Notes on concepts and interpretation of control systems (Draft)*

João Borges de Sousa
Departamento de Engenharia Electrotécnica e de Computadores
Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias, 4200-465, Porto, Portugal
jtasso@fe.up.pt

April 13, 2008

Contents

*This document is a work in progress. Comments and suggestions are welcome. Please send them to jtasso@fe.up.pt

1 Introduction

> Signals convey information. Systems transform signals.... A **signal** is a function that maps a domain, often time or space, into a range, often a physical measure such as air pressure or light intensity. A **system** is a function that maps signals from its domain – its input signals - into signals in its range – its output signals. The domain and the range are both sets of signals (signal spaces). Thus, systems are functions that operate on functions..... For systems, we look at the relationship between the input and output signals (this relationship is a **declarative description** of the system) and the procedure for converting an input signal into an output signal (this procedure is an **imperative description** of the system). In [17].

We are interested in designing systems (controllers) to control other systems (controlled systems). To design a controller we need descriptions of:

- Controlled system.
- Specification of the desired behavior for the controlled system.

We need formal models of the controlled system and the specification to reason about the design of the controller and communicate the design in clear way with precise semantics. Such models are much more precise than the English-language descriptions that are commonly used for such systems.

These notes are about reviewing fundamental concepts in control systems and developing the intuition behind these concepts. The focus is on the geometric intuition, rather than on the mathematical details. The notes are intended to develop this geometric intuition around the properties of the system's reach set – informally this is the set of states that can be reached by the trajectories of the system. In this setting, system's properties such as invariance, stability, attainability and safety are described in an uniform way, and the intuition for controller design is developed.

First we discuss systems as functions, and present examples of systems described by differential equations, difference equations, state-machines and hybrid automata. We do this to present the systems' background in all of its generality. Finally, we particularize the discussion to systems described by differential equations. This is done to make the discussion more tangible. With the exception of details specific to differential equations, the presentation is aimed at building a conceptual framework applicable to other system's descriptions.

The notation and approach follow closely [17].

2 Descriptions of systems

This section follows closely [17].

A description of a system as a function involves three entities: the set of input signals, the set of output signals, and the function itself,

$$F : \mathit{InputSignals} \rightarrow \mathit{OutputSignals} \quad (1)$$

Several types of systems of interest to control design are described in this section.

> A broad class of systems can be characterized using the concept of **state** and the idea that a system evolves through a sequence of changes in state, or state transitions. Such characterizations are called state-space models. A state-space model describes a system procedurally, giving a sequence of step-by-step operations for the evolution of a system. It shows how the input signal drives changes in state, and how the output signal is produced. It is thus an **imperative** description. Implementing a system described by a state-space model in software or hardware is straightforward. The hardware or software simply needs to sequentially carry out the steps given by the model. In [17].

2.1 Continuous-time systems

Consider a class of systems given by functions

$$S : \mathit{ContSignals} \rightarrow \mathit{ContSignals} \quad (2)$$

where *ContSignals* is a set of continuous-time signals.

$\mathit{ContSignals} = [\mathit{Time} \rightarrow \mathit{Reals}^n]$ or $\mathit{ContSignals} = [\mathit{Time} \rightarrow \mathit{Complex}^n]$, where $\mathit{Time} = \mathit{Reals}$ or $\mathit{Time} = \mathit{Reals}+$.

These are often called continuous-time systems because they operate on continuous-time signals. Frequently, such systems can be defined by differential equations that relate the input signal to the output signal.

A prototypical description of a controlled (there is a control input signal) continuous-time system is:

$$\dot{x}(t) = f(t, x(t), u(t)), u(t) \in U(t) \quad (3)$$

where $f : \mathit{Time} \times \mathit{Reals}^n \times \mathit{Reals}^p \rightarrow \mathit{Reals}^n$ satisfies the conditions for existence and uniqueness of the ordinary differential equation and u is our control.

2.2 Discrete-time systems

Consider another class of systems given by functions

$$S : \mathit{DiscSignals} \rightarrow \mathit{DiscSignals} \quad (4)$$

where *DiscSignals* is a set of discrete-time signals.

$\mathit{DiscSignals} = [\mathit{Integers} \rightarrow \mathit{Reals}^n]$ or $\mathit{DiscSignals} = [\mathit{Integers} \rightarrow \mathit{Complex}^n]$ or $\mathit{DiscSignals} = [\mathit{Naturals}_0 \rightarrow \mathit{Reals}^n]$.

These are often called discrete-time systems because they operate on discrete-time signals. Frequently, such systems can be defined by difference equations that relate the input signal to the output signal.

A prototypical description of a controlled discrete-time system is:

$$\forall n \in \ \mathit{Integers}\ ; x(n+1) = x(n) + F(n, x(n), u(n)), u(n) \in U(n) \tag{5}$$

where $F : \mathit{Integers} \times \mathit{Reals}^n \times \mathit{Reals}^p \rightarrow \mathit{Reals}^n$.

2.3 State machines

For a state machine, the input and output signals have the form *EventStream* : $\mathit{Naturals}_0 \rightarrow \mathit{Symbols}$; where $\mathit{Naturals}_0 = 0; 1; 2; \ldots$ and *Symbols* is an arbitrary set. The domain of these signals represents ordering but not necessarily time (neither discrete nor continuous time). The ordering of the domain means that we can say that one event occurs before or after another event.

A state machine constructs the output signal one element at a time by observing the input signal one element at a time. Specifically, a state machine *StateMachine* is a 5-tuple,

$$\mathit{StateMachine} = (\mathit{States; Inputs; Outputs; update; initialState}) \tag{6}$$

where *States; Inputs; Outputs* are sets, *update* is a function, and $\mathit{initialState} \in \mathit{States}$. The meaning of these names is:
States is the **state space**,
Inputs is the **input alphabet**,
Outputs is the **output alphabet**,
$\mathit{initialState} \in \mathit{States}$ is the **initial state**,
and *update:* $\mathit{States} \times \mathit{Inputs} \rightarrow \mathit{States} \times \mathit{Outputs}$ is the **update function**.

This five-tuple is called the **sets and functions model** of a state machine. *Inputs* and *Outputs* are the sets of possible input and output values or symbols. The set of input signals consists of all infinite sequences of input values,

$$\mathit{InputSignals} = [Naturals_0 \rightarrow \mathit{Inputs}] \tag{7}$$

The set of output signals consists of all infinite sequences of output values,

$$\mathit{OutputSignals} = [\mathit{Naturals}_0 \rightarrow \mathit{Outputs}] \tag{8}$$

Let $x \in \mathit{InputSignals}$ be an input signal. A particular element in the signal can be written x(n) for any $n \in \mathit{Naturals}_0$. We write the entire input signal as a sequence

$$x(0), x(1), x(2), \ldots \tag{9}$$

This sequence defines the function x in terms of elements $x(n) \in Inputs$, which represent particular input values. We reiterate that the index n in $x(n)$ does not refer to time, but rather to the step number.

The interpretation of *update* is this. If $s(n) \in States$ is the current state at step n, and $x(n) \in Inputs$ is the current input, then the current output and the next state are given by

$$(s(n+1); y(n)) = update(s(n); x(n)) \tag{10}$$

Thus the update function makes it possible for the state machine to construct the output signal step by step by observing the input signal step by step.

The state machine StateMachine of (6) defines a function

$$F : InputSignals \to OutputSignals \tag{11}$$

such that for any input signal $x \in InputSignals$ the corresponding output signal is $y = F(x)$. However, it does much more than just define this function. It also gives us a procedure for evaluating this function on a particular input signal. The state response $(s(0); s(1); \dots)$ and output signal y are constructed as follows:

$$s(0) = initialState; \tag{12}$$

$$\forall n \geq 0; (s(n+1); y(n)) = update(s(n); x(n)); \tag{13}$$

Observe that if the initial state is changed, the function F will change, so the initial state is an essential part of the definition of a state machine.

Each evaluation of (13) is called a **reaction** because it defines how the state machine reacts to a particular input symbol. Note that exactly one output symbol is produced for each input symbol. Thus, it is not necessary to have access to the entire input sequence to start producing output symbols. This feature proves extremely useful in practice, since it is usually impractical to have access to the entire input sequence (it is infinite in size!). The procedure summarized by (12, 13) is **causal**, in that the next state s(n+1) and current and current output y(n) depend only on the initial state s(0) and current and past inputs $x(0); x(1); \dots; x(n)$.

2.4 Hybrid automata

The formal definition of hybrid automata is not presented here. Informally, it is a hybrid of the descriptions of a differential equation and a state-machine.

3 Specifications for state-space models

The formal specification of the desired behavior for a controlled system is strongly dependant on the system's description. Specifications for state-space models typically concern problems of invariance, attainability, safety and performance. The problem of invariance consists of keeping the state of the system inside a subset of the state-space. The problem of attainability consists of driving the state of the system to enter a subset of the state-space. The problem of safety consists of preventing the state of the system to enter a subset of the state-space. The problem of performance involves of evaluating trajectories of the system according to some cost function for problems of invariance, attainability and safety. Depending on the time (or step) horizon, there may be finite or infinite-time specifications. These may require different mathematical machinery. For example, properties like stability for continuous-time systems are specified using limits when time tends to infinity; properties concerning fairness for state-machines are specified using some logic.

In what follows we relate problems of invariance, attainability, safety and performance to the system's reach set. This is illustrated for continuous-time systems.

4 Continuous-time systems

4.1 Model

Consider the following model of a system whose state x evolves in $\mathbb{R}^n$:

$$\dot{x}(t) = f(t, x, u), u \in U(t) \subset \mathbb{R}^p \tag{14}$$

where f satisfies the conditions for existence and uniqueness of the ordinary differential equation and u is our control.

When this system's description is well "behaved" (f is Lipschitz in x and continuous in u) there is an equivalent description:

$$\dot{x}(t) \in F(t, x(t)) \subset \mathbb{R}^p \tag{15}$$

where $F(t, x) := \{s : s = f(t, x, u), u \in U(t)\}$.

This is a differential inclusion. The set-valued map F maps (t,x) onto the set of admissible velocities at (t,x). The local properties of the system depend on the geometry of this set.

Exercise 1 (Geometry of differential inclusions) *Consider the differential inclusion $\dot{x} \in F(x) \subset \mathbb{R}^2$ ($x = (x_1, x_2)$) describing a well "behaved" system. Is it feasible for the system to move in all directions in $\mathbb{R}^2$ when the origin is not an interior point of F?*

Remark 1 (Local controlability) *The condition $0 \in Int(F)$ is necessary for local controlability.*

Given an initial condition $x(t_0) = x_0$ it is natural to ask what is the set of points that can be reached at time $t > t_0$ starting from x_0.

Exercise 2 (Reach set calculation) *Consider the system:*

$$\dot{x}(t) = u, u \in B_1(0) \subset \mathbb{R}^2, x_0 = (0,0) \tag{16}$$

where $B_1(0)$ denotes the unit ball centered at the origin.

1. *Draw the set of all positions that can be reached in one time unit starting at x_0.*

2. *Consider the sets of all positions that can be reached in 1, 2, 3 and 4 time units. Draw the evolution of these sets with respect to time (in $\mathbb{R}^3$).*

Exercise 3 (Reach set calculation under state constraints) *Consider the system given by equation (16) and state constraint:*

$$x_1 \leq 0$$

1. *Draw the set of all positions that can be reached in one time unit starting at x_0 under this state constraint.*

2. *What is the relation between the trajectories of the system and the line $x_1 = 0$?*

4.2 Reach sets

Informally, the reach set of a system described by a differential equation is the set of all states that can be reached from an initial state within a given time interval.

The knowledge of the reach set is quite important for control applications. Consider, as examples, the following applications in vehicle control:

1. When will a car collide with an obstacle?
2. Where and when is it feasible for two vehicles to rendezvous?
3. What is the set of initial positions such that if a vehicle departs from this set it will be able reach a target in a given time interval?
4. What is the set of initial positions such that the trajectories of the vehicle will never leave a closed set S?

Exercise 4 (Reach sets and applications) *Use the informal concept of reach set to give a geometric interpretation to these vehicle control applications.*

4.2.1 Forward reachability

Consider the system described by equation (3).

Definition 1 (Reach set starting at a given point) *Suppose the initial position and time (x_0, t_0) are given. The reach set $R[\tau, t_0, x_0]$ of system (3) at time $\tau \geq x_0$, starting at position and time (x_0, t_0) is given by:*

$$R[\tau, t_0, x_0] = \bigcup \{x(\tau), u(s) \in U(s), s \in (t_0, \tau]\} \tag{17}$$

Definition 2 (Reach set starting at a given set) *The reach set at time $\tau > t_0$ starting from set X_0 is defined as:*

$$R[\tau, t_0, X_0] = \bigcup \{R[\tau, t_0, x_0] | x_0 \in X_0\} \tag{18}$$

Consider now the case of adversarial behavior:

$$\dot{x}(t) = f(x(t), u(t), v(t), t), u(t) \in U(t) \subset \mathbb{R}^u, v(t) \in V(t) \subset \mathbb{R}^v \tag{19}$$

- u is our control.
- v is controlled by an adversary. We don't know what the adversary will do (you may assume the worse case scenario).

Exercise 5 (Reach set under adversarial behavior) *Write the definition of reach set for this system.*

Consider the following type of state constraints:

$$\phi(x, t) \leq 1 \tag{20}$$

Exercise 6 (Reach set under state constraints) *Write the definition for this case.*

The reach set of a dynamic system is a complex 'creature'.
See examples in the power point presentation

4.2.2 Backward reachability

Until now we have discussed the problem of 'forward' reachability – we integrate the differential equation 'forward' in time. Now we briefly discuss 'backward' reach sets.

Exercise 7 *Given a set X_f and the dynamic system (3) write the definition for the 'backward' reach set. [Hint: integrate the differential equation backwards in time.]*

Analogously, we can write the definition for backward reach sets under state constraints or adversarial behavior.

Exercise 8 *Explain why the notion of 'backward' reach set is useful in some control applications? [Hint: try to relate this notion to the problem where a vehicle is trying to reach a target destination.]*

4.2.3 Why are we interested in reach sets?

Reach sets of dynamic systems are pervasive in control. This is because the reach set describes the motion capabilities of a dynamic system.

Consider a problem of control synthesis abstractly formulated as follows: given a dynamic system and a specification synthesize a controller so that the composition of system with the controller satisfies the specifications.

Consider a problem of verification abstractly formulated as follows: given a dynamic system and a controller check if the composition of the two satisfies a given property.

Consider a dynamic system and a closed set S. Examples of the properties we are interested in are:

Invariance – the trajectories of the system do not leave S.

Attainability – the trajectories of the system will enter S.

Safety – the trajectories of the system do not enter S.

Checking for those properties amounts to solving reachability problems.

There are synthesis techniques which produce controllers with guaranteed properties: this means it is not necessary to go through the verification phase to check if the system and the controller satisfy a given property. This is the case with some techniques from dynamic optimization which will be described in this document.

Arguments involving reachability concepts are also used to prove results in control theory.

4.3 Invariance

We are interested in understanding the geometry of reach sets and the relationships between reach sets, control, and invariance.

Exercise 9 *Consider:*

- *The linear system (B is the identity* 2×2 *matrix), whose state* $x \in \mathbb{R}^2$*:*

$$\dot{x}(t) = Bu(t), u(t) \in U = \{(u_1, u_2) \in \mathbb{R}^2 : u_1 \geq 0, u_2 = 0\} \tag{21}$$

- *The closed unit ball in* $\mathbb{R}^2$*, centered at the origin,* $\mathcal{B}_1(0)$*.*
- *The initial state* x_0 *at time* t_0*:* $x_0 = (0, 1)$*.*

Consider the corresponding differential inclusion $\dot{x} \in F(t, x)$*. Is there any control* $u \in U$ *that is able to drive the state of the system to the interior of* $\mathcal{B}_1(0)$*? [Hint: draw the set of all velocities at that position.]*

Exercise 10 *Consider the same linear system with a different control constraint*

$$U = \{(u_1, u_2) \in \mathbb{R}^2 : u_1 \geq 0, u_2 \in \mathbb{R}\}$$

Is there any control $u \in U$ that is able to drive the state of the system into the interior of $\mathcal{B}_1(0)$? [Hint: draw the set F of all velocities at that position.]

Exercise 11 *Given a point at the boundary of a closed set S what is the condition that F needs to satisfy for the trajectory of the system to penetrate S?*

Exercise 12 *Consider the differential inclusion F corresponding to the dynamic system described by equation (3) and a closed set S.*

1. *What is a necessary condition for the existence of trajectories of the system that will never leave the set? [Hint: use the geometrical interpretation for the solution of the previous exercise.]*
2. *Assume that the previous condition is satisfied. Derive a controller which ensures that the trajectories of the system will never leave the set S.*

Consider a closed set S and a system described by the set-valued map F.

Definition 3 (Weak invariance) *The pair (S,F) is weakly invariant if there exist controls such that a trajectory starting inside S remains inside S.*

Exercise 13 *Same as previous exercise but with the universal quantifier.*

Definition 4 (Strong invariance) *The pair (S,F) is strongly invariant if the trajectories starting inside S never leave S.*

Ok, now you are an expert in invariance of controlled dynamic systems. Invariance is also called viability (see [11], [12], [2], [1],[5]).

Exercise 14 *Is there any relation between the notion of reach set and the notion of invariance with respect to a set? [Hint: see what happens in terms of the velocity vectors at the boundary of the reach set]*

Let us pursue the geometric interpretation behind the design of controller based on the invariance of the pair $(F; S)$.

Exercise 15 *Consider the pair (S,F) from exercise 10, where S is the unit ball centered at the origin. Consider the point* $x = (0, 1.5)$*. Consider the distance function between this point and the set S. Pick one control* $u \in U$ *such that this distance will decrease. What is the control* $u \in U$ *which maximizes the rate at which this distance decreases?*

Remark 2 (Extremal aiming) *This is the geometric interpretation behind the derivation of the extremal aiming controller introduced by Krasovskii (see [10] for details).*

Remark 3 (Non-smooth analysis) *The relationships between reach sets, control, and invariance provide the intuition for developments in non-smooth analysis and control. The notions of convexity, distance, and projection are essential to extend the classical notions of derivative and gradient to families of functions that lack smoothness but present some form of regularity (see [4, 20] for more details).*

4.4 Attainability

In its simplest version, the problem of attainability for continuous-time systems involves a closed target set S and an initial state for the system. Some versions of this problem may involve a target set evolving with time. There is also the question of attainability in finite and infinite time.

In all of these formulations the question of interest consists of finding trajectories of the system which intersect the target set after departing from the initial state.

Consider the system described by equation (3). Let t_f denote the first time when the trajectory of the system hits the target set S.

$$t_f = \inf\{t : x(t) \in S\} \tag{22}$$

Exercise 16 *Why do we need the set S to be closed?*

Exercise 17 *Given a closed set S and the system described by equation (3) derive the conditions for attainability in terms of the reach set of the system for the following cases:*

1. *There are no time constraints.*
2. *The target must be reached during time interval $[t_1, t_2]$*

4.5 Computation of reach sets

The computation of reach sets is not a trivial matter.

Q. Why?

A. Because the reach set inherits the behavior of a dynamic system. See the behavior of some nonlinear systems.

Several techniques for reachability analysis of hybrid systems have been proposed. They can be (roughly) classified into two kinds:

1. Purely symbolic methods based on (a) the existence of analytic solutions to the differential equations and (b) the representation of the state space in a decidable theory of the real numbers.

2. Methods that combine (a) numeric integration of the differential equations and (b) symbolic representations of approximations of state space typically using (unions of) polyhedra or ellipsoids.

These techniques provide the algorithmic foundations for the tools that are available for computer-aided verification of hybrid systems ([24] [6] , [9], [22]).

The set-valued Lebesgue integral provides a conceptual tool for the direct computation of the reach set.

In what follows we describe techniques from dynamic optimization which are used to compute reach sets for dynamic systems.

4.6 Dynamic optimization for reach set computation

4.6.1 Introduction

The relation between dynamic optimization and reachability was first observed in [18].

A typical problem of optimal control can be formulated as follows:

$$\max \int_{t_0}^{t_f} c(t, x(t), u(t))dt + G(x(t_f)) \tag{23}$$

$$\dot{x}(t) = f(x, u, t), u \in U(t) \subset \mathbb{R}^p \tag{24}$$

There are two main techniques to solve this problem: 1) the maximum principle; 2) dynamic programming. The maximum principle gives necessary conditions of optimality. Dynamic programming may be used to derive sufficient conditions of optimality.

A good reference on the maximum principle is [19]. A less known reference with detailed geometric interpretations is [8]. A good reference on dynamic programming is [3].

4.6.2 The maximum principle and reach sets for linear systems

Here we will develop the geometric intuition behind the maximum principle for linear systems.

Exercise 18 *Consider the open unit ball in $\mathbb{R}^2$. Given $c = (c_1, c_2) \in \mathbb{R}^2$ solve the following optimization problem:*

$$\max_{x \in \mathcal{B}_1(0)} (x \cdot c) \tag{25}$$

Exercise 19 *Same as previous but with the closed unit ball.*

Q. What can we conclude? [Hint: closed sets are important for the existence of solution].

Given an optimization problem, such as the one in exercise 19, we define *argmax* as the set of all solutions to the problem.

Exercise 20 *Consider the closed unit ball in* $\mathbb{R}^2$*. Given* $c = (c_1, c_2) \in \mathbb{R}^2$*, for example* $c = (1, 1)$*, find :*

$$\arg\max_x \in \overline{\mathcal{B}}_1(0)(x \cdot c) \tag{26}$$

Exercise 21 *Write the definition of a convex set.*

Exercise 22 *Write the definition of a convex function.*

Definition 5 (Epigraph of a function) *Given a function $f(x) : \mathbb{R}^n \mapsto \mathbb{R}$ the epigraph of f, denoted by $epi(f)$, is the set of all points in $\mathbb{R}^{n+1}$ that lie above the graph of the function:*

$$epi(f) = \{(x, y) \in \mathbb{R}^{n+1} : x \in dom(f), y \geq f(x)\}$$

dom(f) denotes the domain of f.

Exercise 23 *Relate the two definitions of convexity. [Hint: Consider the epigraph of a function.]*

Q. Conclusions?

Definition 6 (Support function to a closed convex set) *Given a closed set S in $\mathbb{R}^n$, the support function in the direction $l \in \mathbb{R}^n$ to the set S, denoted $\rho(l|S)$, is defined as:*

$$\rho(.|.) : \{l|S\} \mapsto \rho(l|S) = max\{(x \cdot l)|x \in S\} \tag{27}$$

The support function satisfies the following property:

$$\rho(l|AS + b) = \rho(A'l|S) + (l \cdot b) \tag{28}$$

Exercise 24 *Consider the support function of the closed unit ball (as in exercise 19). If* $\rho((1,0)|\overline{\mathcal{B}}_1(0)) = 1$ *what is the point in the closed unit ball where the max occurs.*

Exercise 25 *Same as previous but consider the closed unit square instead of the closed unit ball. What happens? [Hint: the argmax in the second case is a compact set, while in the first case it is a singleton].*

<u>Q.</u> Conclusions?

<u>A.</u> Strict convexity is crucial.

Exercise 26 *Given a closed set what is the geometric meaning of the support function calculated at a given vector? [Hint: find the argmax of the problem, and draw the tangent planes to the set at all points in argmax].*

The 'fancy' name for that plane is 'Separating hyperplane'. We can also define separating ellipsoids.

Q. Separate what from what?

Q. Imagine that we know that the reach set of linear system under bounded controls is closed and strictly convex. Imagine that somebody gives us the support function to the reach set at time t. Are we able to characterize the reach set at that time?

Consider the linear system:

$$\dot{x} = Ax + Bu, u \in U \tag{29}$$

Now imagine that we are able to solve the following optimal control problem.

$$\max x(1) \cdot c \tag{30}$$

where

$$c \in \overline{\mathcal{B}_1(0)} \tag{31}$$

$$x(0) = x_0 \tag{32}$$

An equivalent formulation is:

$$\max_{x \in R[1,0,x_0]} x \cdot c \tag{33}$$

Q. Can we relate the solution of this problem to the support function to the reach set at the final time?

Q. Imagine we vary c. Are we able to determine approximations to the reach set?

For more details on the relation between the maximum principle and reach set computation see [23].

Assignment. Read [23].

For more details on ellipsoidal approximations and reach set computation see [12].

A fundamental reference in convex and variational analysis is [20].

4.6.3 Dynamic programming

Dynamic programming (DP) is one of the techniques used to solve optimal control problems. This is done by embedding the original optimal control problem into a family of optimization problems which depend on the state of the system. DP introduces the value function to characterize this dependency.

In this section we introduce the concept of value function which is associated to optimal control problems and show how to derive the Hamilton-Jacobi-Bellman equation (which is satisfied by the value function) from the Principle of Optimality. This is done for a simple example of minimum-time optimal control. For a thorough treatment of this problem see [3], pag. 239.

Consider the following model of a system whose state x evolves in $\Re^n$:

$$\dot{x}(t) = f(x, u), u \in U \subset \mathbb{R}^p \tag{34}$$

where f satisfies the conditions for existence and uniqueness of the ordinary differential equation and u is our control.

Assumption 1 *In what follows we assume that the system (15) is locally controllable.*

Consider a bounded and closed set S with non-empty interior.

Let t_f denote the first time when the trajectory of the system hits the target set S.

$$t_f = \inf\{t : x(t) \in S\} \tag{35}$$

Exercise 27 *Why is the assumption on local controllability useful?*

Consider the problem.

Problem 1 *Let $x(0) = x_0$. Find:*

$$\inf_{u(.)} t_f \tag{36}$$

where $u(.) : \mathbb{R} \to \mathbb{R}^p$ is an admissible control function.

Under the assumptions for system and for the target set the infimum is attained at a time $T \in \mathbb{R}$.

Introduce the value function $T : \mathbb{R}^n \to \mathbb{R}$ as

$$T(x) = \inf_{u(.)} t_f \tag{37}$$

Take an optimal trajectory departing from $x(0) = x_0$. Consider a pair (x^*, t^*) on this trajectory.

Exercise 28 *Consider a new optimization problem when the trajectory of the system departs from $x(0) = x^*$. Find:*

$$\inf_{u(.)} t_f \tag{38}$$

What can you say about the solution of this problem?

Consider the pair (x^*, t) on a trajectory departing from $(x_0, 0)$. The principle of optimality (PO) for problem (1) can be expressed as follows:

$$T(x_0) \leq t + T(x^*) \tag{39}$$

Equality holds for optimal trajectories. The interpretation is quite simple. If a point is on the optimal trajectory, it is optimal to stay on the optimal trajectory.

Assumption 2 *The value function T is differentiable.*

The Hamilton-Jacobi-Bellman (HJB) equation for this problem can be interpreted as an infinitesimal version of the principle of optimality. For this purpose divide both terms of equation (39) by t and take limits when $t \to 0$. Keep in mind that we are taking the total derivative of T with respect to t. This means that first we take the derivative with respect to x and multiply it by the derivative of x with respect to time.

$$\inf_{u \in U} -\nabla T(x_0) \cdot f(x, u) = 1 \tag{40}$$

This is a partial differential equation. The boundary condition is $T(x) = 0, x \in S$.

Exercise 29 *Derive the HJB equation (40) for the time optimal control problem (Problem 1). Hint: use an infinitesimal version of the PO.*

4.6.4 Value functions for reach set computations

In this section we use the approach from [16] to formulate the problem of reach set computation as an optimization problem (see [11], [12], [2], [1]). The key observation is that the reach set is the level set of an appropriate value function [16]. To illustrate this point consider the following value function:

$$V(\tau, x) = \min_{u(.)} \{d^2(x(t_f), \mathcal{X}_f) | x(\tau) = x\}$$
$$V(t_f, x) = d^2(x, \mathcal{X}_f) \tag{41}$$

where u(.) is an admissible control function defined for $[t_0, \tau]$ and $d(x(t_f), \mathcal{X}_f)$ is the Euclidean distance between the state of the system at time t_0 and the initial set $\mathcal{X}_0$ for a trajectory starting at (τ, x). Obviously, (τ, x) belongs to the forward reach set if this distance is zero. But this also means that the forward reach set is the zero level set of the value function V:

$$W[t_0, \tau, \mathcal{X}_0] = \{x | V(\tau, x) \leq 0\} \tag{42}$$

Exercise 30 *Consider the dynamic system (3) and X_0. Given a point (x,t), where $t \geq t_0$, can you devise a test to check if this point belongs to $R[t, t_0, X_0]$*

The question now is how to compute the value function. This is not a trivial matter. The idea is to transform this global problem into a local one. We do this by transforming the global problem onto a partial differential equation.

In general, the value function V can be computed through the generalized Hamilton-Jacobi-Bellman (HJB) equation. We can only do this if the value function satisfies the principle of optimality.

Theorem 1 *The value function V satisfies the principle of optimality:*

$$V(\tau, x|V(t_0, .)) = V(\tau, x|V(t, .|V(t_0, .))), t_0, \leq t \leq \tau \tag{43}$$

Basically the principle of optimality states that the value function satisfies a semi-group property. The value function inherits this property from the semi-group property of the reach set.

An infinitesimal version of the principle of optimality leads to Hamilton-Jacobi-Bellman equation:

$$V_t(t, x) + \max_{u \in U(t)} (V_x(t, x) \cdot f(t, x, u)) = 0 \tag{44}$$
$$V(t_0, x) = 0, x \in \mathcal{X}_0$$

where V_t, V_x represent the corresponding sub-differentials. This results from the fact that the value function is generally non-differentiable, and we have to use generalized notions of derivatives.

Since V is non-differentiable the usual notion of solution of a partial differential equation does not apply. We consider generalized "viscosity", or equivalent concepts, of solutions for this equation (see [7, 10, 21, 3]).

Theorem 2 *The value function V is the unique viscosity solution of (44).*

Next we apply the same technique to the calculation of the forward reach set with state constraints. The state constraints are expressed as before in equation (20). Consider the following value function.

$$V(\tau, x) = \min_{u(.)} \max\{ \max_{s \in [t_0, \tau]} \{\phi(\tau, x(s)), \phi_0(x(t_0))\}, x(\tau) = x\} \tag{45}$$

where u(.) is a feasible control function ($u(s) \in U(s), s \in [t_0, \tau]$).

The sub-level set of this value function given by the following equation:

$$R(\tau, x) = \{x : V(\tau, x) \leq 1\} \tag{46}$$

Using the techniques from [16] we can derive the HJB equation for this problem. First we introduce some notation:

$$\mathcal{H}(t, x, y, V, u) = V_t(t, x, y) + (V_x(t, x, y) \cdot f(t, x, y, u)) \tag{47}$$

The HJB for this problem is given by:

$$\begin{aligned} V_t(t, x) + \max_{u \in U(t)} (V_x(t, x) \cdot f(t, x, u)) = 0 \\ \text{when } V(t, x) \neq \phi(t, x) \\ \max_{u \in \mathcal{U}} \{\min(\mathcal{H}(t, x, y, V, u), \mathcal{H}(t, x, y, \phi, u)), u \in \mathcal{U}\} \\ \text{when } V(t, x) \neq \phi(t, x) \\ V(t_0, x) = \max(\phi(t_0, x), \phi_0(t_0, x)) \end{aligned} \tag{48}$$

Exercise 31 *Why is this approach to reach set computation so interesting?*

In this approach we can phrase all of the problems of reach set computation in terms of the solution of a Hamilton-Jacobi-Bellman (HJB) or Hamilton-Jacobi-Bellman-Isaacs partial differential equation. See [13, 14, 15, 16] for a detailed description of the application of HJB or of HJBI to reach set computation.

4.6.5 A direct method

Consider the following dynamic system with state $x \in \mathbb{R}$ controlled by two adversary control inputs u and v:

$$\dot{x} = f(t, x, u, v), u \in P, v \in Q \tag{49}$$

where the following hypotheses hold:

H1) f is continuous in all variables and $t \in T = (-\infty, \theta]$.

H2) For any bounded region D in $\mathbb{R} \times \mathbb{R}$, f satisfies the following Lipschitz condition:

$$\|f(t, x_1, u, v) - f(t, x_2, u, v)\| \leq \lambda(D)\|x_1 - x_2\|$$

for any $(t, x_i) \in D, (u, v) \in P \times Q$

H3) For any $(t, x, u, v) \in T \times \mathbb{R} \times P \times Q$ the following inequality, where σ is a constant, is valid:

$$xf(t, x, u, v) \leq \sigma(1 + \|x\|^2)$$

H4) For any $(t, x) \in T \times \mathbb{R}$ and $s \in \mathbb{R}$ the so-called 'saddle point condition in a small game' is valid[1]:

$$\min_{u \in P} \max_{v \in Q} sf(t, x, u, v) = \max_{v \in Q} \min_{u \in P} sf(t, x, u, v)$$

Consider the target set M:

$$M = \{(t, x) \in T \times \mathbb{R} : t = \theta, l_1 \leq x \leq l_2\}$$

Now consider a game with two players controlling u and v respectively. The objective of u is to steer x to the target set M. The objective of v is exactly the opposite. More formally, consider the following cost functional:

$$\gamma(x(t_0, x_0, U(.), V(.))) = \begin{cases} 0 & \text{if } x(.) \text{ intersects M} \\ 1 & \text{otherwise} \end{cases}$$

where U(.) is a control function for the first player and V(.) is a control function for the second player. Note that $\gamma(x(t_0, x_0, U(.), V(.))) = 0$ means that the trajectory $x(.)$ departing from (t_0, x_0) under controls $U(.)$ and $V(.)$ enters the target set M at time θ.

The adversarial aspect of control is captured in an optimization problem where u seeks to minimize γ and v seeks to maximize it. Under the hypotheses (H1-4) this game has a value $V_g(t_0, x_0)$. This means that:

[1]This condition holds, for example, for linear systems.

$$V_g(t_0, x_0) = \inf_{U(.)} \sup_{V(.)} \gamma(x(t_0, x_0, U(.), V(.))) = \sup_{V(.)} \inf_{U(.)} \gamma(x(t_0, x_0, U(.), V(.))) \quad (50)$$

Moreover, the game has a saddle point, i.e., there exist strategies $U^*(.), V^*(.)$ such that for any feedback strategies $U(.), V(.)$ the following holds:

$$\gamma(x(t_0, x_0, U^*(.), V(.))) \leq \gamma(x(t_0, x_0, U^*(.), V^*(.))) \leq \gamma(x(t_0, x_0, U(.), V^*(.))) \quad (51)$$

This is summarized in the theorem on an 'alternative' from [10]:

Theorem 3 *For any closed set M and for any initial position* (t_0, x_0)*, one and only one of the following assertions is valid: 1) The value of the game is 0 and any* (U^*, V) *is a saddle point (V is any feedback strategy); 2) The value of the game is 1. Furthermore, the optimal strategies* U^* *and* V^* *are feedback strategies.*

The notion of u(v)-stable bridge is a very important one in the setting of [10]. Informally, a u(v)-stable bridge $W_0(W)$ is the set of all points (t_0, x_0) such that there exists an optimal strategy $U^*(V^*)$ that keeps the motion of the system departing from (t_0, x_0) inside (outside) $W_0(W)$ until M is reached (avoided).

In this problem setup it is possible to derive a closed form for the u-stable bridge. First set:

$$f_1(t, x) := \max_{u \in P} \min_{v \in Q} f(t, x, u, v) \quad (52)$$

$$f_2(t, x) := \min_{u \in P} \max_{v \in Q} f(t, x, u, v) \quad (53)$$

Consider solutions w_1 and w_2 to the following ODEs:

$$\dot{w_1}(t) = f_1(t, w_1(t)), w_1(\theta) = l_1 \quad (54)$$

$$\dot{w_2}(t) = f_1(t, w_2(t)), w_2(\theta) = l_2 \quad (55)$$

The u-stable bridge is the set:

$$W_0 := \{(t, x) \in T \times \mathbb{R} : t \in T_*, x \in [w_1(t), w_2(t)]\}$$

where $T_* = [\tau_*, \theta], \tau_* = sup\{t \in T : w_2(t) > w_1(t)\}$.

This construction has a simple and appealing geometric interpretation. Keep in mind that the state evolves in $\mathbb{R}$. Equations (54,55) describe the evolution (in reverse time) of the boundaries (l_1, l_2) of the target set M when both players adopt optimal control strategies (given by the argmax and argmin in equations (52,53)). Now, consider an initial state (t,x) in the relative interior of W_0. Then, apply any control strategy U until the state reaches the boundary of W_0.

From this point onwards apply the optimal control strategies to both players. Then, by construction of w_1 and w_2 the state slides along one of the boundaries of W_0 (w_1 or w_2) until it reaches l_1 or l_2, respectively, at time θ.

This construction can be extended to a MIMO problem when the target set is a closed ball centered at the origin since this involves working with a norm (1-dim function).

References

[1] Jean-Pierre Aubin. *Viability theory.* Birkhauser, 1991.

[2] Jean-Pierre Aubin and Helene Frankowska. *Set-valued analysis.* Birkhauser, 1990.

[3] Martino Bardi and I. Capuzzo-Dolcetta. *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Birkhauser, 1997.

[4] F. H. Clarke. *Optimization and Nonsmooth Analysis.* SIAM, 1990.

[5] F. H. Clarke and P. R. Wolenski. Control of systems to sets and their interiors. *Journal of Optimization Theory and Applications*, 88(1):3–23, 1996.

[6] C. Daws, A. Olivero, S. Tripakis, and S. Yovine. The tool kronos. In *In Hybrid Systems III, Verification and Control*, Lecture Notes in Computer Science, pages 1066–1080. Springer Verlag, 1996.

[7] L. C. Evans. *Partial Differential Equations.* Graduate Studies in Mathematics. American Mathematical Society, 1998.

[8] I. V. Girsanov. *Lectures on mathematical theory of extremum problems.* Springer, 1972.

[9] T. Henzinger, X. Nicollin, J. Sifakis, and S. Yovine. Symbolic model checking for real-time systems. *Information and Computation*, 111(2):193–244, 1994.

[10] N.N. Krasovskii and A.I. Subbotin. *Game-theoretical control problems.* Springer-Verlag, 1988.

[11] A. B. Kurzhanskii. *Advances in nonlinear dynamics and control : a report from Russia.* Birkhauser, 1993.

[12] A. B. Kurzhanskii and I. Valyi. *Ellipsoidal calculus for estimation and control.* Birkhauser, 1997.

[13] A. B. Kurzhanskii and P. Varaiya. Ellipsoidal techniques for reachability analysis. In N. Lynch and B. Krogh, editors, *Computation and control*, Lecture Notes in Computer Science, pages 202–214. Springer-Verlag, 2000.

[14] A. B. Kurzhanskii and P. Varaiya. Ellipsoidal techniques for reachability analysis: internal approximation. *Systems & Control Letters*, 41(3):201–11, 2000.

[15] A. B. Kurzhanskii and P. Varaiya. Dynamic optimization for reachability problems. *Journal of Optimization Theory & Applications*, 108(2):227–51, 2001.

[16] A. B. Kurzhanskii and P. Varaiya. Optimization methods for target problems of control. In *Proceedings of Mathematical Theory of Networks and Systems Conference*, 2002.

[17] E. Lee and P. Varaiya. *The structure and interpretation of signals and systems.* Addison Wesley, 2002.

[18] G. Leitmann. Optimality and reachability with feedback controls. In G. Leitmann A. Blaquiere, editor, *Dynamic Systems and Microphysics.* Academic Press, 1982.

[19] L. S. Pontryagin, V. Boltyanskii, R. Gamkerelidze, and E. Mischenko. *The Mathematical Theory of Optimal Processes.* Interscience Publishers, 1962.

[20] R. Rockafellar and R. J. B. Wets. *Variational analysis.* Springer, 1998.

[21] A.I. Subbotin. *Generalized solutions of first-order PDEs : the dynamical optimization perspective.* Birkhauser, 1995.

[22] P. H. Ho T. Henzinger and H. Wong-Toi. Hytech: The next generation. In *Proceedings of IEEE Real-Time Systems Symposium, PTSS'95*, pages 169–183. IEEE Publications, 1995.

[23] P. Varaiya. Reach set computation using optimal control. In *Proceedings of the KIT Workshop on Verification of Hybrid Systems.* Verimag, Grenoble, France, 1998.

[24] S. Yovine. *Methods et Outils pour la Verification Symbolique de Systemes Temporises.* PhD thesis, Institut National Polytechnique de Grenoble – France, 1993.