



**HAL**  
open science

# Hybrid modelling and limit cycle analysis for a class of five-phase ABS algorithms

William Pasillas-Lépine

► **To cite this version:**

William Pasillas-Lépine. Hybrid modelling and limit cycle analysis for a class of five-phase ABS algorithms. *Vehicle System Dynamics*, 2006, 44 (2), pp.173-188. 10.1080/00423110500385873. hal-00525744

**HAL Id: hal-00525744**

**<https://hal.science/hal-00525744>**

Submitted on 10 Jun 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## Hybrid Modelling and Limit Cycle Analysis for a Class of Five-phase Anti-lock Brake Algorithms

William Pasillas-Lépine, Laboratoire des signaux et systèmes, CNRS - UMR 8506,  
École supérieure d'électricité, 3 rue Joliot-Curie, Plateau de Moulon, 91192  
Gif-sur-Yvette Cedex, FRANCE, E-mail: [pasillas@lss.supelec.fr](mailto:pasillas@lss.supelec.fr)  
(December 15, 2004)

The aim of our paper is to provide a new class of five-phase anti-lock brake algorithms (that use wheel deceleration logic-based switching) and a simple mathematical background that explains their behavior. Firstly, we completely characterize the conditions required for our algorithm to work. Secondly, we explain how to compute analytically an approximation of the Poincaré map of the system (without using numerical integration) and show how to calibrate de algorithm's parameters to obtain the most efficient limit cycle.

### Introduction

In the literature, one can distinguish two completely different kinds of anti-lock brake system designs: those based on logic switching from wheel deceleration information and those based on wheel slip regulation.

Approches based on wheel slip regulation (see, e.g., the articles included in Johansson and Rantzer 2003), have several nice features: they are often based on a clear mathematical background, the torque applied to the wheel converges to a fixed value (there are no periodic oscillations), and they work even if there is no well-defined maximum point in the friction coefficient curve. Their usage is nevertheless confronted to two difficulties. Firstly, it is not always very clear how one can estimate wheel slip precisely (or equivalently the speed of the vehicle). Secondly, the value of wheel slip  $\lambda^*$  for which tyre force is maximal is in general unknown (and not easy to estimate in real-time).

Approches based on wheel deceleration thresholds (see, e.g., Kiencke and Nielsen 2000, Bosch GmbH 2003), have quite interesting properties too: they are very robust with respect to friction coefficient changes and are able to keep wheel slip in a neighborhood of the optimal point, without using explicitly the value of  $\lambda^*$ . But a particularly unpleasant characteristic of these approaches is that they are often based on heuristic arguments, and thus tuning the thresholds involved in this kind of algorithms might be a difficult task.

In this paper, we modify the standard approach based on wheel deceleration thresholds in such a way that the behavior of the resulting algorithm can be understood and predicted using a simple mathematical model. In particular, we will be interested in the conditions that wheel deceleration thresholds must satisfy if we want the algorithm to work properly. We will also be interested in the existence and stability of limit cycles, which is determined by the properties of the Poincaré map. Our results were tested only using computer simulations; no validation has been done on a real vehicle up to now.

It is important to stress that our aim was not to invent a new class of anti-lock brake systems that would be more efficient than those that already exist. Our aim was to investigate, through a simplified algorithm, the basic phase-plane behavior of ABS.

This research was partially supported by the French company PSA Peugeot Citroën.

## 1 Vehicle dynamics

The aim of this section is to derive a pair of equations, namely (2) and (3), that will be used in the rest of the paper to describe the dynamical evolution of the wheels during an ABS regulation.

### 1.1 Tyre forces

The longitudinal tyre force  $F_x$  is often modelled by a relation

$$F_x(\lambda, F_z) = \mu(\lambda)F_z.$$

That is, by a function that depends linearly on vertical force  $F_z$  and nonlinearly on wheel slip

$$\lambda = \frac{R\omega - v_x}{v_x},$$

where  $\omega$  denotes the angular velocity of the wheel and  $v_x$  the speed of the vehicle. It should be noted that this kind of models can only be used at high speeds.

The function  $\mu(\cdot)$  will be described (see Figure 1), for negative wheel slips, using a second order rational fraction

$$\mu(\lambda) = \frac{a_1\lambda - a_2\lambda^2}{1 - a_3\lambda + a_4\lambda^2}.$$

The coefficients  $a_i$  are all positive and depend on tyre characteristics, road conditions, tyre pressure, temperature, etc. They should thus be assumed to be unknown.

### 1.2 Wheel velocity

The angular velocity  $\omega$  of a given wheel of the vehicle has the following dynamics:

$$I\dot{\omega} = -RF_x + T, \quad (1)$$

where  $I$  denotes the inertia of the wheel,  $R$  its radius,  $F_x$  the longitudinal tyre force, and  $T$  the torque applied to the wheel.

This torque  $T = T_e - T_b$  is composed of the engine torque  $T_e$  and of the brake torque  $T_b$ . We will assume that during an ABS braking the clutch pedal is kept engaged, and thus neglect the engine torque. The brake torque is usually supposed to be proportional to brake pressure  $T_b = \gamma P_b$ .

### 1.3 Chassis dynamics

In order to understand fully several phenomena associated to ABS regulation, one should consider at least a two-wheel model with suspension dynamics. Nevertheless, the basic dynamic behavior of wheel angular acceleration, which will be central to our study, can be analyzed with a simpler single-wheel model.

In this simplified model, the vehicle is supposed to brake with the maximal constant deceleration  $a_x^*$  allowed by road conditions, which is  $a_x^* = -\mu(\lambda_0)g$ .

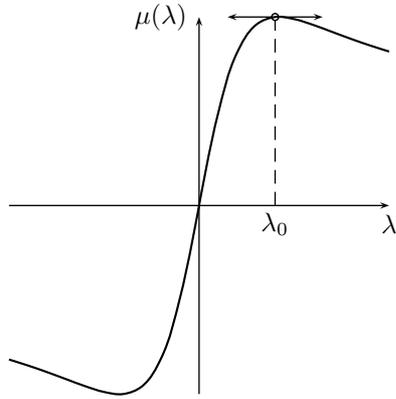


Figure 1. Tyre forces. The shape of the curve  $\mu(\cdot)$ .

In other words

$$\dot{v}_x = a_x^*.$$

The vertical forces  $F_z$  on the front and rear axles are assumed constant and equal to those one would have at equilibrium for this constant deceleration  $a_x^*$ . The front and rear wheel dynamics are then completely decoupled.

#### 1.4 Wheel acceleration

Let  $\lambda^* = -\lambda_0$  be the optimal negative wheel slip, such that that  $\mu'(\lambda^*) = 0$ . If we define the *wheel slip offset*  $x_1$  and the *wheel acceleration offset*  $x_2$  by

$$\begin{aligned} x_1 &= \lambda - \lambda^* \\ x_2 &= R\dot{\omega} - a_x^*, \end{aligned}$$

then we obtain the following control system:

$$\dot{x}_1 = \frac{1}{v_x} (x_2 - (\lambda^* + x_1)a_x^*) \quad (2)$$

$$\dot{x}_2 = -\frac{a}{v_x} \bar{\mu}'(x_1) (x_2 - (\lambda^* + x_1)a_x^*) + u, \quad (3)$$

where

$$a = \frac{R^2}{I} F_z \quad \text{and} \quad u = \frac{R}{I} \dot{T}.$$

The function  $\bar{\mu}(\cdot)$  is defined as

$$\bar{\mu}(x) = \mu(\lambda^* + x) - \mu(\lambda^*).$$

Observe that, for a function  $\mu(\cdot)$  that is a second order rational fraction (see Appendix A), we have

$$\bar{\mu}(x) = \frac{x^2}{\bar{a}_1 - \bar{a}_2 x + \bar{a}_3 x^2}.$$

We refer the reader to Ono et al. (2003) for a different wheel deceleration model.

In this paper, our goal will be to keep the unmeasured variable  $x_1$  in a small neighborhood of zero, with a control  $u$  that only uses the measured variable  $y = R\dot{\omega} - a_x^*$ ; the function  $\mu(\cdot)$  being unknown.

## 2 Hybrid modelling

The aim of this section is to define our ABS regulation logic. In order to do this, we need to study the basic properties of the wheel dynamics, when a constant or a quickly varying torque is applied to the wheel. The basic tools in this study are the first integrals introduced in Subsection 2.1, which are then used in Subsection 2.2 to construct our ABS regulation logic. The conditions that should be satisfied by the algorithm's thresholds, if we want this strategy to work properly, are explored in Subsections 2.3 and 2.4.

### 2.1 First integrals

The following lemma describes the evolution of the system when a constant brake torque is applied to the wheel (see Olson et al. 2003, for a different approach). On a real vehicle, this situation appears on a given wheel when its associated electromagnetic control valve is closed.

LEMMA 2.1 (Constant torque) *When  $u = 0$ , the function*

$$I_1 = x_2 + a\bar{\mu}(x_1)$$

*is a first integral of the dynamical system defined by equations (2) and (3).*

*The equilibrium points of the system are located on the set*

$$E = \{x_2 = (\lambda^* + x_1)a_x^*\}.$$

*If we restrict the system to the domain  $x_1 < \lambda_0$ , these equilibria are locally stable when  $x_1 > 0$  and unstable when  $x_1 < 0$ .*

*Outside these equilibria, the system evolves along the integral curves of  $I_1$ ; in the direction of increasing  $x_1$ 's above the equilibrium set  $E$  and in the opposite direction below this set.*

*Proof* The fact that  $I_1$  is a first integral, when  $u = 0$ , is trivial:

$$\frac{dI_1}{dt} = \frac{dx_2}{dt} + a\bar{\mu}'(x_1)\frac{dx_1}{dt} = 0.$$

It is also easy to see that a point  $(x_1^*, x_2^*)$  is an equilibrium point if and only if it belongs to the set  $E$ .

The direction of the system's evolution along the integral curves of  $I_1$  is determined by the sign of  $\dot{x}_1$ , given by equation (2), and thus the system evolves in the direction of increasing  $x_1$ 's above the equilibrium set  $E$  and in the opposite direction below this set. The stability (or instability) of the equilibrium points is a direct consequence of this fact.  $\square$

For large torque variations (when the associated electromagnetic control valve is open), an approximative description of the system's evolution can still be obtained. We are going to consider the particular case of a torque derivative that is inversely proportional to wheel speed (namely  $\dot{T}_b = u_0/R\omega$ , for some constant  $u_0$ ). This assumption might seem strange at a first glance, but is explained by the fact that the wheel dynamics becomes time invariant when such a control is applied, which is a necessary condition if we want to converge to a limit cycle. From an automotive point of view, this assumption is also natural because it induces smoother brake pressure variations at high vehicle speeds, at which the driver is very sensitive to these perturbations.

LEMMA 2.2 (Large torque variations) *Consider a constant  $u_0$ , big enough in absolute value. For a control of the form  $u = u_0/R\omega$ , the function*

$$I_2 = \ln(1 + \lambda^* + x_1) - \frac{1}{2u_0} (x_2 - (\lambda^* + x_1)a_x^*)^2$$

*is approximatively a first integral of the system; in the sense that, if we put  $\epsilon = 1/|u_0|$ , the system evolves inside a tube of radius  $O(\epsilon^2)$  along the integral curves of  $I_2$ .*

*For a given  $M > 0$ , consider the restriction of the system to the domain*

$$D = \{|x_2| \leq M\}.$$

*Then, for  $\epsilon$  small enough, the system has no equilibrium point inside  $D$ . It evolves approximatively along the integral curves of  $I_2$ , in the direction of increasing  $x_2$ 's when  $u_0 > 0$  and in the opposite direction when  $u_0 < 0$ .*

*Proof* Though the original system is not autonomous, because the vehicle's speed depends on time, if we consider a new time-scale  $s$ , defined by the relation  $ds = dt/\epsilon v_x(t)$ , we obtain an autonomous system

$$\begin{aligned} \frac{dx_1}{ds} &= \epsilon (x_2 - (\lambda^* + x_1)a_x^*) \\ \frac{dx_2}{ds} &= -a\epsilon\bar{\mu}'(x_1) (x_2 - (\lambda^* + x_1)a_x^*) + \frac{\text{sgn}(u_0)}{1 + \lambda_* + x_1}, \end{aligned}$$

for which the dependence on time has disappeared.

Since we have

$$\begin{aligned} \frac{dI_2}{ds} &= \frac{1}{1 + \lambda_* + x_1} \frac{dx_1}{ds} - \epsilon \operatorname{sgn}(u_0) (x_2 - (\lambda^* + x_1)a_x^*) \left( \frac{dx_2}{ds} - a_x^* \frac{dx_1}{ds} \right) \\ &= \epsilon^2 (x_2 - (\lambda^* + x_1)a_x^*)^2 (a_{\bar{\mu}}'(x_1) + a_x^*), \end{aligned}$$

the system evolves inside a tube of radius  $O(\epsilon^2)$  along the curve

$$I_2(x_1, x_2) = I_2(x_1(0), x_2(0)).$$

This curve gives thus an approximation of the phase-plane evolution of the system when  $\epsilon$  is small.

Observe that, for any fixed  $M$ , for small enough values of  $\epsilon$  the sign of  $dx_2/ds$  on the specified domain will be the same as the sign of  $u_0$ , thus the system has no equilibrium point in  $D$ , and evolves in the direction of increasing  $x_2$ 's when  $u_0 > 0$  and in the opposite direction when  $u_0 < 0$ .  $\square$

## 2.2 A hybrid regulation logic

Our control system (2) and (3) can be simplified even further by using the following guidelines:

- The torque applied to the system will be either kept constant or changed very quickly. We will thus be able to use Lemmas 2.1 and 2.2 in order to have an approximation of the phase-plane evolution of the system.
- The important point will not be the timewise evolution of the system, but the unparametrized curve followed in the phase plane together with the direction of movement.
- The switches between constant torque, torque decrease, and torque increase will be triggered by thresholds on the value of the wheel acceleration offset  $x_2$ . The regulation logics will be chosen in such a way that that unmeasured variable  $x_1$  remains in a small neighborhood of zero.

A particular case of regulation logic that follows these guidelines is proposed in Figure 2. From now on, we will restrict our attention to this class of five-phase ABS algorithms.

Roughly speaking, a hybrid automaton is a dynamical system that has both discrete and continuous states. On the one hand, the evolution of the discrete states is determined by an automaton (like, for example, the one of Figure 2), which state transitions are governed by guard conditions (the labels on the automaton's graph edges) that are inequalities on the continuous states of the system. These inequalities determine whether it is possible to have a discrete

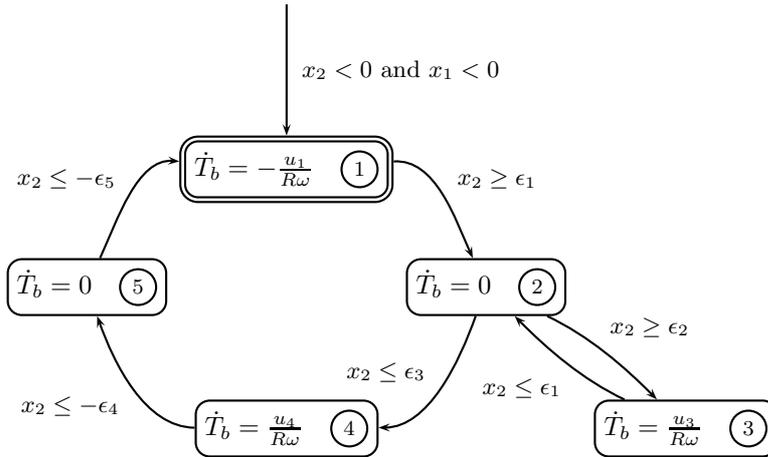


Figure 2. Our five-phase ABS regulation logic.

state transition. On the other hand, the evolution of the continuous states is determined by a dynamical system (that may depend on the discrete state of the automaton). For a background on hybrid automata, we refer the reader to Lygeros et al. (2003) and to Van der Schaft and Schumacher (2000).

In the case of anti-lock brake systems, the discrete state of the automaton indicates the current phase of the regulation algorithm (Figure 2), while the continuous state of the system describes the dynamical evolution of the wheel, given by the differential equations (2) and (3). The transitions between the different phases of the algorithm are given by the guard conditions of the automaton (Figure 2). In other words, if we take the regulation logics of Figure 2 and we plug it into the control system defined by equations (2) and (3), then we will obtain a hybrid automaton.

More precisely, the hybrid automaton obtained by taking

- the vertices of the graph of Figure 2 as the set of discrete states ;
- the phase plane  $(x_1, x_2)$  as the set of continuous states ;
- the differential equations (2) and (3) with the control  $u$  defined by the label of a given vertex as the vector field associated to the corresponding discrete state ;
- the label of each edge of the graph as the guard condition corresponding to this edge ;
- the intersection of the complements of the associated guards, for all the edges leaving a given vertex, as the domain or invariant set associated to the corresponding discrete state ;
- and the third quadrant of the phase plane as the set of initial conditions

will be called the *five-phase ABS hybrid automaton*.

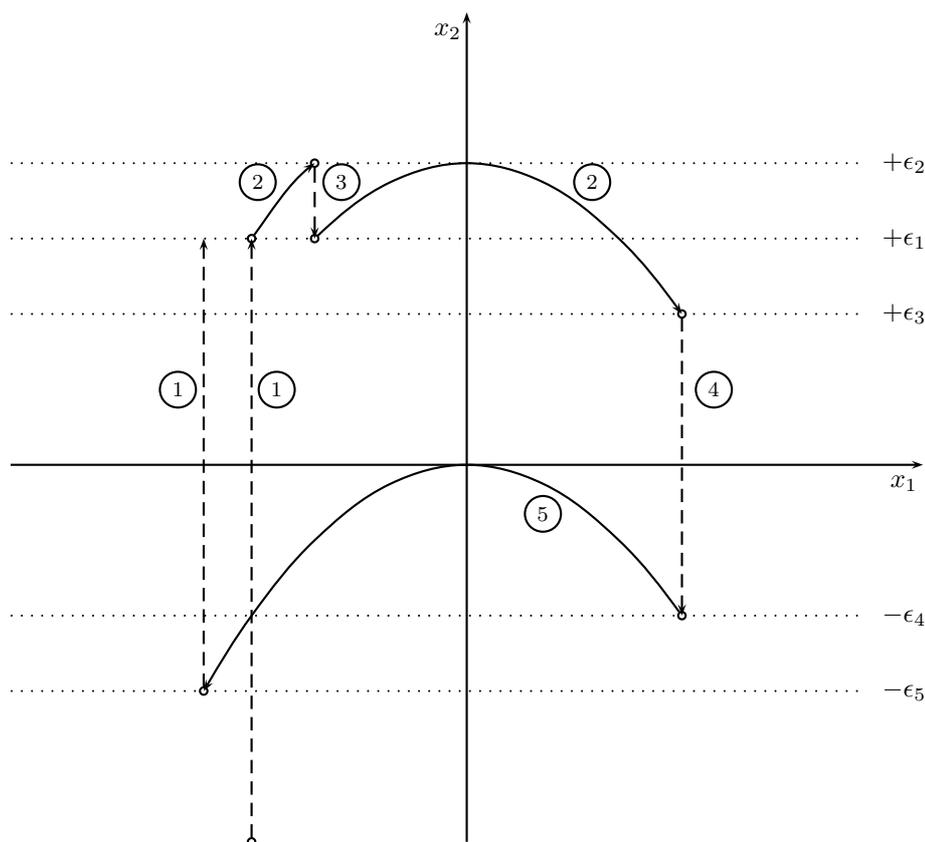


Figure 3. Phase-plane evolution of the system during an ABS regulation.

At this point, the reader should not be afraid of the previous (quite mathematical) definition. Indeed, we are now going to explain in detail what our strategy does during each phase of the algorithm, which might clarify our regulation logic. The associated phase-plane evolution is illustrated in Figure 3.

**Phase 0** We assume that the initial condition of the system is such that  $x_1 < 0$  and  $x_2 < 0$ . In other words, the wheel has entered the unstable domain: if the brake pressure is maintained constant then the wheel will lock.

**Phase 1** We rapidly decrease brake torque by taking  $\dot{T}_b = -u_1/R\omega$ . By Lemma 2.2, if  $u_1$  is large enough, we will move approximately along an integral curve of  $I_2$  until  $x_2 \geq \epsilon_1$ . From a physical point of view, the aim of this phase is to change the sign of the wheel's angular acceleration. The consequence of this brake pressure decrease is that

the wheel slip will first stop increasing and then will start decreasing (in absolute value).

**Phase 2** As soon as  $x_2 \geq \epsilon_1$ , we maintain a constant brake torque. By Lemma 2.1, since  $x_2 > 0$  we will move along an integral curve of  $I_1$  in the direction of increasing  $x_1$ 's (we thus return towards the stable zone of the tyre). The variable  $x_2$  will first increase while  $x_1 < 0$ ; and then decrease once  $x_1 > 0$  (provided that Phase 3 is not triggered before). All along Phase 2 we have  $x_2 < \epsilon_2$  and  $x_2 > \epsilon_3$ . Indeed, Phase 3 is triggered when  $x_2 \geq \epsilon_2$ ; while Phase 4 is triggered when  $x_2 \leq \epsilon_3$ . The practical aim of this phase is to detect whether we are in the stable or in the unstable region of the wheel dynamics.

**Phase 3** If we have  $x_2 \geq \epsilon_2$ , we rapidly increase brake torque by taking  $\dot{T}_b = u_3/R\omega$ . By Lemma 2.2, if  $u_3$  is large enough, we will move approximately along an integral curve of  $I_2$  until  $x_2 \leq \epsilon_1$ , which will trigger the second phase of the algorithm. Phases 2 and 3 will then alternate until  $x_2 \leq \epsilon_3$ , which will trigger the fourth phase. From a physical point of view, the aim of this phase is to prevent the wheel from returning too far into the stable region (which would increase the braking distance).

**Phase 4** As soon as  $x_2 \leq \epsilon_3$ , we rapidly increase brake torque by taking  $\dot{T}_b = u_4/R\omega$ . By Lemma 2.2, if  $u_4$  is large enough, we will move approximately along an integral curve of  $I_2$  until  $x_2 \leq -\epsilon_4$ , which will trigger the fifth phase of the algorithm. Once this phase has been triggered, we know that we are in the stable region of the wheel dynamics. We can thus increase the brake pressure in order to change the sign of wheel deceleration and increase wheel slip again.

**Phase 5** As soon as  $x_2 \leq -\epsilon_4$ , we maintain a constant brake torque. By Lemma 2.1, since  $x_2 < 0$  we will move along an integral curve of  $I_1$  in the direction of decreasing  $x_1$ 's (we thus return towards the unstable zone of the tyre). The variable  $x_2$  will first increase while  $x_1 > 0$ ; and then decrease once  $x_1 < 0$  (provided that we do not intersect the equilibrium set, which will be granted by the threshold conditions of Subsection 2.3). As soon as  $x_2 \leq -\epsilon_5$  the first phase of the algorithm is triggered, and everything starts again (in order to be sure that the set  $x_2 = -\epsilon_5$  is intersected, we must impose the threshold conditions of Subsection 2.3). From a physical point of view, the aim of this phase is to detect that the wheel has reached the unstable region, and thus that it is necessary to start again the algorithm from the beginning.

### 2.3 When does it work?

Of course, the choice of thresholds  $\epsilon_i$  cannot be made arbitrarily.

Firstly, we must obviously have

$$\epsilon_3 < \epsilon_1 < \epsilon_2 \quad \text{and} \quad \epsilon_4 < \epsilon_5. \quad (4)$$

Secondly, if we impose

$$\epsilon_3 > |a_x^*|, \quad (5)$$

we can avoid the integral curves of  $I_1$  to intersect the equilibrium set during Phase 2.

Thirdly, we must have

$$\epsilon_4 > \epsilon_2 - \epsilon_3, \quad (6)$$

in order to avoid the integral curves of  $I_1$  to intersect the equilibrium set during Phase 5, which would block the algorithm in the stable zone of the tyre.

Fourthly, we must have

$$a(\mu(\lambda_0) - \mu(+\infty)) > \epsilon_5 - \epsilon_4 + \epsilon_2 - \epsilon_3, \quad (7)$$

in order to allow the integral curves of  $I_1$  to intersect the set  $\{x_2 = -\epsilon_5\}$  during Phase 5, and thus trigger again Phase 1. Otherwise wheel slip would continue to increase until wheel lock-up.

### 2.4 What does it do?

Define  $T_0 = v_x(0)/|a_x^*|$ . It is clear that we can only study the system on the time interval  $[0, T_0]$ , because for our model  $v_x(T_0) = 0$  and afterwards wheel slip is thus undefined. The following result states that our hybrid automaton is well posed on this time interval (we refer the interested reader to Zhang et al. (2001) for a definition of the Zeno phenomenon). Roughly speaking, we just make sure that the behavior of the automaton will not be pathological. Readers that are not familiar with the theory of hybrid automata might just skip this result and jump to Proposition 2.4.

**PROPOSITION 2.3** *Assume that the conditions of Subsection 2.3 are satisfied and that the gains  $u_i$  are big enough. Then, our five-phase ABS hybrid automaton is deterministic and non-blocking. For each initial condition such that  $x_1 < 0$  and  $x_2 < 0$ , it admits a unique infinite execution. This execution is Zeno, with Zeno time  $T_0$ .*

*Proof* Even though our hybrid automaton is not autonomous, a time-scaling  $ds = dt/v_x(t)$  eliminates the dependence on time (like in the proof of Lemma 2.2) and thus allows us to use the well-posedness conditions of Lygeros et al. (2003) to prove the stated results. Observe, moreover, that since all the guards associated to the edges leaving a given vertex  $p$  of the graph are closed, and since the domain  $\text{Dom}(p)$  of the corresponding discrete state is defined as the intersection of their complements, all the domains of the hybrid automaton are, by construction, open sets. It thus follows that (Lygeros et al. 1999, Proposition 1), for each discrete state  $p$ , the set  $\text{Out}(p)$  (of points for which continuous evolutions is impossible) is the complement of its domain  $\text{Dom}(p)$ . The set  $\text{Out}(p)$  is thus the union of all guards associated to the edges leaving the associated vertex of the graph.

The fact that the automaton is non-blocking is a trivial consequence of (Lygeros et al. 2003, Lemma III.1). Indeed, since for our automaton  $\text{Out}(p)$  is the union of all guards associated to the edges leaving the associated vertex of the graph, for each point  $x \in \text{Out}(p)$  there exists a vertex  $(p, q)$  of the graph such that  $x \in \text{Guard}(p, q)$ ; which corresponds precisely to the condition of the cited Lemma.

The fact that the automaton is deterministic is a trivial consequence of (Lygeros et al. 2003, Lemma III.2). Indeed, the first condition of the cited Lemma is satisfied again because for our automaton  $\text{Out}(p)$  is the union of all guards associated to the edges leaving the associated vertex of the graph; the second condition because in our case the intersection of all guards associated to the edges leaving a given vertex of the graph is empty; and the third condition because we do not have resets.

Now, in order to analyse the Zeno phenomenon on the executions of our automaton, it is convenient to consider the explicit form of the time-scaling

$$s(t) = \frac{T_0}{v_x(0)} \ln \left( \frac{1}{1 - \frac{t}{T_0}} \right).$$

In the new time scale  $s$ , it is easy to check that there exist two strictly positive lower and upper bounds for the time spent by the system in any phase of the algorithm. Indeed, when the conditions of Subsection 2.3 are satisfied and when the gains  $u_i$  are big enough, the vector fields associated to each discrete state have no equilibrium point in the adherence of the corresponding domain; and the continuous state of the system must thus leave this domain in a finite amount of time (Lemmas 2.1 and 2.2). It follows that, in the new time scale  $s$ , the unique execution associated to a given initial condition has infinitely many switchings and is not Zeno. Nevertheless, in the original time

scale  $t$ , the corresponding time trajectory has an accumulation point at  $t = T_0$ , because the transformation between the two time scales maps a finite interval into an infinite one.  $\square$

The following result gives bounds on wheel slip.

**PROPOSITION 2.4** *Define  $\lambda_{\min}$  and  $\lambda_{\max}$ , respectively, as the only negative and positive solutions of the two following equations*

$$\begin{aligned} a\bar{\mu}(\lambda_{\min}) &= \epsilon_5 - \epsilon_4 + \epsilon_2 - \epsilon_3 \\ a\bar{\mu}(\lambda_{\max}) &= \epsilon_2 - \epsilon_3. \end{aligned}$$

*Assume that the conditions of Subsection 2.3 are satisfied and that the gains  $u_i > 1/\epsilon$  are big enough. Then, there exists two positive constants  $c_{\min}$  and  $c_{\max}$  such that, once the algorithm has reached Phase 4, the wheel slip offset remains in a neighborhood*

$$I = [\lambda_{\min} - c_{\min}\epsilon, \lambda_{\max} + c_{\max}\epsilon]$$

*of zero. Moreover  $c_{\min} \simeq \epsilon_5^2/2$  and  $c_{\max} \simeq \epsilon_3^2/2$ .*

*Proof* The idea is the same as in the proof of Proposition 3.1, we use  $I_1$  and  $I_2$  to approximate the phase-plane evolution of the system.  $\square$

At this point, one must be careful. The previous relations show that the operating interval around zero could be made arbitrarily small, by taking small enough values for the thresholds  $\epsilon_i$ . But this cannot be done on a real vehicle, because for too small values of the thresholds  $\epsilon_i$  our algorithm would be very sensitive to measurement noise on the output  $y = R\dot{\omega} - a_x^*$ .

### 3 Limit cycle analysis

In some sense, limit cycles are a generalization of the notion of equilibrium point. Instead of having a point toward which all trajectories starting from neighboring points converge, we have a closed curve that attracts all neighboring trajectories (see Figure 6).

The basic tool for studying limit cycles is Poincaré's first return map. In the case of smooth dynamical systems, the Poincaré map is obtained by taking a transverse section to a closed curve, and by analyzing the evolution of the system's state each time this transverse section is intersected. This can be done by linearizing the system along the closed curve (see, e.g., Guckenheimer and Holmes 1997). In the case of hybrid systems, a standard approach is to consider a guard condition of the automaton as the equivalent of the transverse section.

This leads to analyse the evolution of the continuous state each time a fixed discrete state is triggered. In both cases, the main interest of the Poincaré map is to replace the study of an  $n$ -dimensional continuous dynamical system by the analysis of a discrete dynamical system of dimension  $n - 1$ , which usually appears to be an easier problem to solve.

In the case of ABS, we have a two-dimensional hybrid automaton. The Poincaré map of the system is thus a map defined on a real interval. The aim of this section will be to compute explicitly an approximation of this map ; and to tune the parameters of the ABS algorithm in such a way that the Poincaré map will have a stable equilibrium point (which corresponds to a stable limit cycle for the original system). Different limit cycles can be obtained using this approach, but we point out a choice of the parameters for which the cycle efficiency is optimal (in the sense that it minimizes the braking distance).

### 3.1 First return map

In order to describe the behavior of our algorithm, we will consider the value  $x_1(t_n)$  of the variable  $x_1$  at each instant  $t_n$  for which the fourth phase of the algorithm is triggered (for the  $n$ -th time). This value will be denoted  $P_n$ . The Poincaré first return map  $f(\cdot)$  gives, by definition, the value of  $P_{n+1}$  as a function of  $P_n$ . In other words  $P_{n+1} = f(P_n)$ .

Instead of considering the variable  $P_n$ , it is sometimes easier to compute the Poincaré map using the variable  $Q_n$ , that we define as

$$Q_n = \frac{a\bar{\mu}(P_n) + \epsilon_3 - \epsilon_1}{\epsilon_2 - \epsilon_1}.$$

When  $u_3$  is big enough, we have  $Q_n \in [0, 1)$ .

### 3.2 The general case

The original problem (computing the phase-plane evolution during a whole ABS cycle) can be decomposed in a sequence of subproblems (for a single ABS phase), which are obviously easier to solve.

We will thus consider an arbitrary phase of the algorithm, for which the state  $(x_1, x_2)$  of the system evolves from a point  $(p_i, q_i)$  to a point  $(p_{i+1}, q_{i+1})$ . In any case, the values of  $q_i$ ,  $q_{i+1}$ , and  $p_i$  are known. What should be computed is the value of  $p_{i+1}$ .

**PROPOSITION 3.1** *For a phase that preserves the first integral  $I_2$ , that is when brake torque is increased or decreased using a control  $u = u_0/R\omega$ , we have*

$$p_{i+1} = p_i + \frac{1 + \lambda^* + p_i}{2u_0} (q_{i+1}^2 - q_i^2 - 2(\lambda^* + p_i)a_x^*(q_{i+1} - q_i)) + O(\epsilon^2).$$

where  $\epsilon = 1/|u_0|$ .

For a phase that preserves the first integral  $I_1$ , that is when brake torque is maintained constant, define

$$y = \bar{\mu}(p_i) + \frac{1}{a}(q_i - q_{i+1}).$$

When  $p_{i+1}$  is positive

$$p_{i+1} = \frac{\sqrt{\bar{a}_2^2 y^2 + 4\bar{a}_1 y(1 - \bar{a}_3 y)} - \bar{a}_2 y}{2(1 - \bar{a}_3 y)}$$

and when  $p_{i+1}$  is negative

$$p_{i+1} = \frac{-\sqrt{\bar{a}_2^2 y^2 + 4\bar{a}_1 y(1 - \bar{a}_3 y)} - \bar{a}_2 y}{2(1 - \bar{a}_3 y)}.$$

*Proof* For a phase that preserves a given first integral  $I_k$ , with  $k = 1$  or  $2$ , we must have

$$I_k(p_{i+1}, q_{i+1}) = I_k(p_i, q_i).$$

The formulas stated in the proposition are obtained by solving this equation, with unknown variable  $p_{i+1}$ ; exactly for the first integral  $I_1$  and up to terms of order  $O(\epsilon^2)$  for the approximative first integral  $I_2$ .  $\square$

The main interest of the previous result is to allow an *analytic* approximation of the Poincaré map; without using any numerical integration of the nonlinear system of differential equations (2) and (3). It is then possible to use an optimization routine in order to tune the parameters of the algorithm, since the computation time needed to calculate the cost function is divided by several orders of magnitude.

This result allows, moreover, a simple study of certain particular cases (like the bang-bang and symmetric cases, considered below) that explains the role played by the different parameters in the behavior of the algorithm and, more precisely, the existence and stability of fixed points for the Poincaré map (and thus the existence of limit cycles for the original system).

### 3.3 The bang-bang limit

As a particular case, one can consider the limit behavior of our algorithm when the parameters  $u_i$  are taken infinitely big. Then during the first, third,

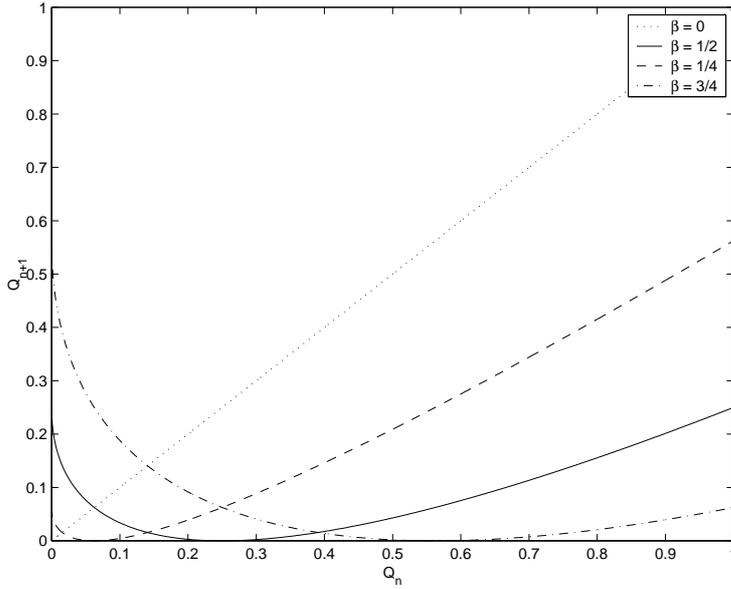


Figure 4. The  $L^\infty$ -norm of the Poincaré map is minimized when  $\beta = 1/2$ .

and fourth phases, the system jumps instantaneously at some instant  $t_0$  from a given value  $x_2(t_0^-)$  of wheel deceleration offset to a target value  $x_2(t_0^+)$ . This behavior can be obtained by applying to the wheel a well chosen discontinuous torque. Of course, it is physically impossible to increase or decrease torque instantaneously; but the bang-bang model appears to be helpful for understanding the dynamics of the continuous model. Indeed, since in this case we have  $p_{i+1} = p_i$  for the Phases 1, 3, and 4, a simple iteration of Proposition 3.1 along a complete ABS cycle gives

$$Q_{n+1} = Q_n + \frac{\epsilon_5 - \epsilon_4 + \epsilon_1 - \epsilon_3}{\epsilon_2 - \epsilon_1} \text{ mod } 1.$$

The Poincaré map is thus a translation of the circle: either all points are periodic of order  $k$ , or no periodic point exists at all, and the map is then ergodic.

In particular, if

$$\epsilon_5 - \epsilon_4 + \epsilon_1 - \epsilon_3 = k(\epsilon_2 - \epsilon_1), \quad \text{with } k \in \mathbb{N},$$

then the Poincaré map is just the identity. That is  $Q_{n+1} = Q_n$ . In other words, independently of the initial condition, the system's state will return to the same point of the phase plane each time the fourth phase of the algorithm is triggered.

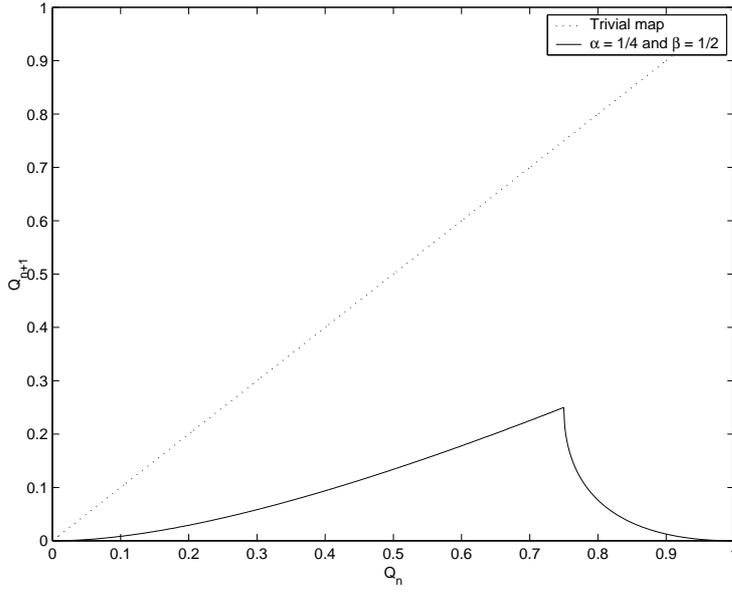


Figure 5. When  $\beta = 1/2$ , by taking  $\alpha = 1/4$  the best limit cycle becomes globally attractive.

### 3.4 The symmetric case

Another particular case appears if we assume that  $\epsilon_4 = \epsilon_3$  and  $\epsilon_5 = \epsilon_1$ . Indeed, in this case, like in the bang-bang case, the Phases 1 and 4 of the algorithm will be such that  $p_{i+1} = p_i$ , which simplifies considerably the explicit computation of the Poincaré map.

With the help of the following approximations

$$\bar{\mu}(x) \simeq \frac{x^2}{a_1} \quad \text{and} \quad I_2 \simeq x_1 + \frac{1}{2u_3}x_2^2,$$

it is then easy to show, again by an iteration of Proposition 3.1 along a complete ABS cycle, that the Poincaré map is approximately given by the following formula:

$$Q_{n+1} \simeq \left( \sqrt{(Q_n + \alpha) \bmod 1} - \beta \right)^2,$$

with

$$\alpha = \frac{\epsilon_5 - \epsilon_4 + \epsilon_1 - \epsilon_3}{\epsilon_2 - \epsilon_1} \quad \text{and} \quad \beta = \frac{1}{2u_3} \frac{\epsilon_2^2 - \epsilon_1^2}{\sqrt{\epsilon_2 - \epsilon_1}} \sqrt{a\mu''(\lambda_0)}.$$

The previous formula can be used to tune the parameters of the algorithm

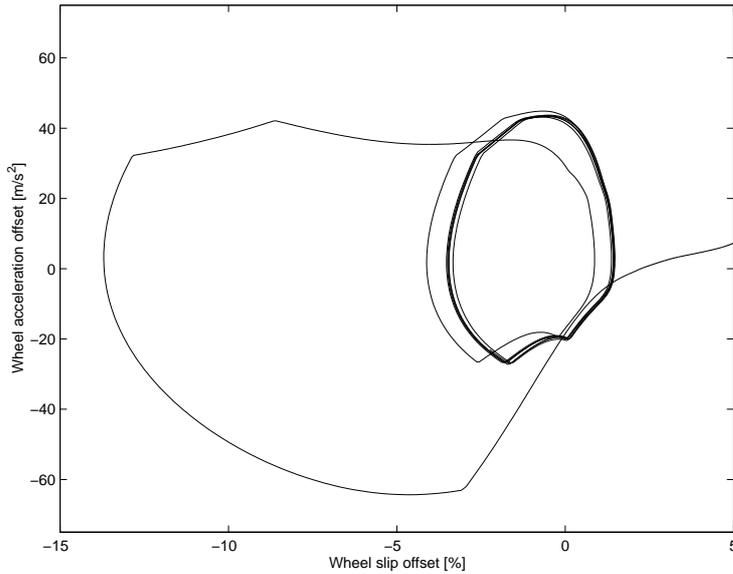


Figure 6. Simulation with PSA's advanced vehicle model SIMULINKCAR. Phase plane evolution.

in the symmetric case. One should first observe that the  $L^\infty$ -norm of the Poincaré map does not depend on the parameter  $\alpha$ , and that we can minimize this norm of the map by taking  $\beta = 1/2$ . With this choice (see Figure 4), we will minimize the value of  $Q_{n+1}$  for the worst initial condition  $Q_n$ . Then, if  $\beta = 1/2$ , by taking  $\alpha = 1/4$  the best limit cycle (which corresponds to a fixed point at  $Q_n = 0$  for the Poincaré map) becomes globally attractive (see Figure 5).

Some simulations with an advanced vehicle model are shown in Figures 6–8, at the end of the paper. In these simulations, we took  $\epsilon_1 = 27.5$  and  $\epsilon_2 = 39.5$  and  $\epsilon_3 = 20.0$ , which corresponds to  $\alpha = 1/4$ ; we moreover chose a value of  $u_3$  such that  $\beta = 1/2$ .

#### 4 Conclusion

We proposed and studied a new class of five-phase ABS algorithms. Our work is placed in a context where the vehicle brakes on a straight line, the speed is high, and there is a well defined maximum for the  $\mu(\cdot)$ -curve (which, though assumed to be unknown, is supposed to remain unchanged during the whole braking).

The case of a  $\mu(\cdot)$ -curve that does not have a well defined maximum (typically this is the case when braking on ice) is special, because condition (7) of Subsection 2.3 is not satisfied. Nevertheless, one can solve this problem by

increasing slowly the brake torque during the fifth phase of the algorithm, instead of keeping a constant torque. But for this modified approach the computation of the Poincaré map is more complicated and, up to now, we have not been able to obtain its analytic form.

The case of discontinuous transitions of road characteristics cannot be tackled with the five-phase algorithm proposed here. Nevertheless, we have obtained recently, in Ait-Hammouda and Pasillas-Lépine (2004), an eleven-phase strategy that is able to handle this situation.

Even though our interest was focused on a particular class of five-phase anti-lock schemes, our method for computing the Poincaré map can be applied to more general classes of ABS algorithms, based on wheel deceleration thresholds. It can also be applied in order to simplify the analysis of other algorithms, like the four-phase control strategies of Kuo and Yeh (1992).

## REFERENCES

- Ait-Hammouda, I. and Pasillas-Lépine, W. (2004), On a class of eleven-phase anti-lock brake algorithms, robust with respect to discontinuous transitions of road characteristics, *in* ‘Proceedings of the IFAC Symposium on Systems, Structure and Control’, Oaxaca (Mexico).
- Bosch GmbH, R. (2003), *Conventional and Electronic Braking Systems for Passenger Cars*, Bentley.
- Guckenheimer, J. and Holmes, P. (1997), *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer-Verlag.
- Johansson, R. and Rantzer, A., eds (2003), *Nonlinear and Hybrid Systems in Automotive Control*, Springer-Verlag.
- Kiencke, U. and Nielsen, L. (2000), *Automotive Control Systems*, Springer-Verlag.
- Kuo, C. Y. and Yeh, E. C. (1992), ‘A four-phase control scheme of an anti-skid brake system for all road conditions’, *IMechE Part D: Journal of Automobile Engineering* **206**, 275–283.
- Lygeros, J., Johansson, K., Sastry, S. and Egerstedt, M. (1999), On the existence of executions of hybrid automata, *in* ‘Proceedings of the IEEE Conference on Decision and Control’, Phoenix (Arizona), pp. 2249–2254.
- Lygeros, J., Johansson, K., Simić, S., Zhang, J. and Sastry, S. (2003), ‘Dynamical properties of hybrid automata’, *IEEE Transaction on Automatic Control* **48**(1), 2–17.
- Olson, B., Shaw, S. and Stépán, G. (2003), ‘Nonlinear dynamics of vehicle traction’, *Vehicle System Dynamics* **40**(6), 377–399.
- Ono, E., Asano, K., Sugai, M., Ito, S., Yamamoto, M., Sawada, M. and Yasui, Y. (2003), ‘Estimation of automotive tire force characteristics using wheel

- velocity', *Control Engineering Practice* **11**(6), 1361–1370.
- Van der Schaft, A. and Schumacher, H. (2000), *An Introduction to Hybrid Dynamical Systems*, Springer-Verlag.
- Zhang, J., Johansson, K., Lygeros, J. and Sastry, S. (2001), 'Zeno hybrid systems', *International Journal of Robust and Nonlinear Control* (11), 435–451.

### Appendix A: Fitting rational tyre parameters

Even though the proposed ABS algorithm does not use the tyre characteristics, its behavior will obviously be influenced by them. Usually, tyres are characterized using Pacejka's magic formula, or other tyre models. To compute from these models the parameters  $a_i$  of the rational tyre model, we need the braking stiffness at zero  $\mu'(0)$ , that is the derivative with respect to wheel slip of the function  $\mu(\cdot)$ , evaluated at zero; the maximal value of the adhesion coefficient  $\mu(\lambda_0)$ ; the wheel slip  $\lambda_0$  for which the wheel adherence is maximal, which obviously satisfies the relation  $\mu'(\lambda_0) = 0$ ; and the value of the adhesion coefficient when the wheels slip completely or are completely blocked  $\mu(+\infty)$ .

Indeed, there exists a unique set of parameters  $a_i$  matching these four conditions

$$\begin{aligned} a_1 &= \mu'(0) & a_3 &= \frac{\mu'(0)\lambda_0 - 2\mu(\lambda_0)}{\mu(\lambda_0)\lambda_0} \\ a_2 &= \frac{\mu(+\infty)\mu(\lambda_0)}{\lambda_0^2(\mu(\lambda_0) - \mu(+\infty))} & a_4 &= \frac{\mu(\lambda_0)}{\lambda_0^2(\mu(\lambda_0) - \mu(+\infty))}. \end{aligned}$$

The coefficients  $\bar{a}_i$  can then be obtained as a function of the coefficients  $a_i$  and of the optimal wheel slip  $\lambda_0$  by taking

$$\begin{aligned} \bar{a}_1 &= \frac{1 + 2a_3\lambda_0 + (2a_4 + a_3^2)\lambda_0^2 + 2a_3a_4\lambda_0^3 + a_4^2\lambda_0^4}{(a_1a_4 - a_2a_3)\lambda_0 - a_2} \\ \bar{a}_2 &= \frac{a_3 + (2a_4 + a_3^2)\lambda_0 + 3a_3a_4\lambda_0^2 + 2a_4^2\lambda_0^3}{(a_1a_4 - a_2a_3)\lambda_0 - a_2} \\ \bar{a}_3 &= \frac{a_4 + a_3a_4\lambda_0 + a_4^2\lambda_0^2}{(a_1a_4 - a_2a_3)\lambda_0 - a_2}. \end{aligned}$$

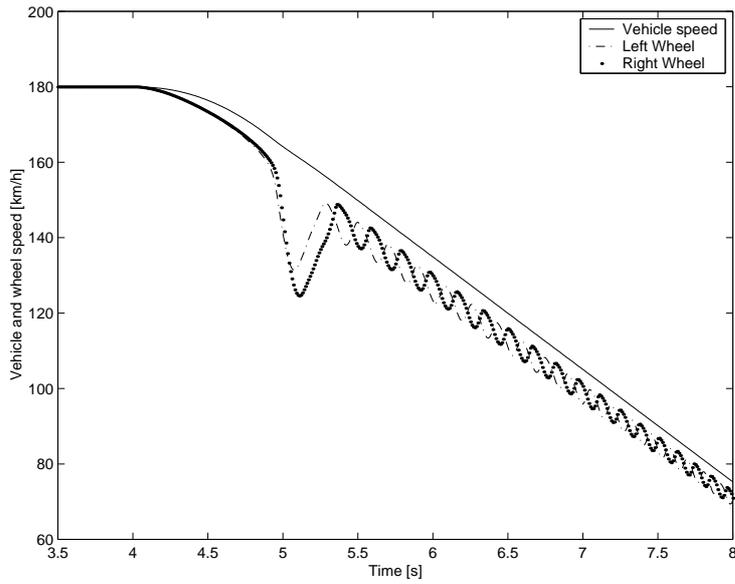


Figure 7. Simulation with PSA's advanced vehicle model SIMULINKCAR. Time evolution for the right and left wheel speeds.

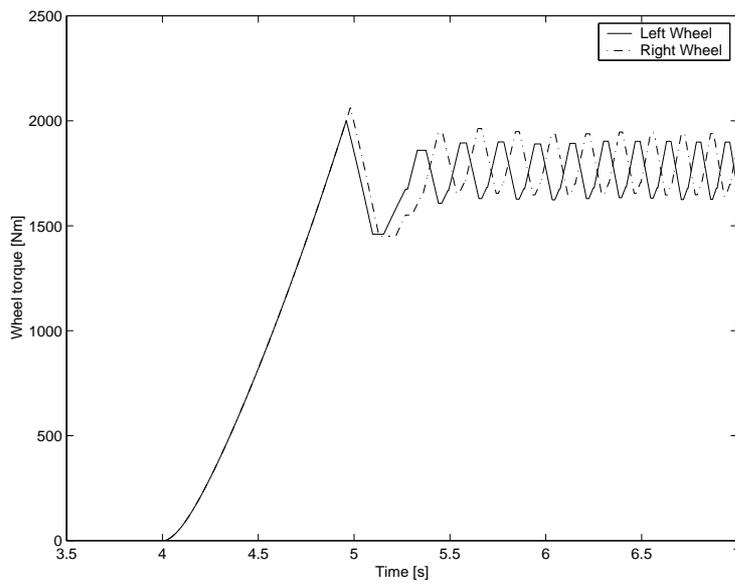


Figure 8. Simulation with PSA's advanced vehicle model SIMULINKCAR. Time evolution for the right and left wheel torques.