Description and Optimization of Beta Feedback

Stefan Gerhardt

Multiple Stages In the Algorithm

- Compare filtered β_N value from rtEFIT to a request, and compute an error.
- Use PID on the error to compute a new requested power.
- Use requested power, source powers, and “batting order” to determine the duty cycles for each source.
- Use the duty cycles and min. on/off times to determine when to block.

What do the Gains Mean?

$$\beta_T = \frac{2\mu_0 W_{MHD}}{VB_T^2}, \quad \beta_N = 100\frac{\beta_T a B_T}{I}$$

$$\beta_N = 100\frac{a}{I_P}\frac{2\mu_0 W_{MHD}}{VB_T} \Rightarrow W_{MHD} = \beta_N \frac{I_P V B_T}{200\mu_0 a}$$

$$W_{MHD} = \frac{P_{inj}}{\tau} \Rightarrow \Delta P_{inj} = \tau \frac{I_P V B_T}{200\mu_0 a}\Delta\beta_N$$

Final Equation

$$\Delta P_{inj} = P_{\beta_N}\overline{C}_{\beta_N} e + I_{\beta_N}\overline{C}_{\beta_N}\int e dt + D_{\beta_N}\overline{C}_{\beta_N}\frac{de}{dt}$$

$$e = -\left(\beta_{N,RTEFIT} - \beta_{N,reqeust}\right)$$

$$\overline{C}_{\beta_N} = \tau \frac{I_P V B_T}{200\mu_0 a}\cdot\frac{dt}{0.001}$$

V: 12 m^3
I_P: Actual Value
B_T: Actual Value
a: 0.6 m
τ: 0.04 sec
dt: time step between iterations

Knobs That You Can Turn

- Filter time constant on the β_N value sent from rtEFIT.
  - Useful for smoothing transients in the measured β_N.
- Proportional, integral, and derivative gains.
  - Determines the response of the system to transients, and whether you ever get to zero error.
- Batting order array
  - Determines which sources modulate
  - Switch to a different source if a given source reaches the maximum number of blocks.
  - Also able to prevent A modulations, to keep MSE and CHERS.
- Source powers
  - Can be adjusted in order to prevent modulations.
- Minimum Source On/Off Times.
  - Smaller values will lead to better control, but possibly at the expense of source reliability.
  - 20 msec. has seemed OK so far.

IDL Code Written To Exactly Simulate the Algorithm Performance

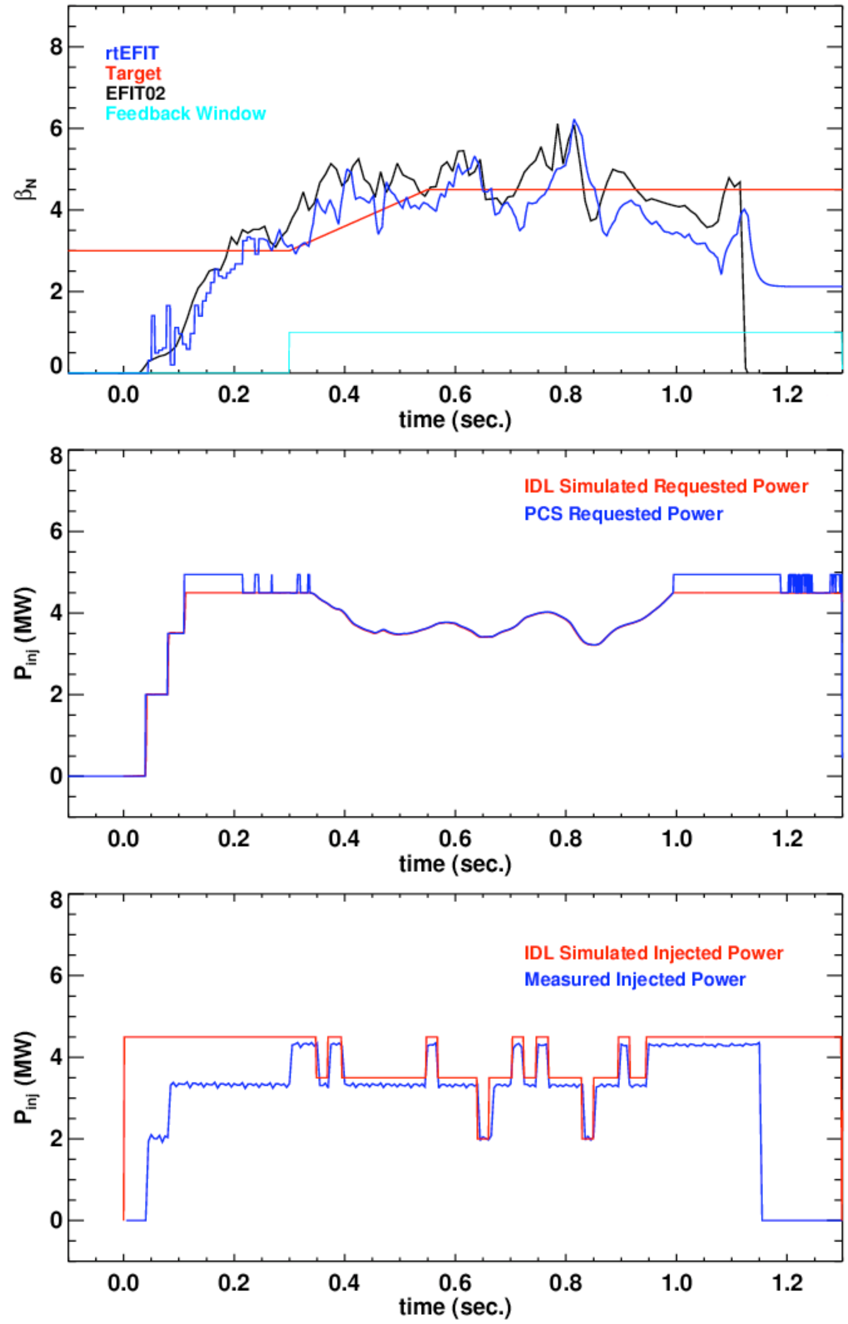
135134

P=10

I=0

D=0

τ=0.01 sec



Adding Integral Gain Alone May Make It More Sluggish (= stable)

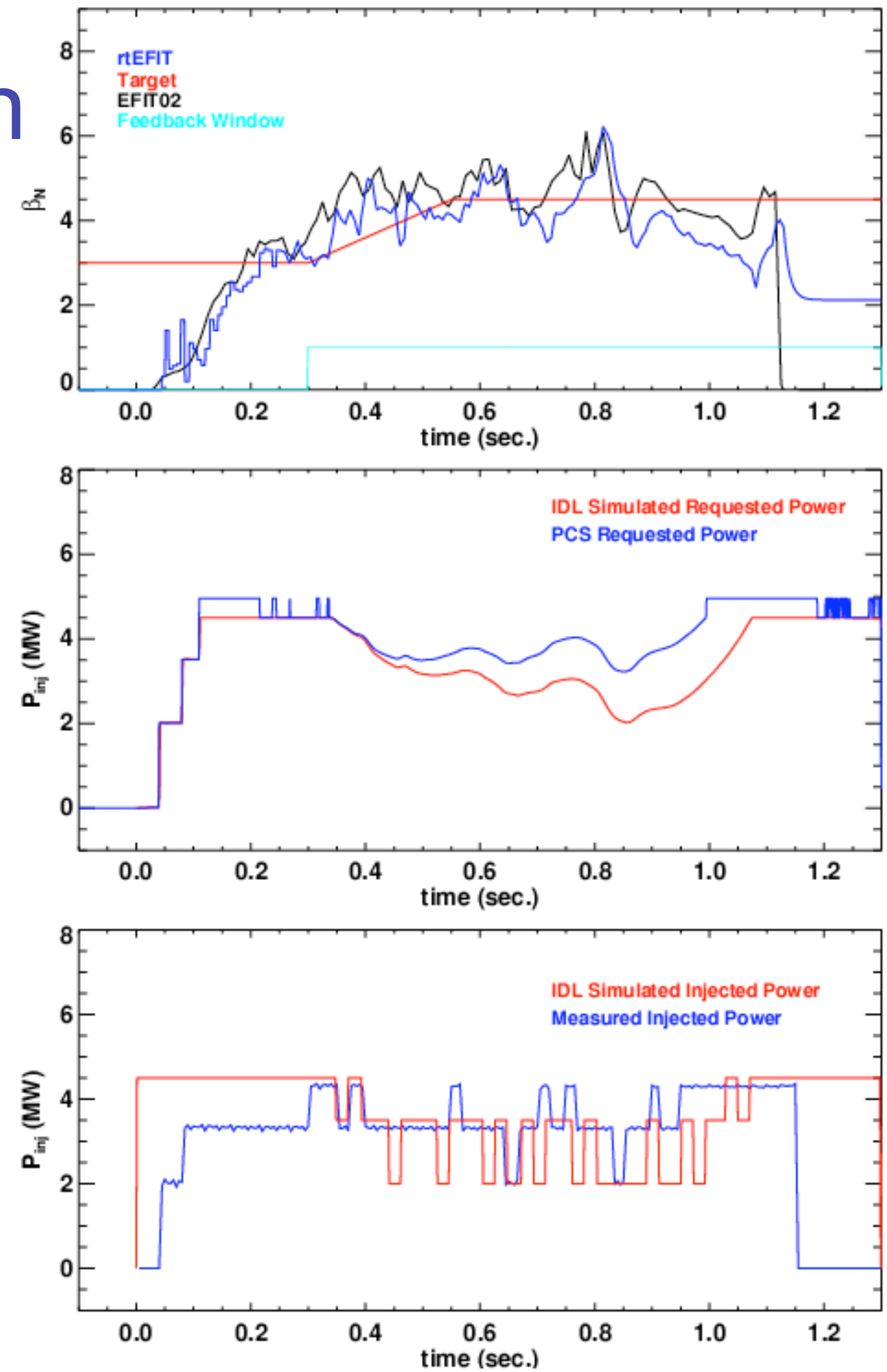
135134

P=10

I=30

D=0

τ=0.01 sec



Derivative Gain Can Make the Algorithm More Responsive to Transients

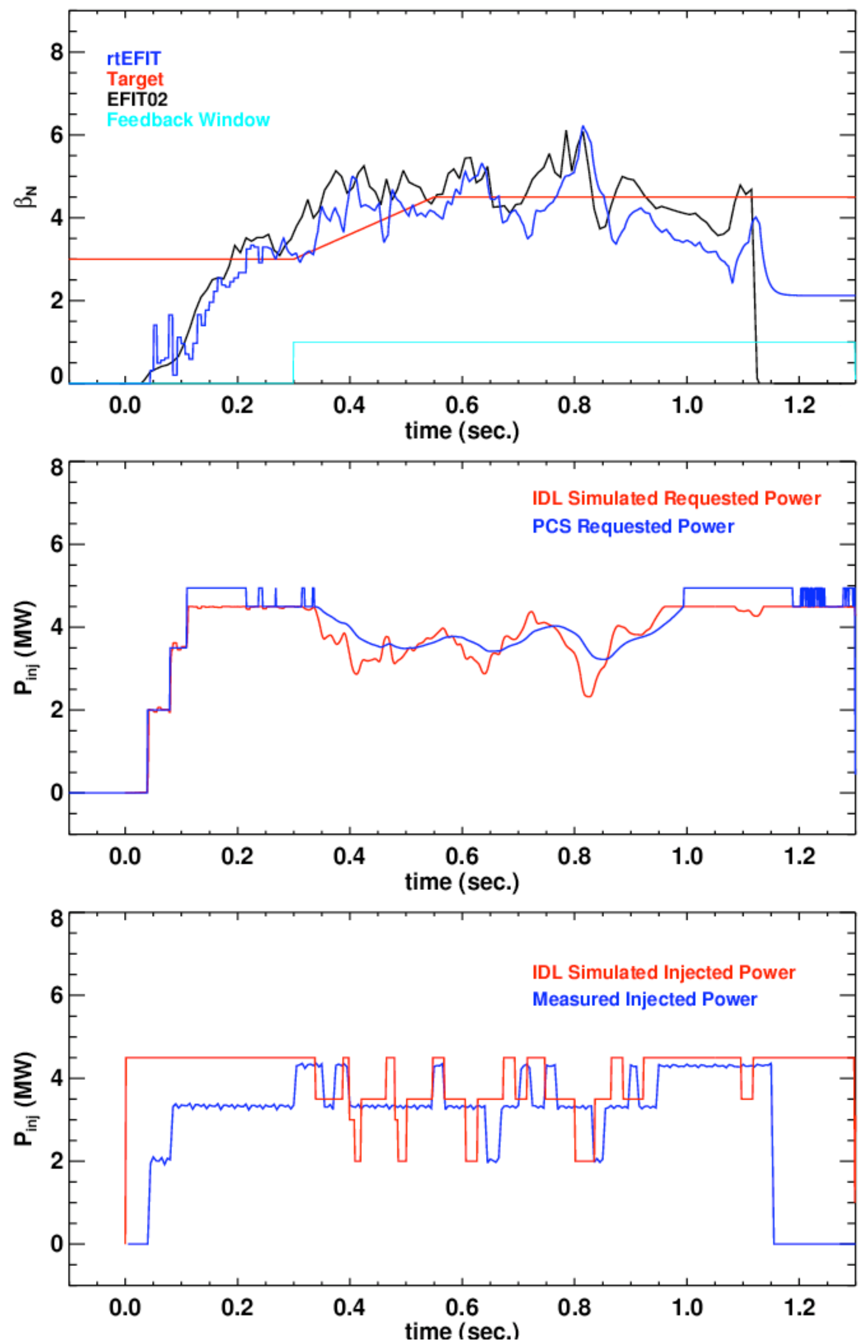
135134

P=10

I=0

D=0.5

τ=0.01 sec



Can Smooth the Response Some By Adding Significant I +D…May be Too Much I

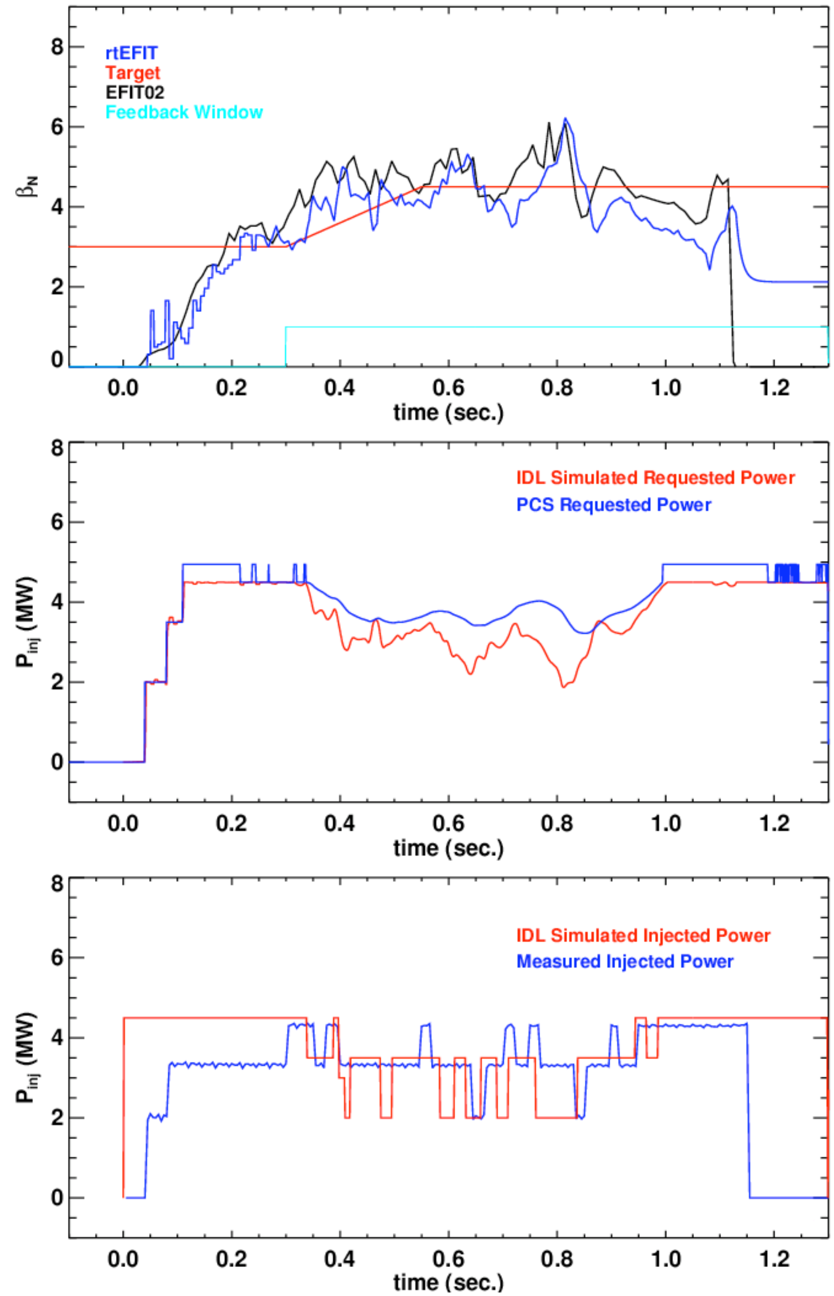
135134

P=10

I=30

D=0.5

τ=0.01 sec



Optimal Setting May Involve a Longer Time Constant, Derivative Gain, and Small Integral Gain

135134

P=10

I=5

D=0.5

τ=0.015 sec