

TO: WECC MVS

FROM: Jamie Weber, PowerWorld Corporation, [weber@powerworld.com](mailto:weber@powerworld.com)

SUBJECT: Modify initialization of Hydro Governor Models which modify Turbine Gain when Mechanical Power is too large in initial condition

DATE: September 16, 2021

---

At the WECC MVS meeting in August 2021, Sean Patterson from USBR pointed out that when initializing some Hydro Governor Models in a situation where the Mechanical Power ( $P_{mech}$ ) was too large in the initial condition for the model to support, software tools were modifying the turbine gain (often called  $A_t$ ) to a value that allows the  $P_{mech}$  to be achieved. Sean suggested that the value of  $A_t$  is a physical parameter of the turbine and adjusting the turbine gain should not be done. This impacts the dynamic behavior of the governor model. While all agree the best thing to do would be to modify the case so that the governor input parameters are consistent with the power flow case, Sean and others advised that if any auto-correction is done it would be better to modify the input parameter value  $H_{dam}$  instead of  $A_t$ . At least  $H_{dam}$  is a physical parameter that might change (it's the water level or head).

The following document describes the mathematics necessary to do this and also provides short Pseudo-Code for this.

It is recommended that at the November 2021 meeting a decision be made regarding how to handle this situation. Choices are

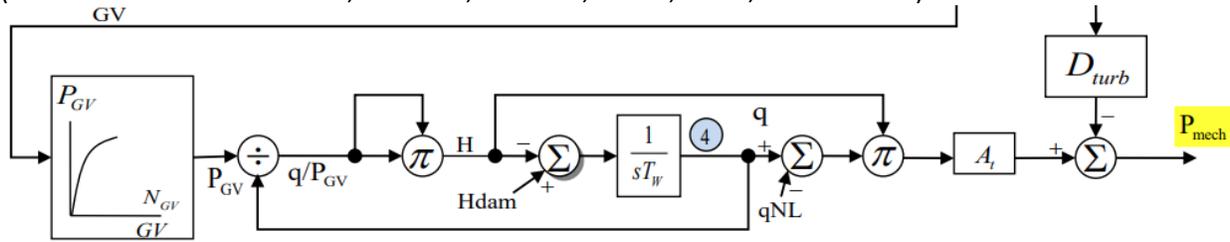
1. Continue modifying the  $A_t$  parameter
2. Change the hydro governor models named HYGOV, HYGOVD, HYGOVR, HYPID, HYG3, HYGOV4, and H6E so that they instead modify the value of  $H_{dam}$ .

A discussion should be had at this meeting and a request made to software developers to follow whatever the recommendation of the WECC MVS.

## Initialization of Hydro Governor Models

Many hydro governor models have a portion of the block diagram that looks as follows.

(This includes models HYGOV, HYGOVD, HYGOVR, HYPID, HYG3, and HYGOV4)



Part of this block diagram is the nonlinear gain block that maps GV to Pgv (Gate value to Power).

Typically these nonlinear gain blocks have a maximum value of 1.00, but regardless there is a maximum possible value of Pgv from this. Let's call that maximum value Pgvmax.

To initialize this model, the following equations are solved. The input to the integrator block  $1/sT_w$  must be zero, which means the value going into the multiplication block right before  $A_t$  must be  $H_{dam}$ . Thus we can calculate

$$q = \frac{P_{mech}}{A_t H_{dam}} + q_{nl}$$

The value  $q$  is passed back to the divider block right after the nonlinear gain block.

$$P_{gv} = \frac{q}{\sqrt{H_{dam}}}$$

The trouble you run into is if the value of  $P_{gv} > P_{gvmax}$ .

Presently software tools are modifying the  $A_t$  such that  $P_{gv}$  can be attained. This is done by

$$P_{gvmax} = \frac{q}{\sqrt{H_{dam}}} = \frac{1}{\sqrt{H_{dam}}} \left( \frac{P_{mech}}{A_t H_{dam}} + q_{nl} \right)$$

$$P_{gvmax} \sqrt{H_{dam}} = \frac{P_{mech}}{A_t H_{dam}} + q_{nl}$$

$$P_{gvmax} \sqrt{H_{dam}} - q_{nl} = \frac{P_{mech}}{A_t H_{dam}}$$

$$A_t = \frac{P_{mech}}{H_{dam}} \left( \frac{1}{P_{gvmax} \sqrt{H_{dam}} - q_{nl}} \right) : \text{NEW value of } A_t \text{ used.}$$

Sean Patterson from USBR has pointed out that the value of  $A_t$  is a physical parameter of the turbine and adjusting the turbine gain should not be done. This impacts the dynamic behavior of the governor model. While all agree the best thing to do would be to modify the case so that the governor input parameters are consistent with the power flow case, Sean and others advised that if any auto-correction is to be done it would be better to modify the input parameter value  $H_{dam}$  instead of of  $A_t$ .

To achieve that we need to solve this same equation for  $H_{dam}$  which is done as follows

$$P_{gvmax} \sqrt{H_{dam}} = \frac{P_{mech}}{A_t H_{dam}} + q_{nl}$$

Multiply both sides by  $H_{dam}$ .

$$P_{gvmax} H_{dam} \sqrt{H_{dam}} = \frac{P_{mech}}{A_t} + H_{dam} q_{nl}$$

Write as

$$P_{gvmax} (H_{dam})^{3/2} - q_{nl} (H_{dam})^{2/2} - \frac{P_{mech}}{A_t} = 0$$

This is a cubic equation which can be solved using the following nomenclature

$$a = P_{gvmax}$$

$$b = -q_{nl}$$

$$c = 0$$

$$d = -\frac{P_{mech}}{A_t}$$

$$x = \sqrt{H_{dam}}$$

Solve the cubic equation  $ax^3 + bx^2 + cx + d = 0$

This can be solved using Cardano's Formula ([https://proofwiki.org/wiki/Cardano%27s\\_Formula](https://proofwiki.org/wiki/Cardano%27s_Formula))

I won't bother proving it here but went through this on paper. Because in our equation the parameter  $c = 0$ , then this means that this equation has 1 real root and 2 complex number roots. We only want the real number solution.

$$Q = \frac{c}{3a} - \left(\frac{b}{3a}\right)^2 = -\left(\frac{b}{3a}\right)^2$$

$$R = \frac{bc}{6a^2} - \frac{d}{2a} - \left(\frac{b}{3a}\right)^3 = \left(-\frac{d}{2a}\right) + \left(-\frac{b}{3a}\right)^3$$

$$S = \sqrt[3]{R + \sqrt{Q^3 + R^2}}$$

$$T = \sqrt[3]{R - \sqrt{Q^3 + R^2}}$$

$$x = S + T - \frac{b}{3a}$$

Now apply these equations to our situation and be careful to define variables so the numbers are always positive for our situation to make it clear there will never be any weird numerical issues.

$$m = -\frac{d}{2a} = \frac{P_{mech}}{2A_t P_{gvmax}} \quad \text{Note: } m \text{ is always a positive number}$$

$$n = -\frac{b}{3a} = \frac{q_{nl}}{3P_{gvmax}} \quad \text{Note: } n \text{ is always a positive number}$$

$$R = +m + n^3$$

$$Q = -n^2$$

$$Q^3 + R^2 = -n^6 + (m + n^3)^2 = -n^6 + (m^2 + 2mn^3 + n^6) = m^2 + 2mn^3$$

Note:  $Q^3 + R^2$  is always a positive number because both  $m$  and  $n$  are positive

$$S = \sqrt[3]{R + \sqrt{R^2 - n^6}}$$

$$T = \sqrt[3]{R - \sqrt{R^2 - n^6}}$$

$$x = S + T + n$$

$$H_{dam} = x^2$$

Note:  $\sqrt{Q^3 + R^2} \leq R$  because in our example  $Q^3 = -n^6$  which is always negative.

Expand that out and you'd get the following, but I'd just code with the steps above

You can go through the steps, but if  $q_{nl} = 0$ , then this is a lot simpler. This means that the value  $n = 0$  which means  $R = m$  and  $Q = 0$ . This gives

$$S = \sqrt[3]{2R} = \sqrt[3]{\frac{2P_{mech}}{2A_t P_{gvmax}}} = \sqrt[3]{\frac{P_{mech}}{A_t P_{gvmax}}} = \left(\frac{P_{mech}}{A_t P_{gvmax}}\right)^{1/3}$$

$$H_{dam} = x^2 = S^2$$

$$H_{dam} = \left(\frac{P_{mech}}{A_t P_{gvmax}}\right)^{2/3}$$

Now that we're sure there are never going to be weird numerical situations, let's just code this as follows

$$m = \frac{P_{mech}}{2A_t P_{gvmax}}$$

$$n = \frac{Q_{nl}}{3P_{gvmax}}$$

$$R = m + n^3$$

$$Q^3 + R^2 = m^2 + 2mn^3$$

$$S = \sqrt[3]{R + \sqrt{Q^3 + R^2}}$$

$$T = \sqrt[3]{R - \sqrt{Q^3 + R^2}}$$

$$H_{dam\ new} = (S + T + n)^2$$

$$P_{gv} = P_{gvmax}$$

$$q = P_{gvmax} \sqrt{H_{dam}}$$

In the end the pseudo-code is very simple and is as follows

```

InputPmech      : Present Pmech initializing governor too
Pgvmax, Qnl, At, : input parameters to model
Hdam
NewPgv, NewQ    : Make sure you update these values
M, N, Q3R2, S, T : local variables to this function

```

Procedure FixHdamPgvQToAchieveInputPMech

```

  (InputPmech : Input
   Pgvmax, Qnl, At : Input
   Hdam : Output
   NewPgv, NewQ : Output)
{
  M   = InputPmech / (2*At*Pgvmax)
  N   = Qnl / (3*Pgvmax)
  Q3R2 = M*(M + 2*N*sqr(N))
  R   = M + N*sqr(N)
  S   = Power(R + sqrt(Q3R2), 1/3)
  T   = Power(R - sqrt(Q3R2), 1/3)
  ??? Add some logging to let user know what is being changed
  Hdam = sqrt( S + T + N )      // change value of Hdam
  NewPgv = Pgvmax              // update Pgv value
  NewQ   = Pgvmax*sqr(Hdam)    // update Q value
}

```

Alternative Approach would have been to write the quadratic equation in function of  $q$  instead

$$q = \frac{P_{mech}}{A_t H_{dam}} + q_{nl}$$

$$H_{dam} = \left(\frac{q}{P_{gv}}\right)^2$$

$$q = \frac{P_{mech}}{A_t \left(\frac{q}{P_{gv}}\right)^2} + q_{nl}$$

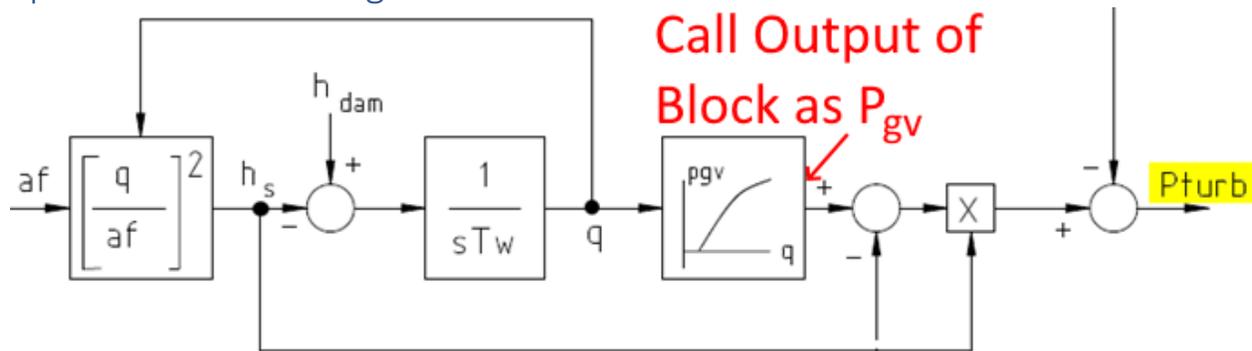
$$q - q_{nl} = \frac{P_{mech}}{A_t \left(\frac{q}{P_{gv}}\right)^2}$$

$$(q - q_{nl})q^2 = \frac{P_{gv}^2 P_{mech}}{A_t}$$

$$q^3 - q_{nl}q^2 - \frac{P_{gv}^2 P_{mech}}{A_t} = 0$$

Solve that equation for  $q$  and then calculate  $H_{dam}$  from that works just fine too and will give the same result.

### Special Case for H6E governor



The nonlinear function for H6E is in a different location. Thus to modify  $H_{dam}$  is easier here.

$$P_{turb} = P_{gv} H_{dam}$$

If we end up calculating that  $P_{gv} > P_{gvmax}$  then we change the value of

$$H_{dam} = \frac{P_{turb}}{P_{gvmax}}$$

Initialization from there proceeds as normal.