Trajectory Deadlock in Power System Models*

Ian A. Hiskens
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109 USA

***Abstract*—A number of standard power system models implement non-windup limits on control blocks in ways that can lead to trajectory deadlock. This deadlock behaviour takes the form of infinitely fast switching, due to incompatibility between the conditions that govern switching on either side of the switching surface. Solutions can only be continued in a Filippov sense. The paper explores this phenomenon in the context of a standard voltage regulator model, and an industry-standard model of a doubly-fed induction generator that is widely used to represent wind turbine generators. Two approaches to preventing deadlock are considered: the first employs a deadband, while the second uses feedback.**

I. Introduction

It is common practice to approximate detailed dynamic processes by simplified models. In doing so, it is important to ensure that the simplified model exhibits appropriate accuracy over the spatial and temporal scales of interest. Such simplifications frequently arise when the behavioural characteristics of an input-output relationship are important, but the exact processes driving those characteristics are of less interest. It is not uncommon for the simplification process to generate models that are not physically realizable. Furthermore, it is becoming more common for simplified models to incorporate switching action, thereby giving rise to representations that are hybrid dynamical systems [1], [2].

Even though models may not describe actual physical processes, they should at least be mathematically realizable. In the case of smooth dynamical systems, mathematical integrity of a model is closely connected with existence and uniqueness of solutions. Conditions governing such properties are well understood [3]. Equivalent properties for hybrid (non-smooth) dynamical systems are not so easily stated [4]. It is perhaps not surprising, therefore, that systems which appear relatively benign can in fact display complicated behaviour. The paper considers two such cases.

The first case is that of a proportional-integral (PI) block with non-windup limits, as shown in Figure 1. Such a block is used in a number of the models documented in IEEE Standard 421.5-2005 [5], with its behaviour defined in Annex E.5 of that standard. Non-windup limits are a standard attribute of integrators. Considering analog devices, the non-windup limits represent physical constraints associated with op-amp voltage rails [6]. The limits do not have the same physical meaning for a PI block though, hence the need for the standard [5] to establish an unambiguous definition. We will show in Section II, however, that the switching inherent in that definition can lead to deadlock [7], with trajectory continuation requiring the notion of Filippov solutions [8].

*Research supported by the National Science Foundation through grants ECCS-0725710 and DMS-1016467.

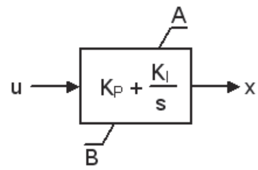

Fig. 1. Non-windup proportional-integral block, from [5].

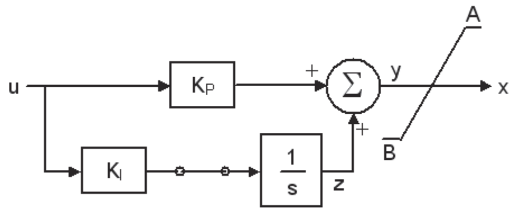

Fig. 2. Definition of the non-windup proportional-integral block, from [5].

The second case stems from modelling type-3 wind turbine generators (WTGs), which are also known as doubly-fed induction generators. A model developed by the Western Electricity Coordinating Council (WECC), and documented in [9], has become a pseudo-standard within industry. This model consists of a number of functional blocks, with the pitch control model shown in Figure 8. The model includes a mechanism for blocking the pitch control and pitch compensation integrators when the blade pitch angle encounters a maximum (PI_{max}) or minimum (PI_{min}) limit. It is shown in Section V that this blocking interaction can also result in deadlock.

II. Non-windup limits on PI block

There is no natural definition for non-windup limits on a PI block. The standard [5] therefore uses the block diagram of Figure 2 to define limit behaviour according to the rules:

$$\text{If } y > A, \quad \text{then } x = A \text{ and } \frac{dz}{dt} = 0$$
$$\text{If } y < B, \quad \text{then } x = B \text{ and } \frac{dz}{dt} = 0.$$

In order to illustrate these rules, a simple case with $K_P = 1$, $K_I = 2$, $\frac{du}{dt} = -1$, and upper limit $A = 1.2$ will be considered. Figure 3 provides the phase portrait of y versus u for this case. For values of $y > 1.2$, $\frac{dy}{dt} = \frac{du}{dt} = -1$. Trajectories in this region are always directed downwards. On the other hand, when $y \leq 1.2$ we have $\frac{dy}{dt} = \frac{du}{dt} + \frac{dz}{dt} = -1 + 2u$. Trajectories are directed upwards for $u > 0.5$ and downwards for $u < 0.5$.

Consider behaviour in a neighbourhood of the limit $y = 1.2$ with $u > 0.5$. If $y < 1.2$, the integrator will not be blocked, so y will increase and encounter the limit, thereby

978-1-4244-9474-3/11/$26.00 ©2011 IEEE

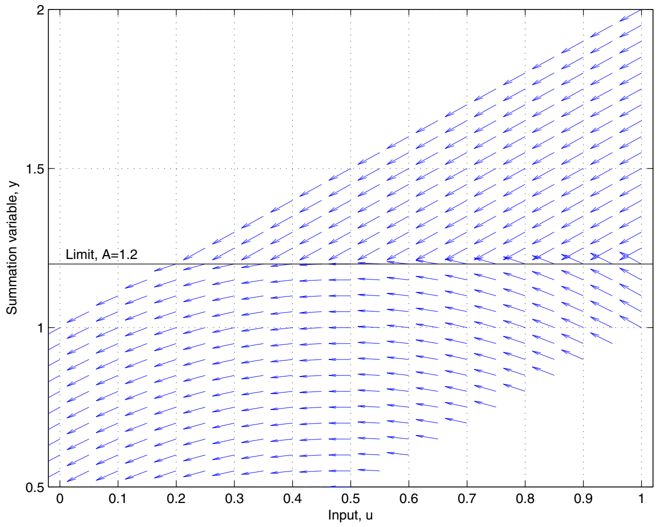

Fig. 3. Phase portrait for non-windup PI block.

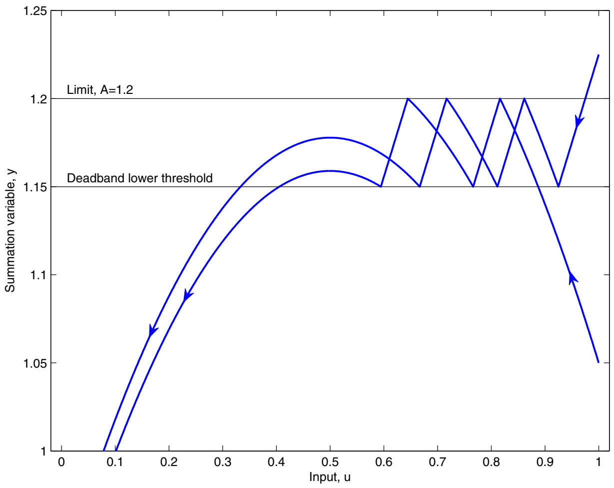

Fig. 4. Response of trajectories to a deadband.

blocking the integrator. On the other hand, if $y > 1.2$ so the integrator is initially blocked, y will decrease and encounter the limit, unblocking the integrator. These two outcomes are inconsistent. Along the limit line, with $u > 0.5$, when the integrator is blocked it should be unblocked, and vice versa. This is apparent in Figure 3, where the vector field points towards the limit line from both sides. As a consequence, trajectories that encounter the limit with $u > 0.5$ cannot proceed. The discrete state describing the status of integrator blocking undergoes infinitely many jumps at the encounter instant. This is referred to as *chattering Zeno* [7].

Notice that when $u < 0.5$, $\frac{dy}{dt} < 0$ above and below the limit line. A trajectory encountering the limit will cross without incident.

III. DEADBAND IMPLEMENTATION

One approach to overcoming the deadlock inherent in the standard model is to introduce a deadband in association with the limit. This will be explained with the aid of Figure 4,

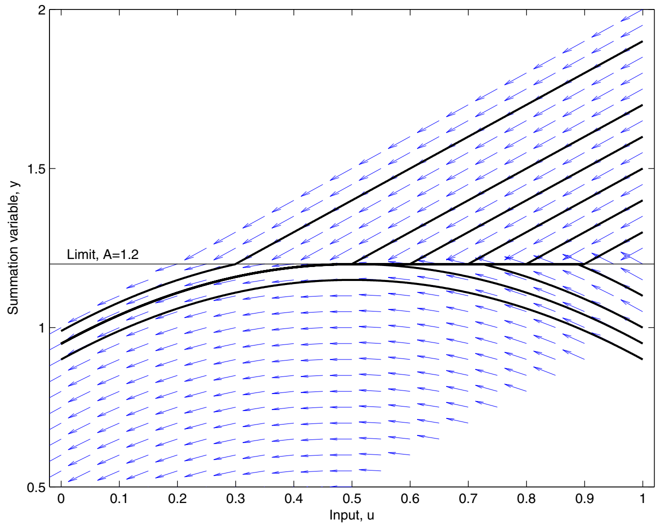

Fig. 5. Phase portrait when the non-windup limit is implemented using a deadband.

where a deadband of width $\epsilon = 0.05$ is shown[1]. Trajectories that emanate from below the deadband, where the integrator is active, rise until they encounter the limit at $y = 1.2$. At that point, the integrator is blocked, and the trajectory falls until the lower threshold of the deadband is encountered at $y = 1.2 - \epsilon$. This triggers the integrator to unblock, and the trajectory proceeds accordingly. The model for this system can be expressed in the differential-algebraic impulsive switched (DAIS) form established in [2],

$$\dot{u} = -1 \tag{1}$$

$$\dot{z} = u(x_{st} + 1) \tag{2}$$

$$\dot{x}_{st} = 0 \tag{3}$$

$$y = u + z \tag{4}$$

$$y_{trig} = y - \big(1.2 + \frac{\epsilon}{2}(x_{st} - 1)\big) \tag{5}$$

$$x_{st}^{+} = -x_{st}^{-} \quad \text{when } y_{trig} = 0 \tag{6}$$

where $x_{st} = 1$ when the integrator is active and $x_{st} = -1$ when it is blocked. Equation (6) is a state reset equation which maps x_{st}^{-}, the value of x_{st} at the instant the trigger condition is satisfied, to a new value x_{st}^{+}.

By implementing the model (1)-(6) with a small deadband, the non-windup PI block may be used in the manner intended by its authors. Figure 5 shows the behaviour in the vicinity of the limit when a deadband width of $\epsilon = 0.001$ is used. Trajectories apparently converge to the section of the limit line where $u > 0.5$, and subsequently run along that line until they reach $u = 0.5$. At that point, the collection of trajectories breaks away from the limit line. For $u \leq 0.5$, the vector fields on either side of the limit line are consistent, so trajectories switch in the normal way. The deadband is not obvious in Figure 5 because it has a very small width. However, it should be reemphasized that trajectories do not run smoothly along the limit line, but rather chatter between the bounds established by the deadband.

[1]The width of this deadband has been deliberately exaggerated to more clearly illustrate behaviour.

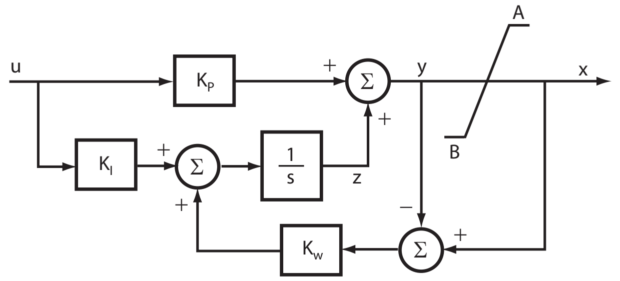

Fig. 6. Non-windup PI block, incorporating "anti-reset windup".

IV. Integrator adjustment using feedback

An alternative approach to addressing the deadlock problem inherent in the non-windup PI block model is to borrow the concept of "anti-reset windup" from control design [10]. Figure 6 shows the basic model of Figure 2 modified to incorporate anti-reset windup. With this approach, the difference $x - y$ induced by the output limiter is fed back through a gain K_w to the integrator input. If y lies within range, x will equal y and the feedback signal will be zero. However if y deviates beyond the limits, the nonzero feedback signal will drive the integrator in the direction that seeks to restore y back to within limits. The model of the non-windup PI block in this case becomes,

$$\dot{u} = -1 \tag{7}$$

$$\dot{z} = 2u + K_w(x - y) \tag{8}$$

$$y = u + z \tag{9}$$

$$x = \begin{cases} y & \text{if } y \leq 1.2 \\ 1.2 & \text{if } y > 1.2. \end{cases} \tag{10}$$

Results obtained by applying this approach are shown in Figure 7. It is instructive to compare this behaviour with that of Figure 5, where the deadband was implemented. For cases where y is initially within limits, the two approaches exhibit quite similar behaviour. The trajectories in Figure 7 deviate a little above the limit $A = 1.2$, with the extent of that deviation dependent upon the feedback gain K_w. The value $K_w = 50$ was used in this example, ensuring that the integrator responded quickly to mismatches between x and y.

In cases where y was initially beyond the limit, Figures 5 and 7 display vastly different behaviour. The feedback associated with the anti-reset windup model provides a large negative input to the integrator, quickly driving y towards the limit. It should be noted, though, that while y is above the limit, the output x will simply be fixed on the limit. Therefore, when viewed in terms of the output x, the behaviour of the two different approaches will appear quite similar.

V. Deadlock in wind turbine generator models

The pitch control block WT3P of the WECC model developed for type-3 wind turbines [9] is shown in Figure 8. Of particular interest is the implementation of the non-windup limiter on the pitch angle θ. As stated in the model documentation,

> "The Pitch Control and Pitch Compensation integrators are non-windup integrators as a function of the pitch, i.e., the inputs of these integrators are set to zero when the pitch is in limits (PI_{max} or PI_{min}) and the integrator input tends to force the pitch command further against its limit."

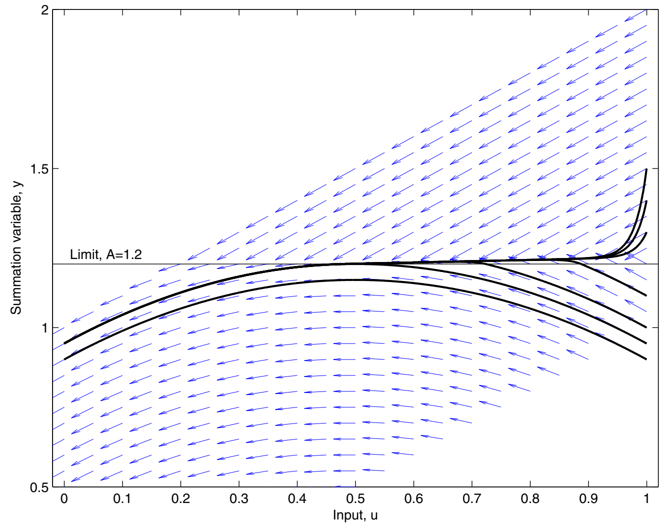

Fig. 7. Phase portrait when anti-reset windup is implemented.

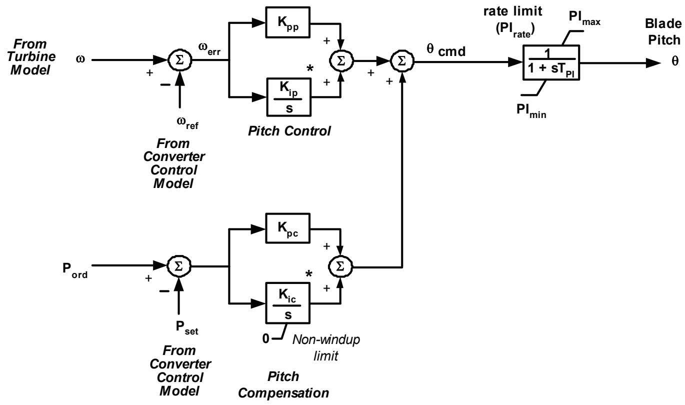

Fig. 8. Pitch control model WT3P, from [9].

To illustrate, consider the case where θ is on its lower limit PI_{min}. A negative input to the pitch-control integrator would cause the corresponding state x_p to reduce, which in turn would force θ further against its PI_{min} limit. To prevent that wind-up effect, the integrator is blocked under such conditions. Similarly, the pitch-compensation integrator is blocked when its input is negative. When θ is on its upper limit PI_{max}, blocking of the up-stream integrators occurs when their respective inputs are positive.

This blocking philosophy is also employed in the converter control model WT3E. It is conceptually similar to the non-windup PI block discussed previously, and can also suffer from switching deadlock. This situation will be illustrated using the following example.

The WECC default parameters and test system, given in [9], form the basis for this example. Resulting trajectories are shown in Figure 9. To further clarify the deadlock behaviour, only the pitch-compensation integrator will be discussed. The

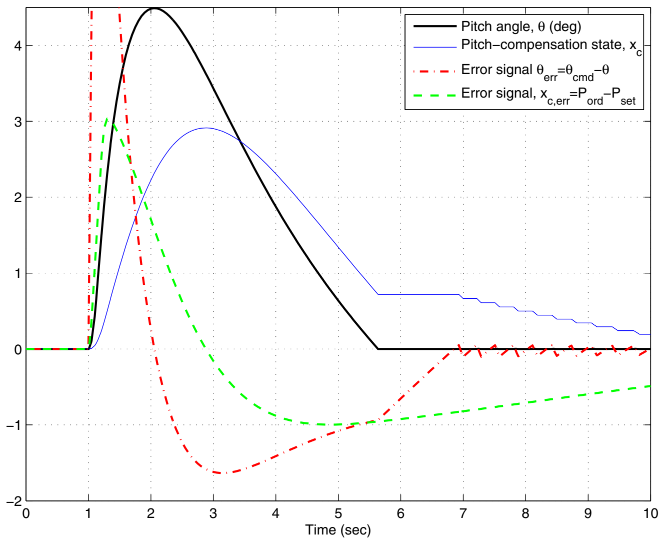

Fig. 9. Large-disturbance response of pitch states.

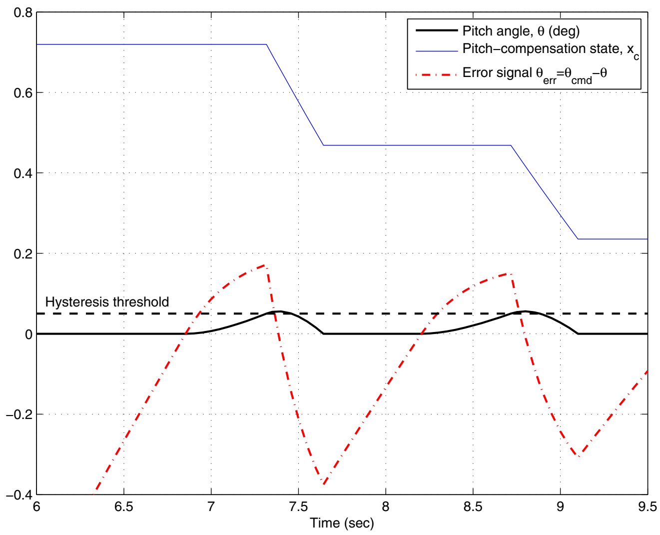

Fig. 10. Expanded view of hysteresis behaviour.

pitch-control integrator exhibits similar behaviour though. It should be noted that in order to generate the trajectories shown in Figure 9, it was necessary to introduce hysteresis into the switching process associated with the pitch angle θ encountering and departing its non-windup limit. The implementation of hysteresis is discussed later.

The pitch angle θ is initially in steady-state on the lower non-windup limit, where $\theta_0 = PI_{min} = 0$ deg. In response to the disturbance, θ undergoes a transient increase, before returning to PI_{min} at 5.6 sec. The error signal $x_{c,err} = P_{ord} - P_{set}$ driving the pitch-compensation integrator is negative at that time, so the corresponding state x_c is frozen. Over the subsequent period, θ and x_c remain frozen, but the signal $\theta_{err} = \theta_{cmd} - \theta$, which drives variations in θ, steadily increases until reaching zero[2] at 7 sec. At that point, the pitch-compensation integrator becomes unblocked. But notice that $x_{c,err}$ is negative, so as soon as x_c is unblocked it decreases, driving θ_{err} negative. This forces θ back onto its PI_{min} limit, and x_c is again blocked. But with x_c blocked, θ_{err} increases above zero, and x_c is unblocked. Without hysteresis, this process would repeat *ad infinitum*. A deadlock is reached, and the trajectory cannot continue.

This deadlock can be overcome by implementing hysteresis in the blocking/unblocking process. The explanation of hysteresis will refer to Figure 10. This is an expanded view of the relevant time interval of Figure 9. In order to provide a clearer view of behaviour, however, the hysteresis band has been widened from 0.002 in Figure 9 to 0.05 in Figure 10.

At 6.85 sec, the error signal θ_{err} crosses through zero. Upon doing so, the integrator driving the pitch angle θ is unblocked, so θ begins to increase. At 7.3 sec, θ encounters the hysteresis threshold, whereupon the pitch-compensation integrator is unblocked. The error signal $x_{c,err}$ driving that integrator is negative, as shown in Figure 9, so x_c immediately begins to reduce. This causes θ_{err} to reduce. Eventually θ_{err} goes negative, and θ begins to fall, encountering the non-windup limit $PI_{min} = 0$ at 7.6 sec. When that limit is encountered, θ and x_c are immediately blocked, so θ_{err} again begins to increase. The process then repeats.

[2]The hysteresis implementation actually allows θ_{err} to rise a little beyond zero before the x_c-integrator is unblocked.

VI. Conclusions

It is not uncommon for models of dynamical systems to involve interactions between continuous processes and discrete events. If such hybrid dynamical systems are not well structured, they may exhibit unusual forms of behaviour. The paper has considered industry-standard models that are used in representing voltage regulators and wind turbine generators, and has shown that both may exhibit infinitely fast switching. This switching phenomenon results in trajectory deadlock. Such non-physical behaviour can be circumvented by carefully restructuring the models.

References

[1] D. Liberzon, *Switching in Systems and Control*. Boston: Birkhauser, 2003.

[2] I. Hiskens, "Power system modeling for inverse problems," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 3, pp. 539–551, March 2004.

[3] H. Khalil, *Nonlinear Systems*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1996.

[4] R. Goebel, R. Sanfelice, and A. Teel, "Hybrid dynamical systems," *Control Systems Magazine*, vol. 29, no. 2, pp. 28–93, April 2009.

[5] IEEE Std 421.5-2005, *IEEE Recommended Practice for Excitation System Models for Power System Stability Studies*. New York: Institute of Electrical and Electronics Engineers, Inc., 2005.

[6] D. Stout and M. Kaufman, *Handbook of Operational Amplifier Circuit Design*. McGraw-Hill, 1976.

[7] A. Abate, A. D'Innocenzo, M. Di Benedetto, and S. Sastry, "Understanding deadlock and livelock behaviors in hybrid control systems," *Nonlinear Analysis: Hybrid Systems*, vol. 3, no. 2, pp. 150–162, May 2009.

[8] A. Filippov, *Differential Equations with Discontinuous Righthand Sides*. The Netherlands: Kluwer Academic Publishers, 1988.

[9] "Generic type-3 wind turbine-generator model for grid studies," Version 1.1, WECC Wind Generator Modeling Group, September 2006.

[10] G. Goodwin, S. Graebe, and M. Salgado, *Control System Design*. Upper Saddle River, New Jersey: Prentice Hall, 2001.