A Formal Proof in Coq of a Control Function for the Inverted Pendulum

Damien Rouhling
Université Côte d'Azur, Inria
Sophia Antipolis, France
damien.rouhling@inria.fr

Abstract

Control theory provides techniques to design controllers, or control functions, for dynamical systems with inputs, so as to grant a particular behaviour of such a system. The inverted pendulum is a classic system in control theory: it is used as a benchmark for nonlinear control techniques and is a model for several other systems with various applications. We formalized in the Coq proof assistant the proof of soundness of a control function for the inverted pendulum. This is a first step towards the formal verification of more complex systems for which safety may be critical.

CCS Concepts • **Mathematics of computing → Ordinary differential equations**; *Topology*; • **Computer systems organization → Robotic control**; • **Software and its engineering → Software verification**;

Keywords Formal proofs, Coq, Control, Stability, Inverted pendulum

ACM Reference Format:
Damien Rouhling. 2018. A Formal Proof in Coq of a Control Function for the Inverted Pendulum. In *Proceedings of 7th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP'18).* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3167101

1 Introduction

The spread of the use of automated entities, namely robots or cyber-physical systems, to perform tasks considered as "hard" for human beings (e.g. transporting heavy objects, assembly line work, driving a car) raises security issues. Testing increases the confidence we have in such objects, but for critical applications it is not sufficient and proofs are necessary. Several aspects in their design must be taken into account for them to be granted as secure. As cyber-physical systems, they are controlled by programs whose flaws (e.g. bugs, approximations through floating-point arithmetic…) must be considered. They also interact with their environment and, besides the important questions behind the use of sensors, they can be considered as dynamical systems with inputs.

A goal of control theory is to design techniques with a strong mathematical foundation to control the behaviour of dynamical systems. A classic example used in control theory is the inverted pendulum (see Fig. 1), or cart and pole system. It is a pole which is weighted at one end, and which pivots around its other end. The pivot is on a cart which moves horizontally along a straight track. This system has two equilibria: the two positions of the pendulum where the pole is along the vertical line. The lower position is stable: if the position of the pole is slightly changed, it will move back to the equilibrium. The upper one is unstable: after any small perturbation of the pole's position, the pole will move further apart from the equilibrium.

The "control challenge" for this system is to move the cart in order to bring the pole to its (upper) unstable equilibrium [7, 25]. A program, or control function, controls the acceleration of the cart in order to reach this goal. This experiment is used as a benchmark for nonlinear control techniques. Moreover, it is a model for several other systems, for instance with applications in object transportation [11, 36] or to control the motion of bipedal systems [38].

In this paper we present a formalization[1] in the Coq proof assistant [39] of the proof of soundness of a control function for the inverted pendulum designed by Lozano et al. [25]. This function, presented as a force applied to the dynamical system of the pendulum, makes the pendulum converge to a homoclinic orbit, thus to a trajectory which brings the pendulum to its unstable equilibrium with zero angular frequency, and the cart to its starting point. The soundness of such a control function is the stability of the system, that is the convergence of the system to the desired region of space. For this formalization, we use the SSReflect exstension of Coq's tactic language [13], the Mathematical Components library[2] and the Coquelicot library [5].

While working on this formalization, we found a few mistakes in the mathematical proof by Lozano et al. [25]. Their

CPP'18, January 8–9, 2018, Los Angeles, CA, USA
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5586-5/18/01…$15.00
https://doi.org/10.1145/3167101

[1] https://github.com/drouhling/LaSalle
[2] https://math-comp.github.io/math-comp/

statement is however true, since we managed to correct all of them. In this paper we will follow the mathematical proof, highlighting in boldface the places where the mistakes were made and describing the solutions we found.

We first introduce the inverted pendulum and the underlying dynamical system (Sect. 2). Then we present the ideas behind the design of the control function from Lozano et al. (Sect. 3). This will give us all the material needed to describe the proof of stability of this system and to pinpoint the places where the formalization has to diverge from the paper proof (Sect. 4). Finally, we focus on some details of the formalization (Sect. 5).

2 The Inverted Pendulum

The proof of stability of the inverted pendulum is very dependent on how this system is characterised. In this section, we first state the differential equation of the system as given by the laws of physics and then we explain which form of equation was chosen by Lozano et al. [25] in order to prove the system's stability. We then state the soundness theorem.

2.1 The Physics of the System

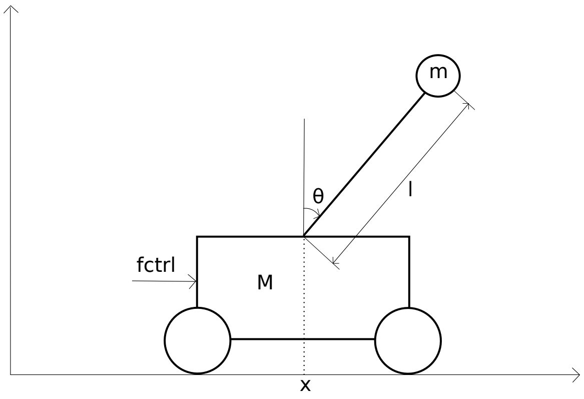

Figure 1. The inverted pendulum

The inverted pendulum is a weighted pole which has a pivot on a cart (Fig. 1). The cart moves horizontally along a line in order to bring the pole to its upper (unstable) equilibrium. We denote by m and M the respective masses of the pole's weight and of the cart, l the length of the pole, fctrl the control force (function) applied to the system, θ the angle that the pole forms with the vertical line and x the position of the cart. We also use the standard notation g for the gravitational acceleration on Earth.

Using Lozano et al.'s notations [25], if we define $q = \begin{pmatrix} x \\ \theta \end{pmatrix}$ the state of the system, $M(q) = \begin{pmatrix} M + m & ml\cos\theta \\ ml\cos\theta & ml^2 \end{pmatrix}$, $C(q, \dot{q}) = \begin{pmatrix} 0 & -ml\dot{\theta}\sin\theta \\ 0 & 0 \end{pmatrix}$, $G(q) = \begin{pmatrix} 0 \\ -mgl\sin\theta \end{pmatrix}$ and $\tau(q, \dot{q}) = \begin{pmatrix} \mathrm{fctrl}(q, \dot{q}) \\ 0 \end{pmatrix}$, then the differential equation that characterises this dynamical system is the following:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau(q, \dot{q}). \tag{1}$$

Note that the position x of the cart will only appear in the expression of the control function. Indeed, the physics of the (free) system does not depend on the position of the cart. However, it is a goal to bring back the cart to its starting position so that x must appear in the control term.

2.2 The System We Formalized

The goal of the control function is to bring this system to a homoclinic orbit, that is a trajectory which converges to the unstable equilibrium of the pendulum. In other terms, the purpose of this function is to make the unstable equilibrium of the (free) pendulum a *stable* equilibrium of the controlled system and, in addition, to make the system converge to this equilibrium.

For a dynamical system, converging to a stable equilibrium is called asymptotic stability [21]. This is an important notion for the control of nonlinear systems. A major tool to prove the stability of such dynamical systems is LaSalle's invariance principle [22]. It was used by Lozano et al. [25] to prove the convergence of their pendulum to a homoclinic orbit.

LaSalle's invariance principle requires the dynamical system to be described by a first-order autonomous differential equation, i.e. where the behaviour of the system only depends on its position. In order to rewrite (1) as such an equation, it is necessary to consider a state that contains more information:

$$z = \begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} = \begin{pmatrix} x \\ \dot{x} \\ \cos\theta \\ \sin\theta \\ \dot{\theta} \end{pmatrix}.$$

It is then possible to transform (1) into the following equation:

$$\dot{z} = \mathrm{Fpendulum}(z), \tag{2}$$

where, for all $p = \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix}$,

$$\mathrm{Fpendulum}(p) = \begin{pmatrix} p_1 \\ \frac{mp_3\left(lp_4^2 - gp_2\right) + \mathrm{fctrl}(p)}{M + mp_3^2} \\ -p_3p_4 \\ p_2p_4 \\ \frac{(M+m)gp_3 - p_2\left(mlp_4^2p_3 + \mathrm{fctrl}(p)\right)}{l\left(M + mp_3^2\right)} \end{pmatrix}.$$

However, with (2), we lose some pieces of information. For instance, since we work with points in $\mathbb{R}^5$, we forget that z_2 is a cosinus. It will then be necessary to pose constraints on the initial value $z(0)$ in order to keep the lost information as an invariant. For example, the equation $z_2(t)^2 + z_3(t)^2 = 1$ is to be proven for any time t.

2.3 Soundness of the Control Function

As explained in Sect. 2.2, the goal is to prove that for a well-chosen control function, the inverted pendulum is asymptotically stable. Instead of proving the convergence of the pendulum to its upper equilibrium, Lozano et al. [25] prove the convergence to a trajectory which converges to the equilibrium, called homoclinic orbit. This trajectory is characterised by the following differential equation:

$$\frac{1}{2}ml^2\dot{\theta}^2 = mgl\,(1 - \cos\theta)\,. \tag{3}$$

As we want the cart to stop at its starting point, on this trajectory we also want to have $x = 0$ and $\dot{x} = 0$. With the notations of Sect. 2.2, we want to prove that the solutions of (2) converge to the set of points p such that

$$p_0 = 0 \text{ and } p_1 = 0 \text{ and } \frac{1}{2}ml^2p_4^2 = mgl\,(1 - p_2)\,.$$

In Coq [39], this set is defined in a very similar way thanks to the use of notations for set comprehension and for the access to components of vectors.

```
Definition homoclinic_orbit :=
  [set p : 'rV[R]_5 | p[0] = 0 /\ p[1] = 0 /\
   (1 / 2) * m * (l ^ 2) * (p[4] ^ 2) =
     m * g * l * (1 - p[2])].
```

Here, `'rV[R]_5` is the type for vectors in $\mathbb{R}^5$ from the Mathematical Components library[3] (see Sect. 5.1 for more details).

Convergence to a trajectory, or more generally to a set, is an easy generalisation of the notion of convergence to a point (see Definition 2.1): it is sufficient to replace in the definition of convergence the distance to a point with the distance to a set.

Definition 2.1. A function of time $y(t)$ converges to a set A as t goes to infinity, denoted by $y(t) \to A$ as $t \to +\infty$, if

$$\forall \varepsilon > 0, \exists T > 0, \forall t > T, \exists p \in A, \|y(t) - p\| < \varepsilon.$$

The choice of control function by Lozano et al. implies constraints on the starting position of the system (see Sect. 3). The stability result from Lozano et al. can then be expressed as follows.

Theorem 2.2. *For some set K of starting positions and for a well-chosen control function* fctrl, *all solutions of (2) starting in K converge to the homoclinic orbit described by (3), together with the property that $x = 0$ and $\dot{x} = 0$, when time goes to infinity.*

As detailed in Sect. 5.3, we represent all the solutions of the differential equation (2) using a function `sol`. This function takes the initial positition of the inverted pendulum as input and computes the trajectory of the system. It also depends on the control function but for simplicity we do not display this dependency in our notations.

We state Theorem 2.2 in Coq through the following lemma, with the notation `f @ +oo --> A` to express the convergence of a function `f` to a set `A` when time goes to infinity. This notation was introduced in our previous work on LaSalle's invariance principle [8].

```
Lemma cvg_to_homoclinic_orbit (p : 'rV[R]_5) :
  K p -> sol p @ +oo --> homoclinic_orbit.
```

3 Design of the Control Function

The proof of stability of the inverted pendulum from Lozano et al. [25] goes through the use of LaSalle's invariance principle [22]. We first give the statement of the version of LaSalle's invariance principle we formalized. Then we explain how the use of this theorem affects the choice of the control function and the constraints imposed on the system.

3.1 LaSalle's Invariance Principle

LaSalle's invariance principle [22] is a tool to prove the asymptotic stability of the solutions of an autonomous system of differential equations in $\mathbb{R}^n$, thus an equation of the form:

$$\dot{y} = F \circ y \tag{4}$$

where y is a function of time and F is a vector field in $\mathbb{R}^n$.

The first aspect of asymptotic stability, i.e. the convergence of solutions of (4) to the equilibrium under some conditions, is expressed by LaSalle as convergence to a given region of space when time goes to infinity (recall Definition 2.1), and he gives examples where the properties of this region imply that it is the equilibrium.

LaSalle's invariance principle extends Lyapunov's second method [24] in order to study asymptotic stability. This method requires the existence of a Lyapunov function V that satisfies some properties. Theses properties are sign conditions on the derivative of V along the trajectories of the solutions of (4) starting in a given compact set K.

The choice of the Lyapunov function V affects the choice of the compact set K, in the sense that the sign conditions impose constraints on K. It also affects the set to which the solutions will converge, when it is not reduced to the equilibrium of the system, since it depends on K. Indeed, we proved in a previous work [8] a generalisation of LaSalle's invariance principle where this set is the union of the positive limiting sets, i.e. the sets of limit points (see Definition 3.1), of all the solutions starting in K.

[3] https://math-comp.github.io/math-comp/

Definition 3.1. Let y be a function of time. The positive limiting set of y, denoted by $\Gamma^+(y)$, is the set of all points p such that

$$\forall \varepsilon > 0, \forall T > 0, \exists t > T, \|y(t) - p\| < \varepsilon.$$

The other aspect of asymptotic stability is the fact that the solutions of (4) do not go too far from the equilibrium. Most often, there is a perturbation from the equilibrium that makes the solutions diverge from it. It can also be hard to determine a basin of attraction of this equilibrium. Fortunately, in order to use LaSalle's invariance principle it is sufficient to find a region around the equilibrium which is invariant with respect to (4) (see Definition 3.2). In our case, the compact set K will be such a region.

Definition 3.2. A set A is said to be invariant with respect to a differential equation $\dot{y} = F \circ y$ if every solution of this equation starting in A remains in A.

We improved our formalization of LaSalle's invariance principle [8] by relaxing an hypothesis on the Lyapunov function V. In this previous work, we supposed this function was differentiable in K. In fact it is only required to be continuous on K and such that it is differentiable along trajectories of solutions starting in K. In other terms the composition of V with any solution starting in K is differentiable at any time.

Among the hypotheses of LaSalle's invariance principle, there is the existence and uniqueness of the solutions of (4) on K. We will then denote by sol_p the unique solution of (4) with initial condition $p \in K$, and $\tilde{V}(p)$ the derivative of $V \circ \mathrm{sol}_p$ at time 0. We proved the following result, illustrated by Fig. 2.

Theorem 3.3. *Assume F is such that we have the existence and uniqueness of solutions of (4) and the continuity of solutions relative to initial conditions on an invariant compact set K. Suppose there is a scalar function V, continuous on K, such that for all point $p \in K$, $V \circ \mathrm{sol}_p$ is differentiable at any time, and $\tilde{V}(p) \leqslant 0$. Let S be the set of all points $p \in K$ such that $\tilde{V}(p) = 0$ and L be the union of all $\Gamma^+(y)$ for y solution starting in K. Then, L is an invariant subset of S and for all solution y starting in K, $y(t) \to L$ as $t \to +\infty$.*

Imposing a sign condition on the derivative of $V \circ \mathrm{sol}_p$ at time 0 is sufficient since we have the important property

$$\forall p \in K, \forall s\, t, \mathrm{sol}_p(s + t) = \mathrm{sol}_{\mathrm{sol}_p(s)}(t), \tag{5}$$

so that sol_p has for value and derivative at time s the value and derivative of $\mathrm{sol}_{\mathrm{sol}_p(s)}$ at time 0. Thus, we get the sign condition on the derivative of $V \circ \mathrm{sol}_p$ at any time as soon as it exists (which is assumed among our hypotheses). As a consequence, LaSalle's invariance principle in fact proves that V is minimised along the trajectories of solutions of (4).

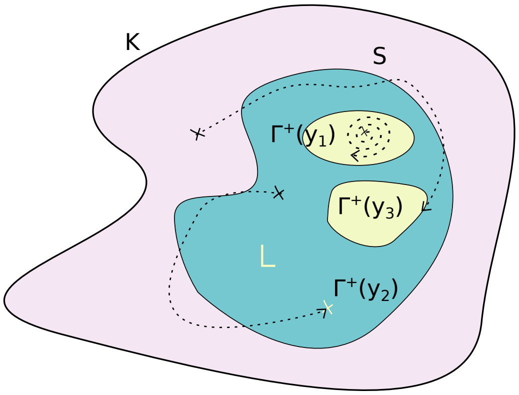

Figure 2. Illustration of the improved version of LaSalle's invariance principle

3.2 Constraints on the System

The challenge behind the use of LaSalle's invariance principle [22] is to find an appropriate Lyapunov function V so that being an invariant subset of $S = \left\{p \in K \mid \tilde{V}(p) = 0\right\}$ grants the desired properties. In the case of Lozano et al. [25], the goal is to converge to a homoclinic orbit. In order to achieve this goal, they choose an energy approach. With the notations of Sect. 2.1, the energy of the system is

$$E(q, \dot{q}) = \frac{1}{2}\dot{q}^T M(q)\,\dot{q} + mgl(\cos\theta - 1), \tag{6}$$

so that at the unstable equilibrium the energy is null. Conversely, when $E(q, \dot{q}) = 0$ and $\dot{x} = 0$, (6) becomes (3), thus the equation of a homoclinic orbit.

Thus, it is sufficient to find a Lyapunov function V such that LaSalle's invariance principle proves the convergence to a set where $E = 0$, $\dot{x} = 0$ and $x = 0$. As mentioned in Sect. 3.1, V is minimised along the trajectories hence there is the obvious choice

$$V(q, \dot{q}) = \frac{k_E}{2}E^2(q, \dot{q}) + \frac{k_v}{2}\dot{x}^2 + \frac{k_x}{2}x^2 \tag{7}$$

with k_E, k_v and k_x positive constants.

With the notations of Sect. 2.2, (7) becomes

$$V(z) = \frac{k_E}{2}E^2(z) + \frac{k_v}{2}z_1^2 + \frac{k_x}{2}z_0^2.$$

An important assumption to be proven is that $\tilde{V}(p) \leqslant 0$ for all point p in the compact set K still to be defined. Computation shows that when z is a solution of (2), the derivative of $V \circ z$ is

$$z_1\left(\mathrm{fctrl}(z)\left(k_E E(z) + \frac{k_v}{M+mz_3^2}\right) + \frac{k_v m z_3\left(l z_4^2 - g z_2\right)}{M+mz_3^2} + k_x z_0\right),$$

so that the control function

$$\mathrm{fctrl}(p) = \frac{k_v m p_3\left(g p_2 - l p_4^2\right) - \left(M + m p_3^2\right)(k_x p_0 + k_d p_1)}{k_v + \left(M + m p_3^2\right) k_E E(p)}$$

for k_d a positive constant gives $\tilde{V}(p) = -k_d p_1^2 \leqslant 0$.

This choice of control function imposes several constraints on the system and on the compact set K. First of all, this function needs to be well-defined and smooth in K in order to have the existence and uniqueness of solutions of (2) and the continuity of solutions relative to initial conditions on K. Moreover, it should not drive the system outside of its domain of definition. Thus, we need to prove that for any solution z starting in K we have $k_v + \left(M + mz_3^2\right) k_E E(z) \neq 0$. Knowing that z_3 represents a sinus (hence $z_3^2 \leqslant 1$) and that K should be invariant, it will be sufficient to have $|E(p)| < \frac{k_v}{k_E(M+m)}$ for all $p \in K$.

Then, to avoid converging to the stable equilibrium of the pendulum, where $E(p) = -2mgl$, we want to ensure that $|E(p)| < 2mgl$ for $p \in K$. Overall, we want that

$$|E(p)| < b = \min\left(\frac{k_v}{k_E(M+m)}, 2mgl\right).$$

It is then sufficient to have $V(p) < B = \frac{k_E b^2}{2}$ for $p \in K$ in order to prove the constraint on E. **This constant corrects the one from Lozano et al., where k_E was forgotten**. Finally, as mentioned in Sect. 2.2, since we forget that p_2 represents a cosinus and p_3 a sinus, we also need to ensure that $p_2^2 + p_3^2 = 1$, which leads to the following definition of the compact set K:

$$K = \left\{p \in \mathbb{R}^5 \mid p_2^2 + p_3^2 = 1 \text{ and } V(p) \leqslant k_0\right\}$$

where k_0 is a constant such that $k_0 < B$.

4 Stability Proof

In order to prove the stability of this inverted pendulum, we have first to show that the system satisfies the hypotheses of the version of LaSalle's invariance principle [22] exposed in Sect. 3.1. Then, we prove that this theorem indeed grants the convergence of the inverted pendulum to the homoclinic orbit given by (3) together with the property that the cart's position and speed are null. We follow here the proof by Lozano et al. [25], briefly giving the missing justifications for the use of LaSalle's invariance principle, and highlighting the places where the proof by Lozano et al. was erroneous.

4.1 Verification of the Hypotheses of LaSalle's Invariance Principle

The first step in the verification of the inverted pendulum from Lozano et al. [25] is to check that the system is well-defined. Looking at the form of Fpendulum in (2), it is sufficient to prove that for all $p \in K$, $M + mp_3^2 \neq 0$ and fctrl(p) is well-defined. The first point is in fact true for any p. The control function is well-defined at points such that its denominator is non zero. Following the reasoning in Sect. 3.2, we get that fctrl is well-defined on K. Hence, the system is well-defined on K.

Then, we need to prove the hypotheses of Theorem 3.3. First, we admit the existence and uniqueness of solutions of (2) and the continuity of solutions relative to initial conditions on K. A way to prove these properties is to use the Cauchy-Lipschitz Theorem (also known as the Picard-Lindelöf Theorem). However, the only formalization of this theorem in the Coq proof-assistant [39] we are aware of is the one by Makarov and Spitters [27]. It is based on the CoRN library of constructive real numbers [9], while our formalization of the inverted pendulum is based on the Coquelicot library [5], which extends Coq's standard library on classically axiomatized real numbers [29].

Then, the set K is compact, since it is closed and bounded and we are in finite dimension (we work in $\mathbb{R}^5$). The remaining hypotheses of Theorem 3.3 are

Hypothesis 1 K is invariant.

Hypothesis 2 For all $p \in K$, $V \circ \text{sol}_p$ is differentiable at any time.

Hypothesis 3 For all $p \in K$, $\tilde{V}(p) \leqslant 0$.

While proving the invariance of K, **we found a circular dependency between two properties in the proof by Lozano et al., none of them being proven in the end**. Indeed, given a point $p \in K$, the goal is to prove that for all t, $\text{sol}_p(t) \in K$. This decomposes into two parts:

$$\left(\text{sol}_p(t)\right)_2^2 + \left(\text{sol}_p(t)\right)_3^2 = 1$$

and

$$\left(V \circ \text{sol}_p\right)(t) \leqslant k_0.$$

The former is easy but in order to prove the latter, the authors argue that $V \circ \text{sol}_p$ is non increasing, which concludes the proof since $\text{sol}_p(0) = p$ and $p \in K$. However, to show that $V \circ \text{sol}_p$ is non increasing, it is necessary that fctrl $\circ\, \text{sol}_p$ is well-defined, hence that sol_p stays in K.

We managed to eliminate this circular dependency and to prove the three remaining hypotheses by proving Lemma 4.1.

Lemma 4.1. *For all $p \in K$, for all time t, the derivative at time t of $V \circ \text{sol}_p$ is $-k_d\left(\text{sol}_p(t)\right)_1^2$.*

Let us first check that Lemma 4.1 indeed implies the three remaining hypotheses. First, Hypothesis 2 is an obvious corollary of Lemma 4.1. Then, Hypothesis 3 is true because by definition $\tilde{V}(p)$ is the derivative at time 0 of $V \circ \text{sol}_p$. Finally, Lemma 4.1 implies that if $p \in K$, then $V \circ \text{sol}_p$ is non increasing, which proves Hypothesis 1 as discussed above.

In the remaining of this section, we explain how we managed to eliminate the circular dependency in order to prove Lemma 4.1. In fact, given a time t and a point $p \in K$, the derivative of $V \circ \text{sol}_p$ at time t has indeed the desired value as soon as fctrl is well-defined at point $\text{sol}_p(t)$. Instead of proving that fctrl is well-defined on K, we noticed that the reasoning in Sect. 3.2 proves the more precise Lemma 4.2.

Lemma 4.2. *The control function* fctrl *is well-defined at the points p such that $p_3^2 \leqslant 1$ and $V(p) < B$.*

Then, since the set $\left\{p \in \mathbb{R}^5 \mid p_2^2 + p_3^2 = 1\right\}$ is invariant, it is possible to show that Lemma 4.2 implies Lemma 4.3.

Lemma 4.3. *For all $p \in K$ and all time t, the derivative at time t of $V \circ \mathrm{sol}_p$ is $-k_d \left(\mathrm{sol}_p\,(t)\right)_1^2$* ***if*** $\left(\mathbf{V} \circ \mathbf{sol_p}\right)\mathbf{(t) < B}$.

Using Lemma 4.3, it is sufficient to prove Lemma 4.4 in order to get Lemma 4.1.

Lemma 4.4. *For all $p \in K$ and all time t, $\left(V \circ \mathrm{sol}_p\right)(t) < B$.*

It might seem that we are doing the same reasoning as Lozano et al.: we use the expression of the derivative of $V \circ \mathrm{sol}_p$ in order to prove that $V \circ \mathrm{sol}_p$ stays below a given bound. Moreover, proving that the bound is correct is necessary to prove the validity of the expression for this derivative. However, there is here a crucial difference with the previous goal: now we have a *strict* inequality in Lemma 4.4 compared to the *large* inequality in the definition of K.

This allows us to do the following proof. Consider s the greatest lower bound of the set A of times at which the condition is not satisfied: $s = \mathrm{glb}\left\{t \in \mathbb{R}^+ \mid B \leqslant \left(V \circ \mathrm{sol}_p\right)(t)\right\}$. The goal is to prove that $s = +\infty$. Since $0 \leqslant s$, it is sufficient to prove that s cannot be finite. In the case where s is finite, it is possible to show that s is in fact a minimum. This is where having a *large* inequality in the definition of s is important: if s is not a minimum, then we have the *strict* inequality $\left(V \circ \mathrm{sol}_p\right)(s) < B$, which allows us to build by continuity of $V \circ \mathrm{sol}_p$ a lower bound of A which is greater than s.

We can also prove that s is positive. Moreover, for all time $0 < t < s$, since we know that $\left(V \circ \mathrm{sol}_p\right)(t) < B$ by definition of the greatest lower bound, the derivative of $V \circ \mathrm{sol}_p$ at time t is $-k_d \left(\mathrm{sol}_p\,(t)\right)_1^2$ by Lemma 4.3. Then, using the mean value theorem, we get that $\left(V \circ \mathrm{sol}_p\right)(s) \leqslant V(p)$, which is not possible since $B \leqslant \left(V \circ \mathrm{sol}_p\right)(s)$ and $V(p) < B$.

4.2 Convergence to the Homoclinic Orbit

Theorem 3.3 applied to the inverted pendulum proves the convergence of any solution of (2) starting in K to the set

$$L = \bigcup_{\substack{y \text{ solution of } (2)\\ \text{starting in } K}} \Gamma^+(y).$$

The goal is now to prove that L is included in the homoclinic orbit characterised by (3) with the additional property that for all $p \in L$, $p_0 = 0$ and $p_1 = 0$. As mentioned in Sect. 3.2, it is sufficient to prove that for all $p \in L$,

Goal 1 $p_0 = 0$.
Goal 2 $p_1 = 0$.
Goal 3 $E(p) = 0$.

Let then p be in L. Since L is an invariant subset of the set $S = \left\{p \in K \mid \tilde{V}(p) = 0\right\}$ by Theorem 3.3, we know that the derivative of $V \circ \mathrm{sol}_p$ at time 0 is null. Using (3) we prove that the derivative function of $V \circ \mathrm{sol}_p$ is the identically zero function. As proven in Sect. 4.1, this derivative function is also $t \mapsto -k_d \left(\mathrm{sol}_p\,(t)\right)_1^2$. From this, we deduce Lemma 4.5.

Lemma 4.5. *For all $p \in L$,*

1. *$t \mapsto \left(\mathrm{sol}_p\,(t)\right)_1$ is the identically zero function.*
2. *$t \mapsto \left(\mathrm{sol}_p\,(t)\right)_0$ is constant.*
3. *$E \circ \mathrm{sol}_p$ is constant.*
4. *$t \mapsto \left(\left(\mathrm{Fpendulum} \circ \mathrm{sol}_p\right)(t)\right)_1$.*

In particular, from point 1 of Lemma 4.5 at time 0 we get Goal 2. From all this, we can also derive the important equations

$$\forall t, k_E \left(E \circ \mathrm{sol}_p\right)(t) \left(\mathrm{fctrl} \circ \mathrm{sol}_p\right)(t) + k_x \left(\mathrm{sol}_p\,(t)\right)_0 = 0, \quad (8)$$

and

$$\forall t, \left(\mathrm{sol}_p\,(t)\right)_3 \left(g \left(\mathrm{sol}_p\,(t)\right)_2 - l \left(\mathrm{sol}_p\,(t)\right)_4^2\right) = \frac{\left(\mathrm{fctrl} \circ \mathrm{sol}_p\right)(t)}{m}. \quad (9)$$

We know by Lemma 4.5 that $E \circ \mathrm{sol}_p$ is constant (equal to $E(p)$). Either this constant is zero or it is not. If it is zero, i.e. Goal 3 is true, from (8) at time 0 we prove Goal 1, which ends the proof.

If $E(p) \neq 0$, we want to derive a contradiction by proving that it implies that $E(p) = 0$, or equivalently (thanks to Lemma 4.5) that $E \circ \mathrm{sol}_p$ is the identically zero function. From (8) and points 2 and 3 of Lemma 4.5 we prove Lemma 4.6.

Lemma 4.6. *For all $p \in L$ such that $E(p) \neq 0$, $\mathrm{fctrl} \circ \mathrm{sol}_p$ is constant.*

Until this point, we followed the proof from Lozano et al. [25]. **But then the authors present an erroneous (and unnecessary) proof that $\mathrm{fctrl} \circ \mathrm{sol}_p$ is the identically zero function. They draw from that, again through an erroneous proof, the conclusion that $E \circ \mathrm{sol}_p$ is the identically zero function.**

In order to understand their mistake, it is important to know that Lemma 4.7 is a significant tool for the remaining of the proof. If an equation e between two functions is true on an interval, we will call "derivative of e" the equation obtained from e using Lemma 4.7.

Lemma 4.7. *Let I be an interval and f and g be two real functions differentiable on I.*

If for all $t \in I$, $f(t) = g(t)$, then for all $t \in I$, $\dot{f}(t) = \dot{g}(t)$.

Taking the derivative of (9) and using Lemma 4.6, we get the following equation

$$\forall t, \left(\mathrm{sol}_p\,(t)\right)_4 \left(3g \left(\left(\mathrm{sol}_p\,(t)\right)_2^2 - \left(\mathrm{sol}_p\,(t)\right)_3^2\right) + C \left(\mathrm{sol}_p\,(t)\right)_2\right) = 0, \quad (10)$$

where C is a constant depending on $E(p)$.

Then, for a given time t, if $\left(\mathrm{sol}_p\,(t)\right)_4 \neq 0$, then we get from (10) that

$$3g \left(\left(\mathrm{sol}_p\,(t)\right)_2^2 - \left(\mathrm{sol}_p\,(t)\right)_3^2\right) + C \left(\mathrm{sol}_p\,(t)\right)_2 = 0. \quad (11)$$

The authors then proceed to take the derivative of this new equation, which is incorrect because an equation has to be valid on an interval for one to be allowed to take its derivative. In this particular case, it is still possible to take the derivative of (11) because it is true on a small interval containing t. Indeed, when $\left(\mathrm{sol}_p\ (t)\right)_4 \neq 0$, since sol_p is continuous one can find a positive real number ε such that for all $s \in]t-\varepsilon, t+\varepsilon[$, we have $\left(\mathrm{sol}_p\ (s)\right)_4 \neq 0$. However, they repeat several times this mistake by considering that when a component of sol_p is null at some time t, the derivative of this component also must be null at time t. This cannot be corrected by such an easy argument of continuity.

While looking for a way to correct the proof from Lozano et al., we found a way to simplify the proof that $E \circ \mathrm{sol}_p$ is the identically zero function. It does not require proving that $\mathrm{fctrl} \circ \mathrm{sol}_p$ is the identically zero function, only that it is constant (in order to use (10)), which we already know by Lemma 4.6.

The first step is to prove the following equation, which we obtain from (9) and (13):

$$\forall t, \left(\mathrm{Fpendulum}\left(\mathrm{sol}_p\ (t)\right)\right)_4 = \frac{g}{l}\left(\mathrm{sol}_p\ (t)\right)_3. \qquad (12)$$

$$\forall t, \left(\mathrm{sol}_p\ (t)\right)_2^2 + \left(\mathrm{sol}_p\ (t)\right)_3^2 = 1. \qquad (13)$$

Equation (12) has several consequences. In particular, we can prove Lemma 4.8, from which we deduce Lemma 4.9 using (2), (13) and (12).

Lemma 4.8. *For all $p \in L$ such that $E\ (p) \neq 0$, the component functions $t \mapsto \left(\mathrm{sol}_p\ (t)\right)_2$ and $t \mapsto \left(\mathrm{sol}_p\ (t)\right)_3$ are constant.*

Lemma 4.9. *For all $p \in L$ such that $E\ (p) \neq 0$,*

1. *$t \mapsto \left(\mathrm{sol}_p\ (t)\right)_4$ is the identically zero function.*
2. *$t \mapsto \left(\mathrm{sol}_p\ (t)\right)_3$ is the identically zero function.*
3. *For all time t, $\left(\mathrm{sol}_p\ (t)\right)_2$ is either 1 or −1.*

From (6), point 1 of Lemma 4.5 and point 1 of Lemma 4.9, we know that

$$\forall t, \left(E \circ \mathrm{sol}_p\right)(t) = mgl\left(\left(\mathrm{sol}_p\ (t)\right)_2 - 1\right). \qquad (14)$$

Hence, in order for $E \circ \mathrm{sol}_p$ to be the identically zero function, it is sufficient to prove that

$$\forall t, \left(\mathrm{sol}_p\ (t)\right)_2 = 1.$$

By point 3 of Lemma 4.9, it is sufficient to prove that $\left(\mathrm{sol}_p\ (t)\right)_2 = -1$ is not possible. When $\left(\mathrm{sol}_p\ (t)\right)_2 = -1$, with (14) we get the equation $\left(E \circ \mathrm{sol}_p\right)(t) = -2mgl$, which contradicts the condition $|E\ (p)| < b$ from Sect. 3.2 ($p \in K$ and $E \circ \mathrm{sol}_p$ is constant).

All in all, this gives a (correct) simpler proof that all solutions of (2) starting in K converge to a set L included in the homoclinic orbit characterised by (3), with the additional property that for all $p \in L$, $p_0 = 0$ and $p_1 = 0$. We end this section with an idea of the proof of Lemma 4.8.

Both points of Lemma 4.8 can be proven using Lemma 4.10. Indeed, using (12) and (13) in (10) and its derivative, and discussing the cases where $\left(\mathrm{sol}_p\ (t)\right)_4 \neq 0$ or $\left(\mathrm{sol}_p\ (t)\right)_4 = 0$, we get two polynomials of degree two whose coefficients do not depend on t and such that $\left(\mathrm{sol}_p\ (t)\right)_2$ is a root of one of them. Thus, the component function $t \mapsto \left(\mathrm{sol}_p\ (t)\right)_2$ can have only a finite number of different values and, by Lemma 4.10, is constant. Similarly, using (13) we prove that the component function $t \mapsto \left(\mathrm{sol}_p\ (t)\right)_3$ can have only two different values, since $t \mapsto \left(\mathrm{sol}_p\ (t)\right)_2$ is constant. It is thus constant by Lemma 4.10.

Lemma 4.10. *Let I be an interval and f be a real function.*

If f is continuous on I and if f can only take a finite number of different values on I, then f is constant on I.

5 Formalization

In this section, we give details on the formalization of the proof we present in Sect. 4. We start by giving some context on the libraries and data structures we use. Then, we discuss questions of topology we had to face. Then, we explain how we deal with differential equations. Finally, we explain how we designed a way to compute derivatives and differentials automatically in order to simplify proofs.

5.1 Context and Data Structures

We formalized the proof of stability of the inverted pendulum from Lozano et al. [25] in the Coq proof assistant [39]. Our proof scripts are written using the SSReflect extension of Coq's tactic language [13], which usually gives compact proof scripts. Our development extends our previous work on LaSalle's invariance principle [8] with around 2500 lines of code, where only 1000 lines are devoted to the definition and the proof of the system.

For real analysis, we rely on the Coquelicot library [5], which is a conservative extension of Coq's standard library on real numbers [29]. The main advantage of Coquelicot over the standard library is its filter-based theory of convergence, inspired by the analysis library of Isabelle/HOL [18].

In topology, a filter is a nonempty set of sets, closed under some operations. The most important filters we use are the set of neighbourhoods of a point p, denoted by `locally p` in Coquelicot, and the set of neighbourhoods of +∞. In Coquelicot, the standard characterization of the convergence of a function f at point p to a point q (p and q can also be $\pm\infty$) is abstracted as the inclusion in the neighbourhood filter of q of the image by f of the neighbourhood filter of p. This abstraction proved its efficiency both in Coquelicot and Isabelle/HOL.

Coquelicot also contains a hierarchy of algebraic structures (from abelian groups to complete normed modules) and of topological structures (uniform spaces and complete spaces). This hierarchy is developped using canonical structures [26]. Coquelicot equips Coq's type R of real numbers with all theses structures. In our case, since the system described in Sect. 2.2 is in $\mathbb{R}^5$, we have first to choose a data type which represents vectors in $\mathbb{R}^5$ and to equip this type with Coquelicot's structures.

We follow the example of Paşca's work on multivariate analysis [33, 34] and use the structure of vectors from the Mathematical Components library[4] (see [33] for a discussion on the different possibilities for representing $\mathbb{R}^n$ in Coq). A point in $\mathbb{R}^n$ is thus represented as an element of the type `'rV[R]_n` of row vectors on R of length n. A row vector of length n is a matrix with one line and n columns. Matrices with m rows and n columns are represented as functions from `'I_m * 'I_n` to the type of the coefficients, where `'I_p` is the type of natural numbers k such that k < p. The main difference with Paşca's work is that we have access to the Coquelicot library (and that the Mathematical Components library has evolved since), so that we work in a more convenient framework to do multivariate analysis where we can reuse many theorems already proven.

Still, we have to equip the type `'rV[R]_n` with Coquelicot's structures so that they are automatically inferred where needed. We can prove that when a type T has a certain structure, the type `'rV[T]_n` canonically inherits this structure. This is the matter of a few 500 lines, using straightforward definitions such as componentwise addition for the abelian group structure, or componentwise multiplication by an element of a ring *A* for the *A*-module structure. However, difficulties arise with the definition of a norm over the type of vectors. The maximum norm is a natural choice and using Mathematical Components' library for iterated operators over an indexed set (big operators) is even more natural.

```
Definition vnorm (x : 'rV[T]_n) :=
  \big[Rmax/0]_i (norm (x ord0 i)).
```

In this definition, `(x ord0 i)` is the ith component of the vector x and we iterate the maximum operator Rmax. We also have to give a default element for the case where n is 0, which in our case is 0 since we manipulate non negative numbers. The main difficulty with this definition comes from the fact that some theorems on big operators require some algebraic structure on the operator, in particular the existence of a neutral element. The maximum operator on real numbers does not admit such a structure.

However, since we consider only non negative numbers, 0 is a neutral element. Paşca suggests two possibilities [34]: to build the type of non negative real numbers, equip it with the right structure, and use the theorems in Mathematical Components' library for big operators, or to define a new maximum operator which has the right structure by definition and such that it coincides with Rmax on non negative numbers and move from one operator to the other through rewriting. We experimented with the second possibility, which was Paşca's choice, but we eventually opted for a third choice of implementation which revealed to be shorter: to prove again the theorems from the Mathematical Components library that require some algebraic structure, instantiated on real numbers together with the additional hypothesis that we only consider families of non negative numbers. This is however not a long term solution.

The first possibility mentioned by Paşca seems to be the best way in the long term but developing a theory of non negative real numbers in Coq is not so obvious. For instance, if R^+ is a new type built on top of the type R, it inherits from R its monoid structure but proving this in Coq requires a redefinition of addition as an operation on R^+ and a proof that this new addition is associative and that the 0 of R^+ (which is not the same as the 0 of R) is an identity for this operation. This is a duplication of the code we would like to avoid.

5.2 Topology

Topology is an important topic of this work. Indeed, we heavily rely on filters, we use LaSalle's invariance principle [22] which is a theorem of convergence and we work with compact sets. However, we do not really commit to doing topology. For instance, we speak of compact *sets* instead of compact *spaces*, thus avoiding the use of the subspace topology. A reason for this is that the Coquelicot library [5] does not deal with topological spaces. In fact, the topological structure which is at the base of Coquelicot's hierarchy is the structure of uniform space and not all topological spaces are uniform spaces: only those which are completely regular are uniform spaces. Coquelicot uses a pseudometrics definition of uniform spaces: a uniform space is a type U equipped with a predicate `ball : U -> R -> U -> Prop` satisfying

- reflexivity:

  ```
  forall x (e : posreal), ball x e x,
  ```

- symmetry:

  ```
  forall x y e, ball x e y -> ball y e x,
  ```

- and the triangular inequality:

  ```
  forall x y z e1 e2,
  ball x e1 y -> ball y e2 z ->
  ball x (e1 + e2) z.
  ```

In this context, the neighbourhood filter of a point `p : U`, called `locally p` as mentioned in Sect. 5.1, is easily defined in Coquelicot: a set A is a neighbourhood of p if it contains a ball centered on p and of positive radius.

```
Definition locally (p : U) (A : set U) :=
  exists e : posreal, ball p e `<=` A.
```

[4] https://math-comp.github.io/math-comp/

In our previous formalization of LaSalle's invariance principle [8], we used this notion of neighbourhood to define the notion of compactness in terms of filters. This definition involves the notion of clustering: a filter clusters to a point if each of its elements intersects each neighbourhood of the point. A set A is then said to be compact if every proper filter on A clusters to a point of A, where a proper filter is a filter which does not contain the empty set.

```
Definition cluster (F : set (set U)) (p : U) :=
  forall A B, F A -> locally p B ->
  A `&` B !=set0.
Definition compact (A : set U) :=
  forall (F : set (set U)), F A ->
  ProperFilter F -> (A `&` cluster F) !=set0.
```

Here, we use the notation `A `&` B` for the intersection of A with B and `A !=set0` for `exists p, A p`.

In our formalization of the inverted pendulum from Lozano et al. [25], we prove that the set K is compact by proving that it is closed and bounded, as explained in Sect. 4.1. A closed and bounded set is compact in a finite dimensional space because it is a closed subset of a compact set (the finite product of segments defined by its bound). We have thus to prove that a finite product of compact sets in $\mathbb{R}$ is compact. We decided to formalize the more general Tychonoff Theorem, which admits a simple proof thanks to filters, although it requires the axiom of choice in the form of Zorn's Lemma (we used the version of Zorn's Lemma contained in Schepler's small library on set theory[5]).

Tychonoff's Theorem states that a product of compact topological spaces is compact. Its proof uses the definition of compactness in terms of ultrafilters. An ultrafilter is a proper filter which is maximal for set inclusion. A set A is compact if and only if every ultrafilter on A converges to a point in A, where "F converges to p" means "F contains the neighbourhood filter of p". What makes the proof simple is the fact that, when given a product space $U = \prod_{i \in I} U_i$, an ultrafilter F on U and $p \in U$, F converges to p if and only if for all $i \in I$, the ultrafilter $F_i = \{\pi_i(A) \mid A \in F\}$ converges to $\pi_i(p)$, where π_i is the canonical projection to U_i. When each U_i is compact, we have by definition the convergence of F_i to some p_i and thus we can build a p to which F converges.

However, this proof is not possible if we stay in the world of uniform spaces, because it actually exploits the properties of the product topology. To be more precise, it uses separately the properties of the weak topology and of the supremum topology, which are combined to define the product topology. The product topology on $\prod_{i \in I} U_i$ is the supremum topology of the weak topologies by each projection π_i on U_i (see for instance Wilansky's textbook on topology [40] for the details). In the world of uniform spaces, we are stuck with the uniform topology, which is induced by balls. This topology is semimetrizable, while the supremum topology of an uncountable family of semimetrizable topologies might not be semimetrizable. Thus we might be working with the wrong topology.

Instead, we formalize the notion of topological space and prove Tychonoff's Theorem in the context of topological spaces. A topological space is a type T equipped with a family of sets, called open sets, which is stable by union and by finite intersection and which contains the full set. A neighbour of a point p in T is a set A which is open and that contains p. As before, a filter clusters to a point if each of its elements intersects each neighbourhood of the point and a set A is said to be compact if every proper filter on A clusters to a point of A.

```
Definition neighbour (p : T) (A : set T) :=
  open A /\ A p.
Definition cluster (F : set (set T)) (p : T) :=
  forall A B, F A -> neighbour p B ->
  A `&` B !=set0.
Definition compact (A : set T) :=
  forall (F : set (set T)), F A ->
  ProperFilter F -> (A `&` cluster F) !=set0.
```

We can also prove that balls in a uniform space indeed induce a topology and that the compact sets for this topology are the compact sets according to the previous definition on uniform spaces. Moreover, the uniform topology on the finite product $\mathsf{U}^{\mathsf{n\ +\ 1}}$, where U is a uniform space, is the same topology as the product topology of the uniform topology on U. Thus we can swap from one definition of compactness to the other before using the specificities of the product topology through Tychonoff's Theorem. Finally, we get that a product of `n + 1` compact sets in U is a compact set of $\mathsf{U}^{\mathsf{n\ +\ 1}}$.

```
Lemma vect_prod_topologyP n
  (A : set 'rV[U]_n.+1) :
  @open uniformTopologicalSpace A <->
  @open (prodTopologicalSpace _ ord0) A.
Lemma vect_compact n (A : 'I_n.+1 -> set U) :
  (forall i, compact (A i)) ->
  compact [set v : 'rV[U]_n.+1 |
           forall i, A i (v ord0 i)].
```

Here, the @ notation allows us to specify the implicit argument, which is the topology in the case of the open predicate.

Note that there already exists a library for topology in Coq by Schepler[6], which contains a proof of Tychonoff's Theorem. However, this library has its own definitions of filters and uniform spaces and does not interface well with Coquelicot. For instance, Schepler's filters are Coquelicot's proper filters and Schepler's formalization of uniform

[5] https://github.com/coq-contribs/zorns-lemma

[6] https://github.com/coq-contribs/topology

spaces implies that these are metric spaces while Coquelicot's uniform spaces are not necessarily metric. Moreover, in Schepler's library the definition of compactness concerns topological spaces whereas, as mentionned before, we focus on subsets of such spaces without referring to the subspace topology.

5.3 On Differential Equations

As explained in Sect. 3.1, we generalised our previous version of LaSalle's invariance principle [8] by weakening some hypotheses. We also changed the formal definition of the notion of solution of a differential equation in order to take into account more equations. We present here the former definition of the notion of solution and the issues we pinpointed in our previous paper and we explain how we modified this definition for the present work. In our previous work, a function y was considered to be a solution of the differential equation (4) if its derivative function is equal to $F \circ y$.

```
Definition is_sol (y : R -> U) :=
  forall t, is_derive y t (F (y t)).
```

Here, `U` is the ambient space (e.g. `'rV[R]_5` in the particular case of the inverted pendulum) and `R` the set of real numbers. But in practice we are only interested in systems which have a physical interpretation, or in other terms in functions of time. Some equations might even have no meaning for negative values of `t`. Thus, we would like to work with functions whose domain is the set of non negative real numbers. This would require constructing the type for this set and to develop its theory. An easy and lightweight compromise is to keep functions on $\mathbb{R}$, but only require the solution to satisfy the differential equation for non negative times.

```
Definition is_sol (y : R -> U) :=
  forall t, 0 <= t -> is_derive y t (F (y t)).
```

However, this definition is incompatible with our way to express the existence and uniqueness of solutions of (4). Indeed, as in our previous work, we assume the existence of a function `sol : U -> R -> U` which takes the initial condition as first argument and outputs the corresponding solution of (4). This choice of a function representing the unique solution is similar to the work on ordinary differential equations in Isabelle/HOL by Immler and Hölzl [20]. They use however Hilbert's ε operator to obtain the function corresponding to `sol`, which we cannot do in Coq [39].

In order to admit the existence and uniqueness of solutions and their continuity relative to initial conditions, we do the same assumptions as in our previous work. The fact that the first argument of `sol` is its initial value is expressed through the following assumption.

```
Hypothesis sol0 : forall p, sol p 0 = p.
```

The continuity of solutions relative to initial conditions in K is admitted through the following hypothesis: when we fix the second argument of `sol`, we get a function which is continuous on K.

```
Hypothesis sol_cont :
  forall t, continuous_on K (sol^~ t).
```

The existence and uniqueness of solutions is expressed through the combination of `sol0` and the following hypothesis: a function `y` starting in `K` is a solution of (4) if and only if it is equal to the function `sol (y 0)`, that is the solution which has same initial condition.

```
Hypothesis solP : forall y, K (y 0) ->
  is_sol y <-> y = sol (y 0).
```

This equality between functions, which we can use thanks to the axiom of functional extensionality, matches both the pen and paper and the SSReflect [13] proof-styles. Indeed, replacing a function by another one which is extensionaly equal to it is frequent in mathematics and can be done here through rewriting. Rewriting of equalities takes moreover an important part in proof scripts in the SSReflect tactic language. In addition, using the function `sol` in our statements instead of a function `y` together with the additional hypothesis `is_sol y` makes our statements more readable and simplifies our proofs.

As we explained in our previous work, `solP` is not satisfiable if we constrain the derivative of solutions only for non negative times, since there would be (infinitely) many solutions with the same initial value (the values for negative times are basically free). In order to keep the benefits of this formulation of `solP`, we decided to change the notion of solution rather than to adapt the formulation. We thus fix the values of solutions for negative times in a way which does not constrain the differential equation. It is always possible since we only want to state properties on values for non negative times. In order to keep the solutions derivable everywhere, we made them symmetric with regard to their initial value (i.e. $y(-t) = 2y(0) - y(t)$), which leads to the following definition in Coq:

```
Definition is_sol (y : R -> U) :=
  (forall t, t < 0 ->
    y t = minus (scal 2 (y 0)) (y (- t))) /\
  forall t, 0 <= t -> is_derive y t (F (y t)).
```

This definition does not make the function `sol` harder to use (there is only an assumption to discard) and we hope it will also be adapted for the proof of the Cauchy-Lipschitz Theorem. We expect a not much longer proof with a first step to build solutions without constraints on negative times and a second step to transform these solutions into solutions as characterised by `is_sol`. The only true drawback we encountered yet is that this definition significantly increases (around 50 additional lines) the size of the proof of (5).

Indeed, we cannot directly use the uniqueness of solutions to prove this property anymore because the "shifted solution" $t \mapsto \mathrm{sol}_p\,(s+t)$ where $s \in \mathbb{R}^+$ may not be a solution anymore since it may not be symmetric with regard to its initial value. It is first necessary to build a solution which coincides with the function $t \mapsto \mathrm{sol}_p\,(s+t)$ on $\mathbb{R}^+$ and to prove that it is indeed a solution. Only then we can use the uniqueness of solutions to conclude the proof.

5.4 Automated Differentiation

In our previous work on LaSalle's invariance principle [8], we encountered difficulties because of the way we deal with differentials in Coquelicot [5] (this also applies to derivative functions in Coq). To express the fact that a function `df` is the differential of a function `f` at point `p`, Coquelicot gives us access to a predicate `filterdiff f (locally p) df`. This is not very convenient to use in practice compared to the way we can deal with derivatives in Coquelicot.

Indeed, Coquelicot provides a total function `Derive f` to represent the derivative of a real function `f`. All theorems about the derivative function of `f` are then expressed using `Derive f`, with the hypothesis that `f` admits a derivative at point `x`, denoted by `ex_derive f x`, when it is needed. However, Coquelicot lacks such a total function for differentials. Thus, we have to provide the differential `df` (or to add it as a parameter when we assume `f` admits a differential) and in order to prove that `df` is the differential of `f` at point `p`, we have to put `df` in the right form in order to apply the rules of differentiation.

For instance, while formalizing the inverted pendulum from Lozano et al. [25], we noticed that several times we had a simplified form of a differential. To prove that it is indeed a differential we performed the same steps:

1. first prove that this differential can be written in a more expanded way,
2. then use the rules of differentiation to prove that this expanded form is the differential of a given function.

For example, in Sect. 4.2 we mentioned that the derivative function of $E \circ \mathrm{sol}_p$ is $t \mapsto \left(\mathrm{sol}_p\,(t)\right)_1 \left(\mathrm{fctrl} \circ \mathrm{sol}_p\right)(t)$. But in order to prove this fact it is first necessary to put this function under the form $t \mapsto dE_{\mathrm{sol}_p(t)} \left(\mathrm{Fpendulum}\left(\mathrm{sol}_p\,(t)\right)\right)$, where dE_q is the expanded form of the differential of E at point q. Then, applying the rule for the differentiation of a composition of functions, we have to prove that $dE_{\mathrm{sol}_p(t)}$ is the differential of E at point $\mathrm{sol}_p\,(t)$ and that $\mathrm{Fpendulum}\left(\mathrm{sol}_p\,(t)\right)$ is the derivative of sol_p at time t. The first goal is similarly proven, successively using different rules for differentiation (e.g. the first rule will be the one for the addition of two functions, see (6) for the form of E).

For the first step (expanding functions to put them in the right form), the proof of equality is already quite automated thanks to reflection-based decision procedures for the equality of terms such as `ring` [14] or `field` [10]. For the second step (using the rules of differentiation), the only automation provided in Coquelicot is for the case of functions from $\mathbb{R}$ to $\mathbb{R}$ [23].

We later learnt that reversing the order between these two steps is possible thanks to the `evar_last` tactic from Coquelicot and allows for a slight improvement. When the goal is of the form $P\ p_1 \ldots p_n$, with P a predicate and $p_1 \ldots p_n$ its parameters, this tactic replaces the last parameter with an existential variable $?v$, and the goal with the two goals $P\ p_1 \ldots ?v$ and $?v = p_n$. This allows us to use the rules of differentiation without having to provide the expanded form for the differential, since it will be inferred from the constraints these rules impose on $?v$. Our contribution is a way to automate this inference.

The automation of the computation of a differential for a given function is done by means of type classes [37]. We keep a database of differentiation rules thanks to a type class `diff` encapsulating the `filterdiff` predicate.

```
Class diff (f : U -> V) (F : set (set U))
  (df : U -> V) := diff_prf : filterdiff f F df.
```

When we want to prove that a function `df` is the differential of a given function `f`, we apply the following lemma.

```
Lemma diff_eq (f f' df : U -> V)
  (F : set (set U)):
  diff f F f' -> f' = df -> diff f F df.
```

Thanks to the SSReflect tactic language [13], `f'` is introduced as an existential variable and type class inference is triggered in a seamless way. Type class inference will then automatically compute for us the function `f'` and prove the assumption `diff f F f'`. Then we can prove that `f'` is equal to `df` using for instance the axiom of functional extensionality and the `ring` and `field` tactics.

In order for type class inference to succeed, it is necessary to have a well-stocked database of differentiation rules. We turned rules from Coquelicot into instances of the class `diff`. For example, the rule giving the differential of constant functions, or the one for the sum of two functions (the proofs are direct application of the corresponding lemmas from Coquelicot).

```
Instance diff_const (p : V) (F : set (set U)) :
  Filter F ->
  diff (fun _ => p) F (fun _ => zero).
Instance diff_plus (f g df dg : U -> V)
  (F : set (set U)) :
  Filter F -> diff f F df -> diff g F dg ->
  diff (fun p => plus (f p) (g p)) F
    (fun p => plus (df p) (dg p)).
```

Sometimes, the form of the differential is not interesting because it is sufficient to prove that this differential exists.

It is possible to prove automatically the existence of differentials by triggering type class inference thanks to the following lemma.

```
Lemma ex_diff (f df : U -> V)
  (F : set (set U)) :
  diff f F df -> ex_filterdiff f F.
```

We also have a similar mechanism for derivatives of functions whose domain is a ring with an absolute value (`AbsRing` structure in Coquelicot). We define a type class `deriv` encapsulating the `is_derive` predicate from Coquelicot. We prove two lemmas to trigger type class inference whenever we want to prove that an expression is the derivative of a given function at some point or to prove that such a derivative exists. And we build a database of rules easily extracted from the Coquelicot library.

```
Class deriv (f : K -> V) (x : K) (df : V) :=
  deriv_prf : is_derive f x df.
Lemma deriv_eq (f : K -> V) (x : K)
  (df' df : V) :
  deriv f x df' -> df' = df -> deriv f x df.
Lemma ex_deriv (f : K -> V) (x : K) (df : V) :
  deriv f x df -> ex_derive f x.
```

6 Related Work

Several formalizations on dynamical systems and control theory already exist. Important tools in this domain are the KeYmaera prover [35] and its successor, KeYmaera X [12]. They operate however on a quite different domain since they are based on differential dynamic logic. Moreover, KeYmaera and KeYmaera X use Mathematica as a trusted oracle for quantifier elimination.

Anand et al. [1] developped a framework to build certified programs in Coq for robots, ROSCoq. They followed an approach which is similar to ours: first define the physics of the system using differential equations and then prove properties on it. An important difference with our work is that they use the constructive real numbers from the CoRN library [9]. They are thus able to run the programs they certified, whereas it is impossible to do so using Coq's standard library for real analysis [29].

Another framework which is designed for cyber-physical systems is the VeriDrone project [28], developped in the Coq proof assistant [39]. However, it is based on a different logic (it uses an embedding of linear temporal logic in Coq) and also trusts external tools (SMT solvers). In this framework, Chan et al. [6] use a Lyapunov function to prove the stability of a particular system. They have however no proof of a general stability theorem and thus have to do a direct proof.

Herencia-Zapana et al. [17] take another approach to stability proofs: stability proofs using Lyapunov functions, under the form of Hoare triples annotations on C code, are used to generate proof obligations for PVS. By this means, they can directly prove properties on implementations instead of proving them on models of the systems. Mitra and Chandy [31] also formalize in PVS stability theorems using Lyapunov-like functions. They focus however on systems defined by automata while we work on differential equations.

Differential equations have been studied by Boldo et al. [3] and by Immler and Hölzl [20]. In both cases, they focus on numerical approximation schemes, whereas we are interested in the qualitative analysis of the equations: we prove properties on the solutions without finding them analytically and without computing approximations.

About multivariate analysis, besides Paşca's work [33, 34] which we mentioned before, we have access to another formalization in Coq of $\mathbb{R}^n$. In the Coquelicot library [5], a type for $\mathbb{R}^n$ is defined thanks to a recursive function iterating cartesian product. It is used to define a type for matrices, but Coquelicot only contains few theorems on these stuctures and the Mathematical Components-like approach is more convenient in practice. Harrison, in his work on $\mathbb{R}^n$ in HOL Light [15, 16], also comes to the conclusion that functions from a finite type of cardinality n to $\mathbb{R}$ are a good way to represent points in $\mathbb{R}^n$ but the limitations of HOL Light force him to use "encoding tricks". Isabelle/HOL also inherits this formalization from HOL Light.

Concerning differentials, we decided to use type classes to add automation in spite of the existence in Coquelicot of a reflexive tactic computing derivatives [4, 23]. The main issue with this tactic is that it works only on functions from $\mathbb{R}$ to itself, while we have at some point to compute the differential of a function whose domain is $\mathbb{R}^n$ (e.g. we need to compute the differential of E in order to compute the derivative of $E \circ \mathrm{sol}_p$). We can also mention the formal proof of the automatic differentiation algorithm O∂yssée by Mayero [30], which is also limited to functions from $\mathbb{R}$ to itself.

7 Conclusion and Future Work

In this paper we presented our formal proof of soundness of the control function for the inverted pendulum from Lozano et al. [25]. Thanks to this proof, we corrected several mistakes from the work by Lozano et al. and we provide a concrete use case to our previous work on LaSalle's invariance principle [8]. We improved this previous work by making it possible to deal with more differential equations thanks to a slight modification of our definition of the notion of solution.

We made functional analysis in $\mathbb{R}^n$ more accessible in Coq [39] through the combination of the Coquelicot [5] and Mathematical Components[7] libraries. This combination

[7] https://math-comp.github.io/math-comp/

however suffers from the difficulties we have with subtypes in Coq, when we want to use it for dynamical systems which have a physical interpretation (e.g. see our discussion on non negative real numbers in Sect. 5.1).

We also extended Coquelicot with a structure of topological space, which is less constraining than the notion of uniform space. This structure makes easier the proof of Tychonoff's Theorem. Moreover we added some automation for the computation of derivatives and differentials.

This work still has to be completed with a proof of the Cauchy-Lipschitz Theorem in order to prove the existence and uniqueness of solutions to the differential equation, which we admit for now. It also has several possible continuations. We could port the proof of stability to an implementation of the pendulum. This would require us to study numerical approximation schemes. Moreover, this work opens the door to the formalization of other systems which are proven using the same techniques, for instance systems which are modelled as inverted pendulums [11, 36, 38].

Acknowledgments

We thank the anonymous reviewers for their useful feedback. We are also grateful to our Ph.D. advisors Yves Bertot and Cyril Cohen for their comments on this paper and on the formalization. This work was partially funded by the ANR project *FastRelax* (ANR-14-CE25-0018-01) of the French National Agency for Research.

References

[1] Abhishek Anand and Ross A. Knepper. 2015. ROSCoq: Robots Powered by Constructive Reals. In *Interactive Theorem Proving - 6th International Conference, ITP 2015, Nanjing, China, August 24-27, 2015, Proceedings (Lecture Notes in Computer Science)*, Christian Urban and Xingyuan Zhang (Eds.), Vol. 9236. Springer, 34–50. https://doi.org/10.1007/978-3-319-22102-1_3

[2] Sandrine Blazy, Christine Paulin-Mohring, and David Pichardie (Eds.). 2013. *Interactive Theorem Proving - 4th International Conference, ITP 2013, Rennes, France, July 22-26, 2013. Proceedings.* Lecture Notes in Computer Science, Vol. 7998. Springer. https://doi.org/10.1007/978-3-642-39634-2

[3] Sylvie Boldo, François Clément, Jean-Christophe Filliâtre, Micaela Mayero, Guillaume Melquiond, and Pierre Weis. 2013. Wave Equation Numerical Resolution: A Comprehensive Mechanized Proof of a C Program. *J. Autom. Reasoning* 50, 4 (2013), 423–456. https://doi.org/10.1007/s10817-012-9255-4

[4] Sylvie Boldo, Catherine Lelay, and Guillaume Melquiond. 2012. Improving Real Analysis in Coq: A User-Friendly Approach to Integrals and Derivatives. In *Certified Programs and Proofs - Second International Conference, CPP 2012, Kyoto, Japan, December 13-15, 2012. Proceedings (Lecture Notes in Computer Science)*, Chris Hawblitzel and Dale Miller (Eds.), Vol. 7679. Springer, 289–304. https://doi.org/10.1007/978-3-642-35308-6_22

[5] Sylvie Boldo, Catherine Lelay, and Guillaume Melquiond. 2015. Coquelicot: A User-Friendly Library of Real Analysis for Coq. *Mathematics in Computer Science* 9, 1 (2015), 41–62. https://doi.org/10.1007/s11786-014-0181-1

[6] Matthew Chan, Daniel Ricketts, Sorin Lerner, and Gregory Malecha. 2016. Formal Verification of Stability Properties of Cyber-physical Systems.

[7] Debasish Chatterjee, Amit Patra, and Harish K. Joglekar. 2002. Swing-up and stabilization of a cart-pendulum system under restricted cart track length. *Systems & Control Letters* 47, 4 (2002), 355–364. https://doi.org/10.1016/S0167-6911(02)00229-3

[8] Cyril Cohen and Damien Rouhling. 2017. A Formal Proof in Coq of LaSalle's Invariance Principle. In *Interactive Theorem Proving - 8th International Conference, ITP 2017, Brasília, Brazil, September 26-29, 2017, Proceedings (Lecture Notes in Computer Science)*, Mauricio Ayala-Rincón and César A. Muñoz (Eds.), Vol. 10499. Springer, 148–163. https://doi.org/10.1007/978-3-319-66107-0_10

[9] Luís Cruz-Filipe, Herman Geuvers, and Freek Wiedijk. 2004. C-CoRN, the Constructive Coq Repository at Nijmegen. In *Mathematical Knowledge Management, Third International Conference, MKM 2004, Bialowieza, Poland, September 19-21, 2004, Proceedings (Lecture Notes in Computer Science)*, Andrea Asperti, Grzegorz Bancerek, and Andrzej Trybulec (Eds.), Vol. 3119. Springer, 88–103. https://doi.org/10.1007/978-3-540-27818-4_7

[10] David Delahaye and Micaela Mayero. 2001. Field, une procédure de décision pour les nombres réels en Coq. In *Journées francophones des langages applicatifs (JFLA'01), Pontarlier, France, Janvier, 2001 (Collection Didactique)*, Pierre Castéran (Ed.). INRIA, 33–48.

[11] Edward Doskocz, Yuri Shtessel, and Constantine Katsinis. 1998. MIMO Sliding Mode Control of a Robotic "Pick and Place" System Modeled as an Inverted Pendulum on a Moving Cart. In *Proceedings of Thirtieth Southeastern Symposium on System Theory.* 379–383. https://doi.org/10.1109/SSST.1998.660100

[12] Nathan Fulton, Stefan Mitsch, Jan-David Quesel, Marcus Völp, and André Platzer. 2015. KeYmaera X: An Axiomatic Tactical Theorem Prover for Hybrid Systems. In *Automated Deduction - CADE-25 - 25th International Conference on Automated Deduction, Berlin, Germany, August 1-7, 2015, Proceedings (Lecture Notes in Computer Science)*, Amy P. Felty and Aart Middeldorp (Eds.), Vol. 9195. Springer, 527–538. https://doi.org/10.1007/978-3-319-21401-6_36

[13] Georges Gonthier, Assia Mahboubi, and Enrico Tassi. 2015. *A Small Scale Reflection Extension for the Coq system.* Research Report RR-6455. Inria Saclay Ile de France. https://hal.inria.fr/inria-00258384

[14] Benjamin Grégoire and Assia Mahboubi. 2005. Proving Equalities in a Commutative Ring Done Right in Coq, See [19], 98–113. https://doi.org/10.1007/11541868_7

[15] John Harrison. 2005. A HOL Theory of Euclidean Space, See [19], 114–129. https://doi.org/10.1007/11541868_8

[16] John Harrison. 2013. The HOL Light Theory of Euclidean Space. *J. Autom. Reasoning* 50, 2 (2013), 173–190. https://doi.org/10.1007/s10817-012-9250-9

[17] Heber Herencia-Zapana, Romain Jobredeaux, Sam Owre, Pierre-Loïc Garoche, Eric Feron, Gilberto Pérez, and Pablo Ascariz. 2012. PVS Linear Algebra Libraries for Verification of Control Software Algorithms in C/ACSL. In *NASA Formal Methods - 4th International Symposium, NFM 2012, Norfolk, VA, USA, April 3-5, 2012. Proceedings (Lecture Notes in Computer Science)*, Alwyn Goodloe and Suzette Person (Eds.), Vol. 7226. Springer, 147–161. https://doi.org/10.1007/978-3-642-28891-3_15

[18] Johannes Hölzl, Fabian Immler, and Brian Huffman. 2013. Type Classes and Filters for Mathematical Analysis in Isabelle/HOL, See [2], 279–294. https://doi.org/10.1007/978-3-642-39634-2_21

[19] Joe Hurd and Thomas F. Melham (Eds.). 2005. *Theorem Proving in Higher Order Logics, 18th International Conference, TPHOLs 2005, Oxford, UK, August 22-25, 2005, Proceedings.* Lecture Notes in Computer Science, Vol. 3603. Springer.

[20] Fabian Immler and Johannes Hölzl. 2012. Numerical Analysis of Ordinary Differential Equations in Isabelle/HOL. In *Interactive Theorem Proving - Third International Conference, ITP 2012, Princeton, NJ, USA, August 13-15, 2012. Proceedings (Lecture Notes in Computer Science)*,

Lennart Beringer and Amy P. Felty (Eds.), Vol. 7406. Springer, 377–392. https://doi.org/10.1007/978-3-642-32347-8_26

[21] Hassan K. Khalil. 2002. *Nonlinear Systems*. Prentice Hall. https://books.google.fr/books?id=t_d1QgAACAAJ

[22] Joseph LaSalle. 1960. Some Extensions of Liapunov's Second Method. *IRE Transactions on Circuit Theory* 7, 4 (Dec 1960), 520–527. https://doi.org/10.1109/TCT.1960.1086720

[23] Catherine Lelay and Guillaume Melquiond. 2012. Différentiabilité et intégrabilité en Coq. Application à la formule de d'Alembert. In *23èmes Journées Francophones des Langages Applicatifs*. Carnac, France, 119–133. http://hal.inria.fr/hal-00642206/fr/

[24] Alexandre Liapounoff. 1907. Problème général de la stabilité du mouvement. *Annales de la Faculté des sciences de Toulouse : Mathématiques* 9 (1907), 203–474. http://eudml.org/doc/72801

[25] Rogelio Lozano, Isabelle Fantoni, and Dan Block. 2000. Stabilization of the inverted pendulum around its homoclinic orbit. *Systems & Control Letters* 40, 3 (2000), 197–204.

[26] Assia Mahboubi and Enrico Tassi. 2013. Canonical Structures for the Working Coq User, See [2], 19–34. https://doi.org/10.1007/978-3-642-39634-2_5

[27] Evgeny Makarov and Bas Spitters. 2013. The Picard Algorithm for Ordinary Differential Equations in Coq, See [2], 463–468. https://doi.org/10.1007/978-3-642-39634-2_34

[28] Gregory Malecha, Daniel Ricketts, Mario M. Alvarez, and Sorin Lerner. 2016. Towards Foundational Verification of Cyber-Physical Systems. In *2016 Science of Security for Cyber-Physical Systems Workshop, SOSCYPS@CPSWeek 2016, Vienna, Austria, April 11, 2016*. IEEE Computer Society, 1–5. https://doi.org/10.1109/SOSCYPS.2016.7580000

[29] Micaela Mayero. 2001. *Formalisation et automatisation de preuves en analyses réelle et numérique*. Ph.D. Dissertation. Université Paris VI.

[30] Micaela Mayero. 2002. Using Theorem Proving for Numerical Analysis (Correctness Proof of an Automatic Differentiation Algorithm). In *Theorem Proving in Higher Order Logics, 15th International Conference, TPHOLs 2002, Hampton, VA, USA, August 20-23, 2002, Proceedings (Lecture Notes in Computer Science)*, Victor Carreño, César A. Muñoz, and Sofiène Tahar (Eds.), Vol. 2410. Springer, 246–262. https://doi.org/10.1007/3-540-45685-6_17

[31] Sayan Mitra and K. Mani Chandy. 2008. A Formalized Theory for Verifying Stability and Convergence of Automata in PVS, See [32], 230–245. https://doi.org/10.1007/978-3-540-71067-7_20

[32] Otmane Aït Mohamed, César A. Muñoz, and Sofiène Tahar (Eds.). 2008. *Theorem Proving in Higher Order Logics, 21st International Conference, TPHOLs 2008, Montreal, Canada, August 18-21, 2008. Proceedings*. Lecture Notes in Computer Science, Vol. 5170. Springer.

[33] Ioana Paşca. 2008. A Formal Verification for Kantorovitch's Theorem. In *Journées Francophones des Langages Applicatifs*. http://hal.inria.fr/inria-00202808/en/

[34] Ioana Paşca. 2011. Formal proofs for theoretical properties of Newton's method. *Mathematical Structures in Computer Science* 21, 4 (2011), 683–714. https://doi.org/10.1017/S0960129511000077

[35] André Platzer and Jan-David Quesel. 2008. KeYmaera: A Hybrid Theorem Prover for Hybrid Systems (System Description). In *Automated Reasoning, 4th International Joint Conference, IJCAR 2008, Sydney, Australia, August 12-15, 2008, Proceedings (Lecture Notes in Computer Science)*, Alessandro Armando, Peter Baumgartner, and Gilles Dowek (Eds.), Vol. 5195. Springer, 171–178. https://doi.org/10.1007/978-3-540-71070-7_15

[36] Naoji Shiroma, Osamu Matsumoto, Shuuji Kajita, and Kazuo Tani. 1996. Cooperative Behavior of a Wheeled Inverted Pendulum for Object Transportation. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS 1996, November 4-8, 1996, Osaka, Japan*. IEEE, 396–401. https://doi.org/10.1109/IROS.1996.570801

[37] Matthieu Sozeau and Nicolas Oury. 2008. First-Class Type Classes, See [32], 278–293. https://doi.org/10.1007/978-3-540-71067-7_23

[38] Tomomichi Sugihara, Yoshihiko Nakamura, and Hirochika Inoue. 2002. Realtime Humanoid Motion Generation through ZMP Manipulation Based on Inverted Pendulum Control. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation, ICRA 2002, May 11-15, 2002, Washington, DC, USA*. IEEE, 1404–1409. https://doi.org/10.1109/ROBOT.2002.1014740

[39] The Coq Development Team. 2016. *The Coq proof assistant reference manual*. http://coq.inria.fr Version 8.6.

[40] Albert Wilansky. 2008. *Topology for Analysis*. Dover Publ., New York, NY. http://cds.cern.ch/record/2222525