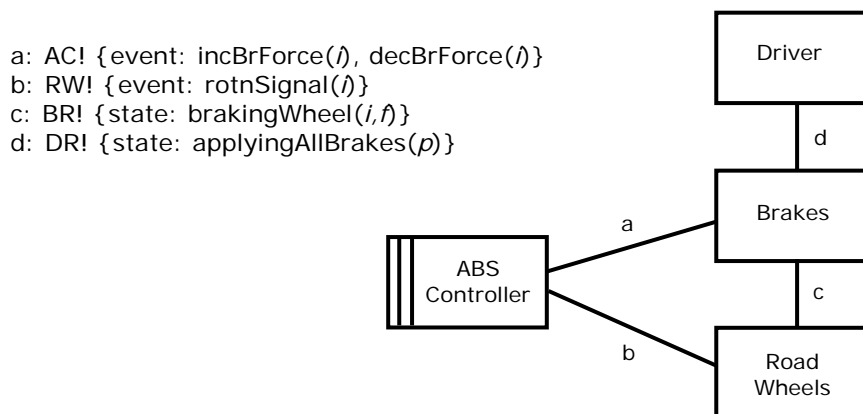


Some Notes on the ABS Problem

Michael Jackson

26 December 2005

1. This note is written in response to Dmitry's document "Anti-locking Brake System (ABS)" and the accompanying extracts from Wikipedia on anti-lock braking systems.
2. From the documents given, the complete context diagram is something like this:



The phenomena are:

- b: rotnSignal(i): this is called sensorsignal _{i} in Dmitry's document.
- d: applyingAllBrakes(p): the driver is pressing with pressure p on the brake pedal.
- c: brakingWheel(i, f): a braking force f is being applied to wheel(i).
- a: incBrForce(i), decBrForce(i): the braking force applied to wheel(i) is increased or decreased by an event at the corresponding valve(i).

It seems from Dmitry's document that the ABS Controller can not detect whether the Driver is pressing on the brake pedal. So we may eventually omit the Driver domain and imagine that the braking pressure is created spontaneously by the Brakes domain. This braking pressure is distinct from the braking force applied to the individual wheels, which is the result of the braking pressure possibly reduced or increased by the ABS Controller's effect on the valves.

3. The functioning of the valves is unclear. Why does Dmitry's document say "C1: .. a signal comes to valve i and the valve increases pressure 1/10 times (10 times per revolution of a wheel)"? What does this mean? Surely the valve behaviour is not dependent on the wheel rotation—that would require a connection between the valve and the wheel rotary position, which seems implausible. If it is not dependent on the wheel rotation, then adjusting the valve setting 10 times per revolution is a responsibility that may be assigned to the ABS Controller in the eventual machine specification, and should not be mentioned in the designation of lockvalve _{i} (e).

I do not understand what the valves do or how they are to be controlled. I can imagine these two possibilities:

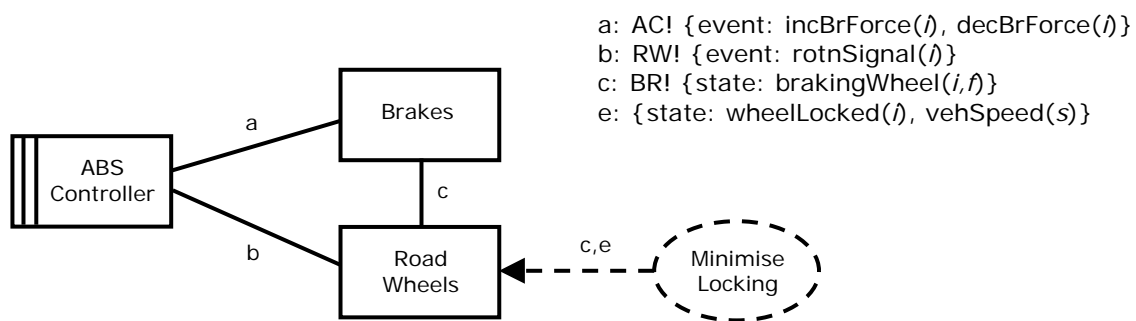
- (a) The valve has two positions: in one it passes the full braking force to the wheel, in the other it cuts off all the braking force. The ABS Controller can choose to hold the valve on for

0/10, 1/10, 2/10, ..., or 10/10 of a wheel revolution and off for the remainder: by holding it on for a higher proportion of each revolution it is effectively passing more of the braking force to the wheel. In this case the Controller is very busily turning the valve on and off as the wheel rotates. This, I suppose, would cause the ‘characteristic pulsing feel’ mentioned in the Wikipedia extracts.

(b) The valve has 10 positions: according to its position it passes more or less of the braking force to the wheel. The valve position is initially set to pass the full braking force. A `decBrForce` event sets it to a lower position, where it passes less of the force, and an `incBrForce` event sets it to a higher position, where it passes more.

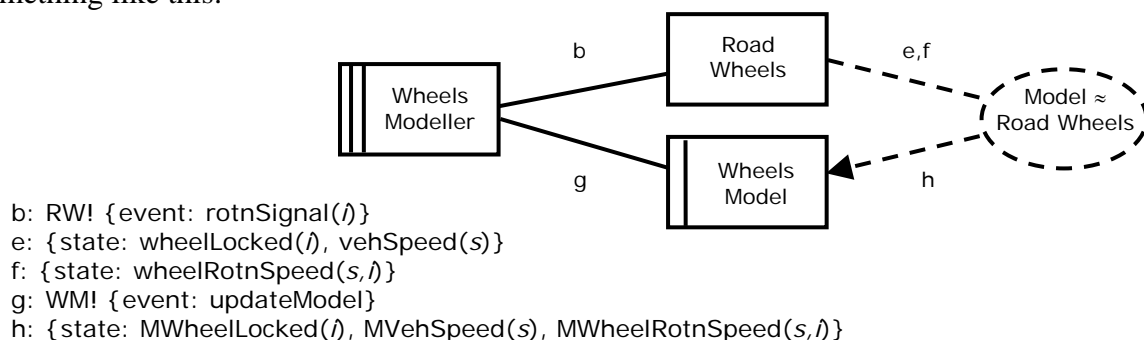
No doubt the truth is a third possibility.

4. What is the complete requirement? It seems to be something like this: “When the vehicle speed is greater than 10km/h, any wheel that is locked should be restored to the unlocked state by reducing the braking force on the wheel; the reduction must be made as fast as is possible and is consistent with smooth operation of the vehicle.” We must then say what we mean by ‘locked’. The diagram IND1 seems to suggest that ‘locked’ is simply ‘not rotating’—that is, the rate of occurrence of `rotnSignal(i)` is 0, and there is no partial locking in which the wheel is rotating but slower than it should be rotating for the vehicle’s current speed. We can represent the requirement in a problem diagram:



The requirement MinimiseLocking mentions only the vehicle speed, the wheel locking, and the braking force on the wheel. It does not mention the requirement that at speeds of less than 10 km/h the ABS Controller must monitor the wheel states: this follows more or less directly from the need to detect a vehicle speed of 10 km/h as soon as it is reached.

5. The introduction of the Wheels Model is certainly right. The questions to be answered are: “What is the vehicle speed?” and, for each wheel individually, “What is the rotation speed?” and “Is the wheel locked?” The subproblem diagram for maintaining the Wheels Model is something like this:



The Brakes domain has no part to play in this subproblem: probably its inclusion in the ABS Controller2 diagram in document-1 was due to a slip of the pen. The requirement is that the

Wheels Model state phenomena h should correspond well enough to the Road Wheels state phenomena e and f.

This subproblem is, I suspect, very subtle. The vehicle speed must be estimated from the individual wheels' rotation speeds; but at the same time locking is detected as a deviation of a wheel's rotation speed from the corresponding vehicle speed. Some domain property of the Road Wheels domain (more properly, from a Vehicle domain which has been omitted) must constrain the possible rate of change of vehicle speed: otherwise a situation in which all four wheels lock simultaneously—for example, because the vehicle is aquaplaning on a wet road—could not be detected. (Perhaps this is why acceleration was included in Dmitry's account of what the ABS Controller needs to know at each instant.)

[end]