots, p-1 \\ x_p'(t) = u(t) \in [-0.5, 0.5] \\ K = [-1,1]^p \times \mathbb{R} \\ C = \left\{0_{\mathbb{R}^{p+1}}\right\}. \end{cases}$$

The projection of the reachable sets at *T* varying in [0; 0.6] onto the plane $[x_1, x_p]$ for *p* = 10 is represented on **Figure 4**, and the same projection at *T* = 0.6 on **Figure 5**. The representation of sets in dimension above three is problematic, but the result is that, in large state dimension, a data set of points can be produced to encompass the reachable set. Then it becomes a problem of delineating a set through knowing a cloud of points.

5. Conclusion

Reachable sets expressed as capture-viability kernels in reverse time allow the use of viability algorithms, either in 2 + 1 dimensions with Saint-Pierre's [6] or in any finite state dimension *p* + 1 with Bonneuil's [4]. This procedure firstly allows avoiding the difficult solving of HJB, and secondly allows the computation of reachable sets in large dimensions. This reasoning in terms of viability
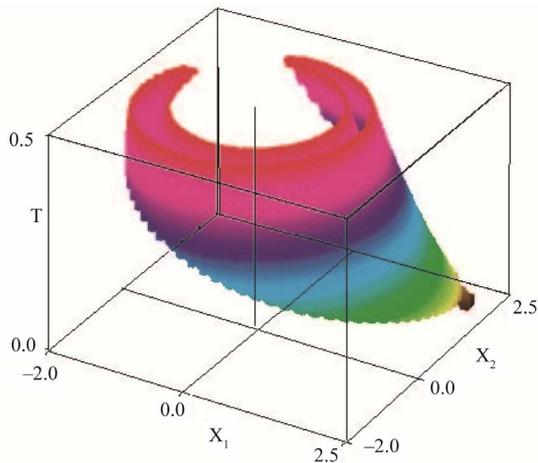
Figure 3. Reachable sets at time _T_ varying: Kenderov.
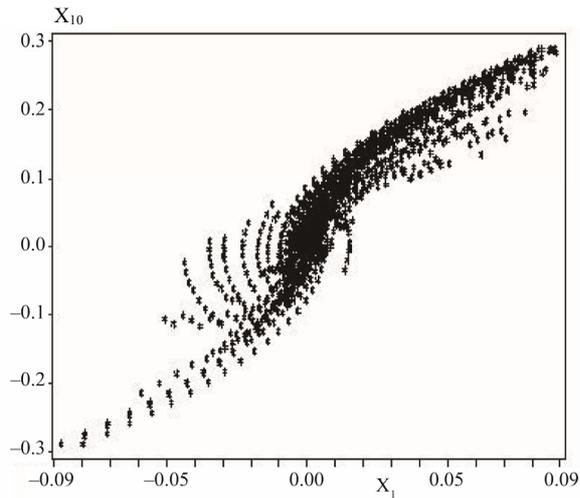
Figure 4. Reachable set at time $T \in [0; 0.6]$ for the 10 + 1 dimension example.
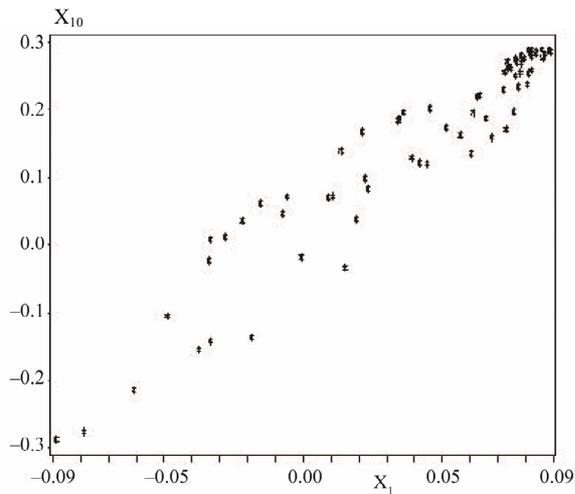
Figure 5. Reachable set at time $T = 0.6$ for the 10 + 1 dimension example.

is flexible and proved useful in the computation of maxima under viability constraints [7].

REFERENCES

[1] R. Baier and M. Gerdts, "A Computational Method for Non-Convex Reachable Sets Using Optimal Controls," *Proceedings of the European Control Conference* 2009, Budapest, 23-26 August 2009, pp. 97-101.

[2] R. Baier, "Set-Valued Integration and Discrete Approximation. Reachable Sets," Bayreuth Mathematical Reports 50, University of Bayreuth, Bayreuth, 1995.

[3] R. Baier, M. Gerdts and I. Xausa, "Approximation of Reachable Sets Using Optimal Control Algorithms," 2011. http://num.math.uni-bayreuth.de/en/publications/2012/baier _gerdts_xausa_approx_reach_sets_2011/index.html

[4] N. Bonneuil, "Computing the Viability Kernel in Large State Dimension," *Journal of Mathematical Analysis and Applications*, Vol. 323, No. 2, 2006, pp. 1444-1454. doi:10.1016/j.jmaa.2005.11.076

[5] J.-P. Aubin, "A concise introduction to Viability Theory, Optimal Control and Robotics," École Normale Supérieure de Cachan, Cachan, 2003.

[6] P. Saint-Pierre, "Approximation of the Viability Kernel," *Applied Mathematics and Optimization*, Vol. 29, No. 2, 1994, pp. 187-209. doi:10.1007/BF01204182

[7] N. Bonneuil, "Maximum under Continuous-Discrete-Time Dynamic with Target and Viability Constraints," *Optimization*, Vol. 61, No. 8, 2012, pp. 901-913. doi:10.1080/02331934.2011.605127