POWER SYSTEMS · AI · RESEARCH

Technical Note 002: Bug Hunting a Power System Stabiliser (PSS) in PSCAD

N-002

This is preliminary research and has not been validated or peer-reviewed. It could therefore be totally wrong.

Key Concepts

- Power System Stabilisers are commonly used in most large rotating synchronous machines. They are essential for helping to provide damping power should a low frequency oscillation occur.

- A badly tuned PSS can cause chaos as it can inject negative damping (via the AVR) into the network and potentially destabilise the machine.

- In this application the study was carried out in PSCAD to analyse an existing GTG that was part of an operational site. The site settings were tested and validated during commissioning. The GTG was equipped with the usual suite of AVR, UEL, OEL, SCL V/Hz and a PSS.

- During the compliance simulations it was noticed that the PSS was working continually and not performing as expected. The PSS settings were 'reasonable' and didn't contain any obvious issues except the PSS gain being a bit higher than would be usually expected.

PSS2B

- The PSS2B is a simple and well-established PSS. It is a dual input (power and speed error) that feeds straight into the AVR.
- For further details on its behaviour and performance look at IEEE 421.5

8.2 Type PSS2B power system stabilizer model

This stabilizer model, shown in Figure 8-2, is designed to represent a variety of dual-input stabilizers, which normally use combinations of power and speed or frequency to derive the stabilizing signal.

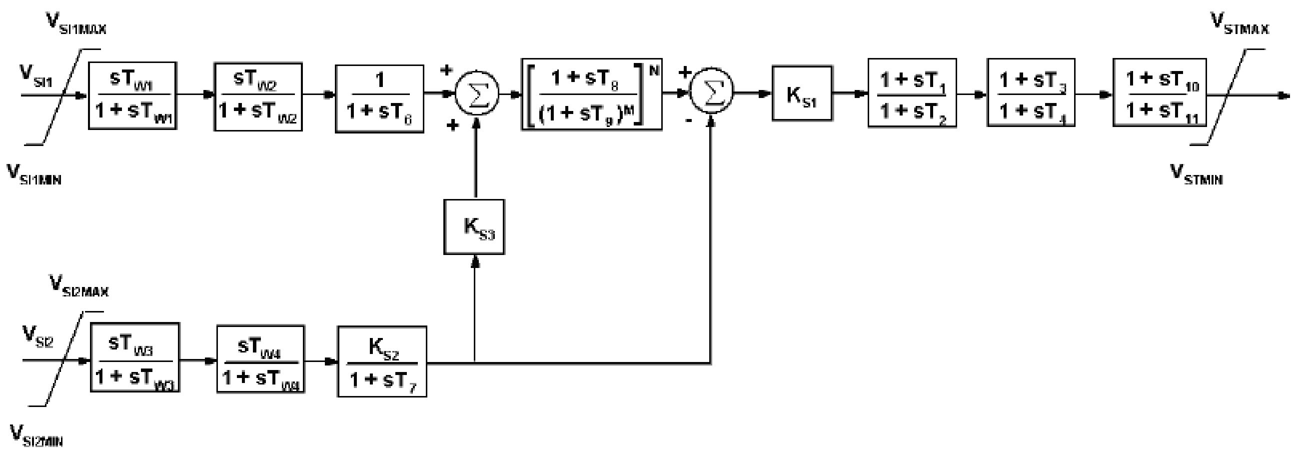


Figure 8-2—Type PSS2B—Dual-input PSS

The Problem

- During the compliance simulations it was noticed that the PSS was operating continuously and working in a bang-bang (i.e. driving from one limit to another limit and back again) control oscillation.

- All the settings were checked, rechecked and checked again between the PSS test data and the PSCAD model.

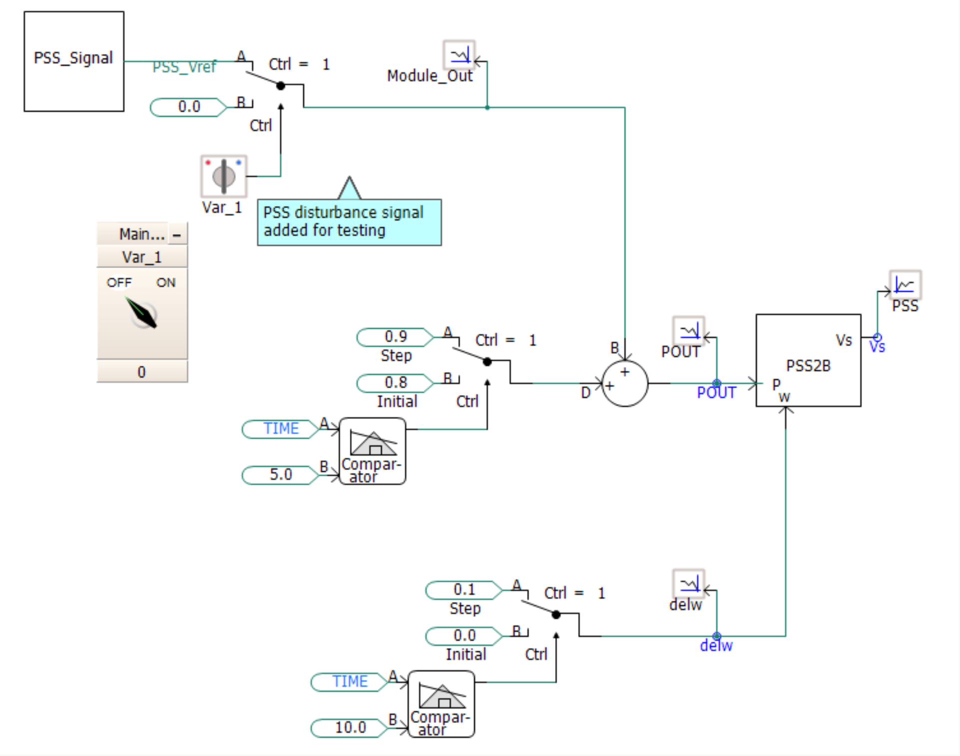
- The input signals were checked to make sure that something daft had not happened like inputting speed instead of delta speed or mixing up the inputs of speed and power.

- The PSCAD PSS2B model is black boxed and cannot be directly interrogated with probes inside the model.

- Investigating these problems manually is a slow and time-consuming process.

Test Rig

- Setup of a simple test rig in PSCAD to investigate the issue. I setup the test rig for the AI, but that was to keep things clean and keep the token burn down.
- The test rig was configured with:
  - Step input to delW (Speed error)
  - Step input to P (Power pu)
  - Applied sinusoid disturbance



The Investigation

- A key point to remember with bug hunting is that 95% of the time it is a data input error, a simple mistake in the wiring an inversion in a signal somewhere or sometimes a typo in the datasheet or notes….. But not always

- A key concept with this type of bug hunting, is isolate the problem part and test that on its own.

- The PSS2B was split off from the main model and then put into an isolated test bed with defined input signals applied to the power and speed and then

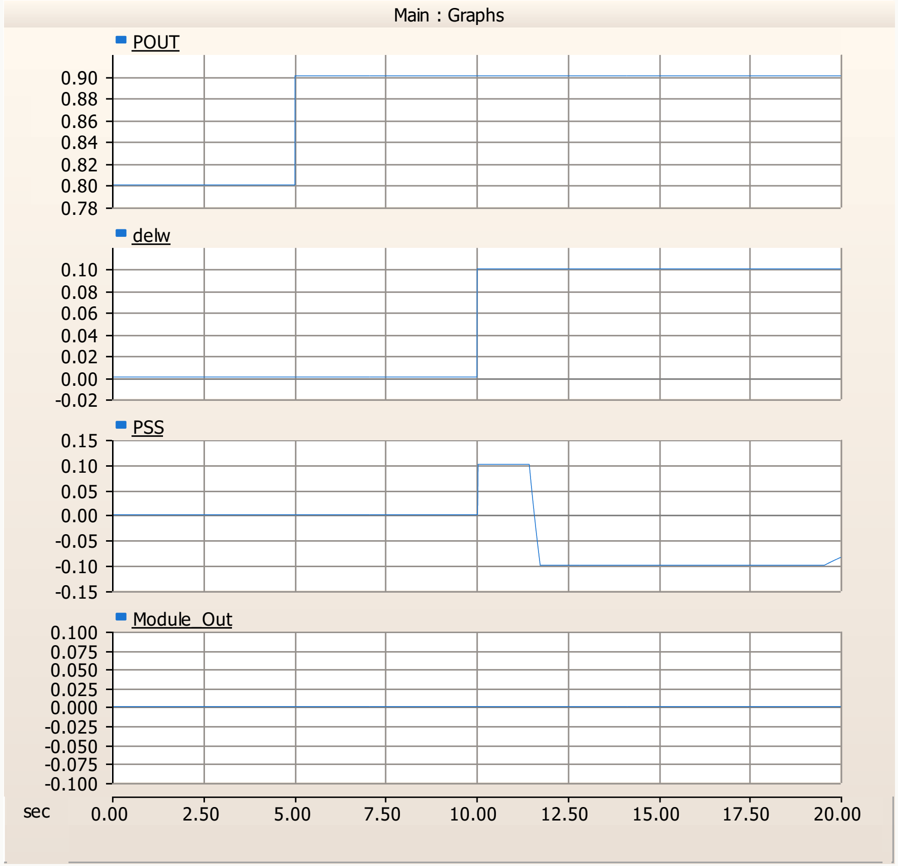
- Using the AI a test regime was setup with the actual PSS settings and the input signals and output signals checked against the expected response for the transfer functions.

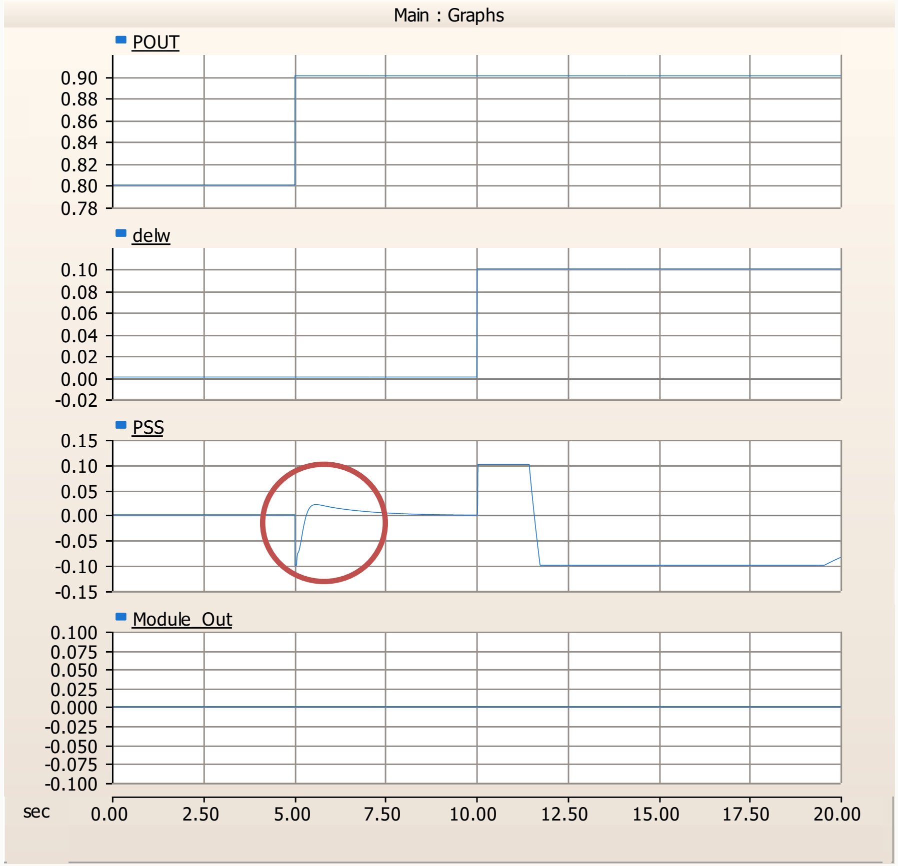
- IEEE 421.5 has a key concept that if a washout stage is set with a zero-time constant is bypassed.

- Instead, the PSS2B module was resolving this to a value of 0, which was effectively removing the whole power branch damping function.

Step Response - Results

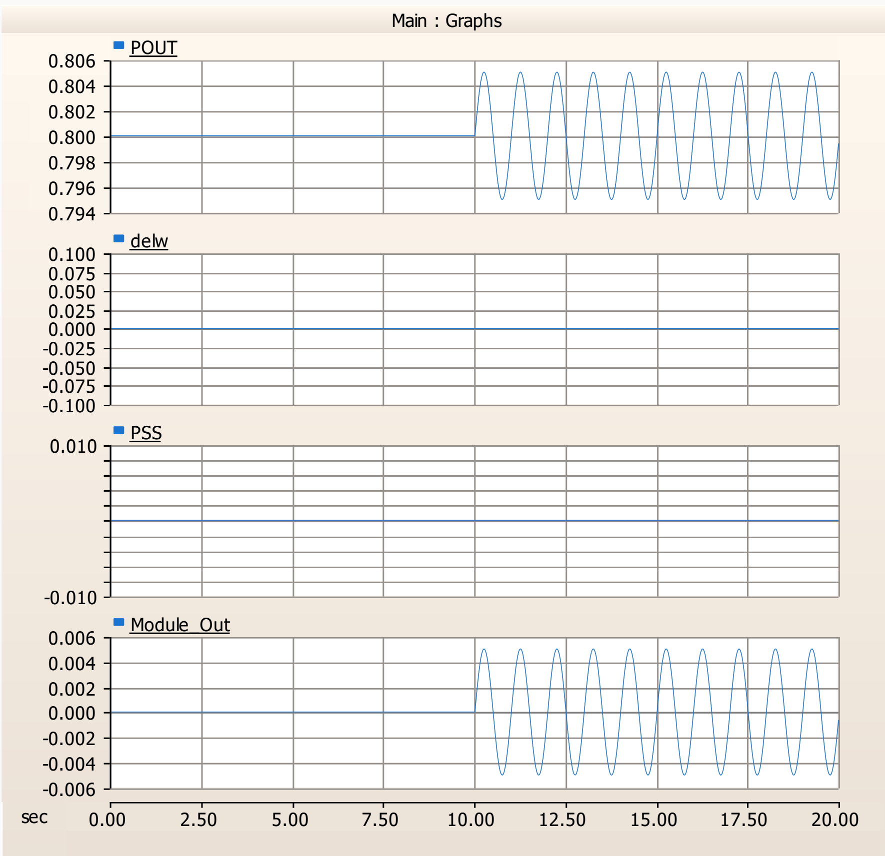
- Plot to simple step response.
  - Plot on the left shows no response to the ΔP (Tw4 = 0.0)
  - Plot on the right shows the correct response to the ΔP (Tw4 = 9999)

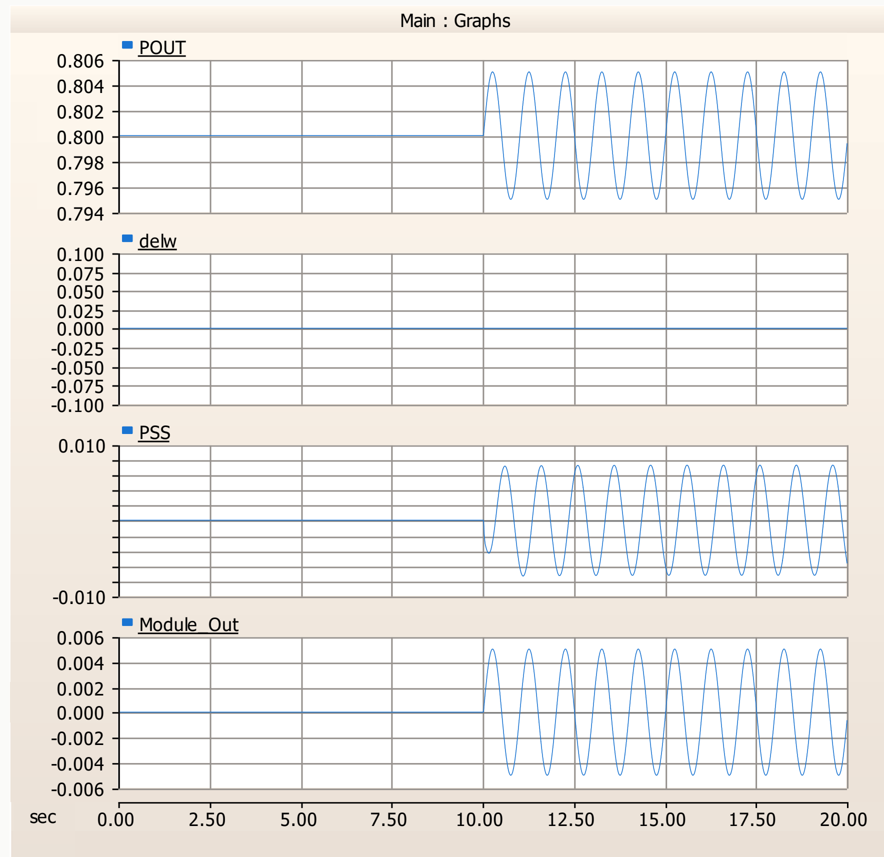




Disturbing Signal Response - Results

- Disturbing sinusoid applied
  - Plot on the left shows no response to the ΔP (Tw4 = 0.0)
  - Plot on the right shows the correct response to the ΔP (Tw4 = 9999)





The Solution

- After identifying the bug inside the PSS2B model, the solution was to set the Tw4 value to a very high number of 9999, this reduces the transfer block to a value of 1, which is almost the same as a bypass.
- For a little bit of extra work, using the MCP interface then the AI was used to build a new PSS2B model entirely in FORTRAN and then test it.
- The adjusted model and the work around were both viable solutions and let the simulation studies be completed.
- Issue followed up and confirmed with PSCAD technical support!

Summary

- The use of AI was a huge help in this and is very much playing to the AI strengths, of mathematical modelling and a direct MCP plug-in to PSCAD.
- A set of defined transfer functions such as a PSS should always have a clear and well-defined response. The AI was able to define a test plan to check and sweep the range of responses of the PSS and identify that the outputs were not as expected.
- It was then able to make the intuitive leap about where the problem was – without any direct prompting from myself, and then suggest a work around of setting the Tw4 value high.
- With only a little more prompting it was able to rebuild the PSS2B from scratch directly in FORTRAN and then retest.