

BAB 4 ANALISA DAN PEMBAHASAN

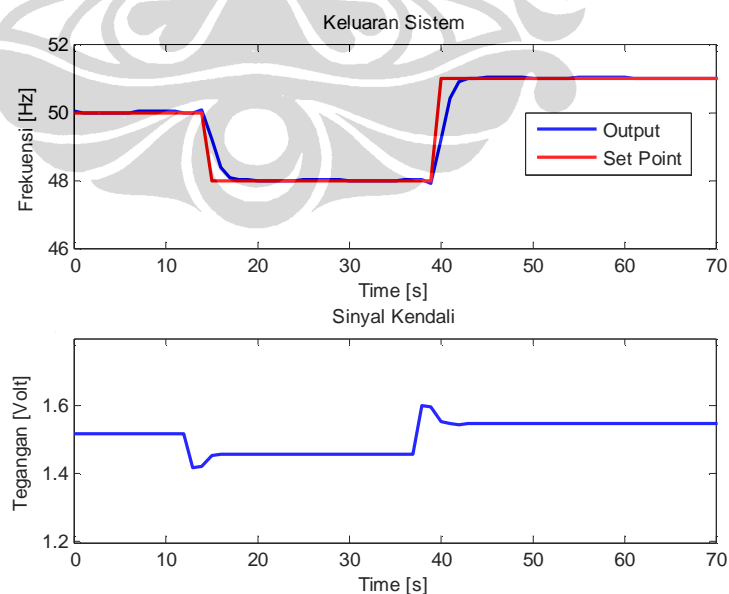
Tujuan dari uji coba dan analisa adalah untuk mengetahui kinerja dari pengendali MPC tanpa *constraint* dan MPC tanpa *constraint* dengan observer dengan parameter penalaan yang berbeda-beda dan gangguan berupa perubahan beban. Perancangan algoritma pengendali menggunakan program Matlab.

4.1 Simulasi pengendali MPC tanpa *constraint* pada sistem PLTMH

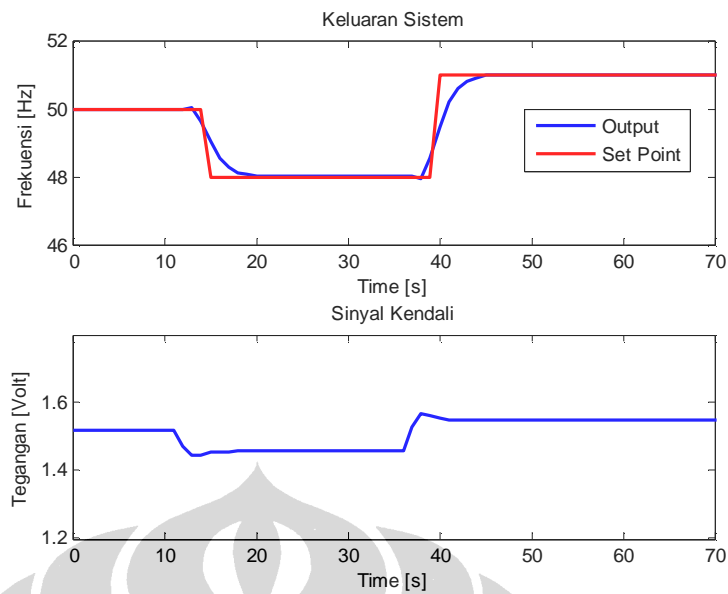
Simulasi pengendali MPC tanpa *constraint* pada sistem PLTMH dimulai pada titik kesetimbangan sistem. Simulasi dilakukan dengan *set point* bervariasi dan beban tetap.

4.1.1 Percobaan dengan nilai *Control Horizon* tetap dan nilai *prediction horizon* bervariasi

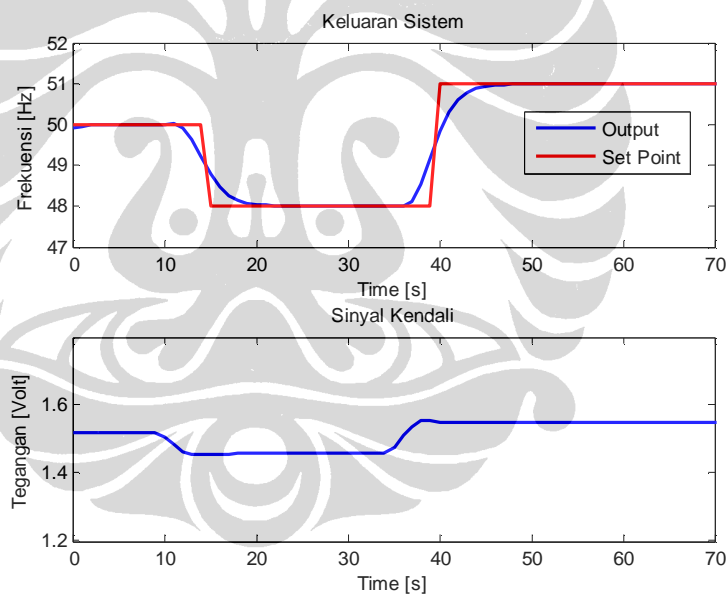
Untuk melihat pengaruh *prediction horizon* (H_p) terhadap hasil pengendalian MPC, dilakukan beberapa percobaan dengan nilai *control horizon* (H_u) dibuat tetap yaitu sebesar 1, dan nilai *prediction horizon* dibuat bervariasi yaitu 3, 6, dan 9. Faktor bobot $\mathbf{Q} = \mathbf{I}_{HP}$, $\mathbf{R} = \mathbf{I}_{HU}$.



Gambar 4.1 Keluaran dan sinyal kendali sistem dengan nilai $H_p=3$



Gambar 4.2 Keluaran dan sinyal kendali sistem dengan nilai $H_p = 6$



Gambar 4.3 Keluaran dan sinyal kendali sistem dengan nilai $H_p = 9$

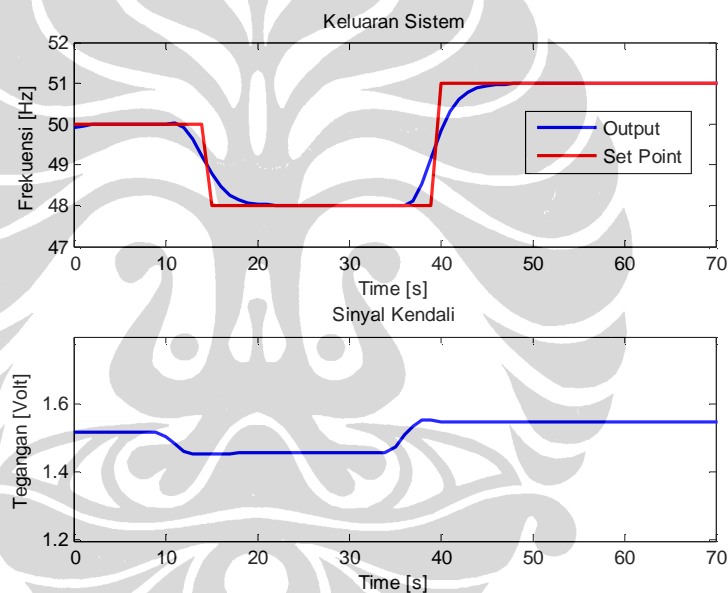
Hasil simulasi dengan *control horizon* tetap dan *prediction horizon* yang berbeda terdapat pada Gambar (4.1), (4.2), (4.3). Dari hasil simulasi terlihat bahwa jika nilai H_p diperbesar maka waktu yang diperlukan oleh keluaran sistem untuk mencapai *set point* akan lebih lama namun sinyal kendalinya akan semakin kecil. Dengan variasi perubahan sinyal kendali semakin kecil jika nilai H_p diperbesar

Universitas Indonesia

maka variansi perubahan *output* juga akan kecil sehingga menyebabkan *output* lebih lama mencapai *set point*-nya

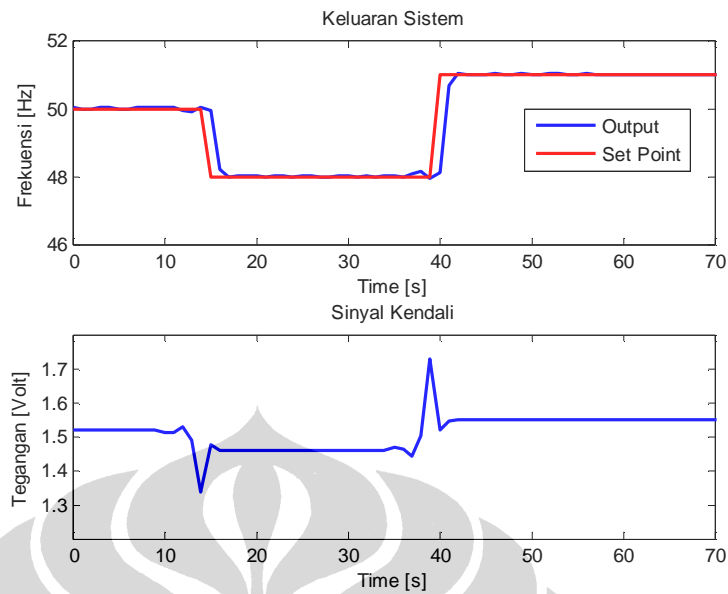
4.1.2 Percobaan dengan nilai *prediction horizon* tetap dan nilai *control horizon* bervariasi

Untuk melihat pengaruh *control horizon* (H_u) terhadap hasil pengendali MPC, dilakukan beberapa percobaan dengan nilai *prediction horizon* (H_p) dibuat tetap, yaitu sebesar 9, dan nilai *control horizon* yang digunakan bervariasi yaitu 1, 3, dan 5. Faktor bobot kesalahan (Q) yang digunakan adalah I_{HP} sedangkan faktor bobot perubahan sinyal kendali (R) yang digunakan adalah I_{HU} .

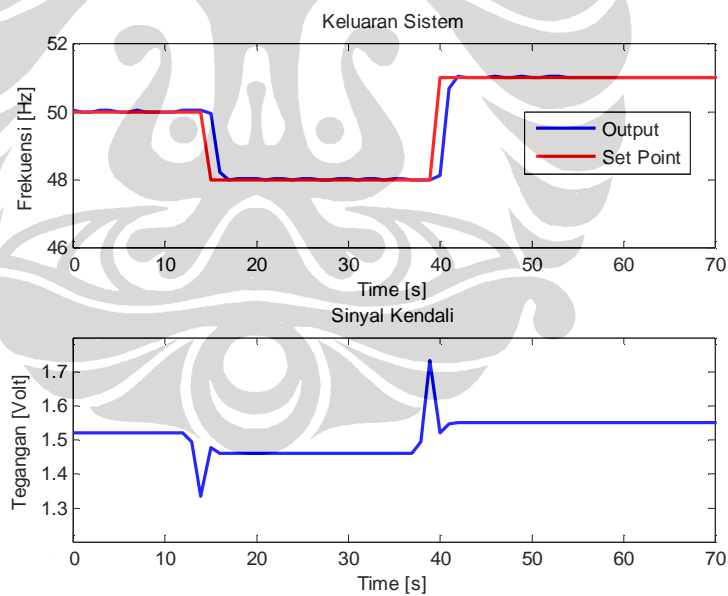


Gambar 4.4 Keluaran dan sinyal kendali sistem dengan nilai *control horizon* $H_u=1$

Gambar (4.4), (4.5), dan (4.6) adalah hasil simulasi dengan nilai H_p tetap dan nilai H_u bervariasi. Dari gambar tersebut terlihat bahwa jika nilai *control horizon* (H_u) diperbesar maka sinyal kendali ingin berubah sebesar-besarnya supaya keluaran sistem dapat dengan segera mencapai *set point*-nya sehingga perubahan sinyal kendali akan memiliki variansi yang besar pada saat akan terjadi perubahan nilai *set point*. Perubahan sinyal kendali dengan variansi yang besar akan menyebabkan keluaran sistem menjadi tidak bagus.



Gambar 4.5 Keluaran dan sinyal kendali sistem dengan nilai *control horizon* $H_u=3$



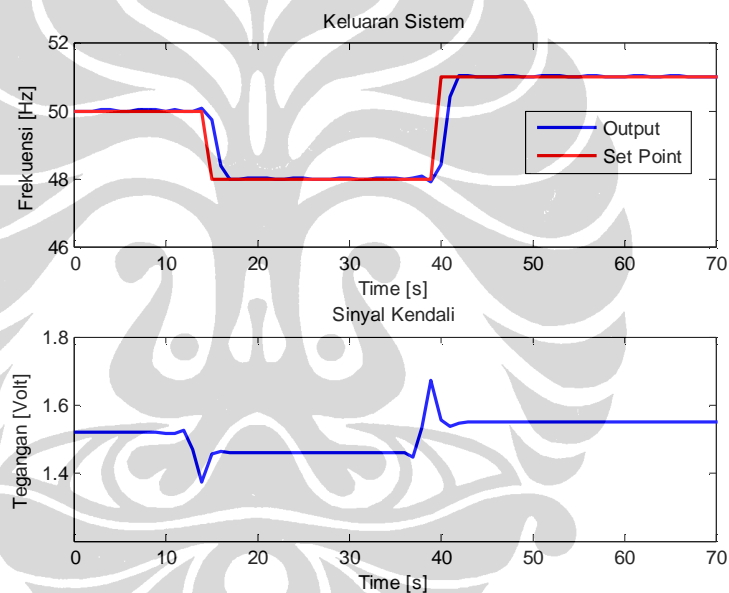
Gambar 4.6 Keluaran dan sinyal kendali sistem dengan nilai *control horizon* $H_u=5$

Untuk sistem PLTMH jika *control horizon* semakin besar maka kemampuan pengendali MPC untuk memprediksi sinyal kendali akan menurun

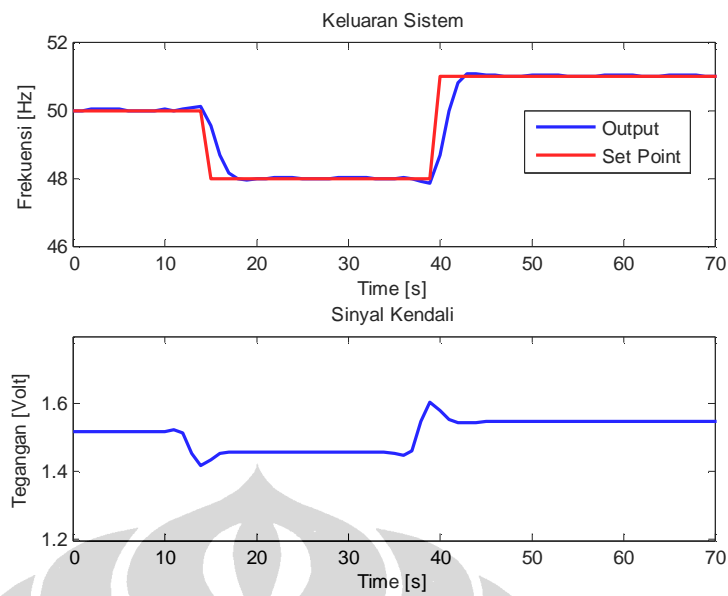
karena untuk mendeteksi kecepatan putar sangat sulit seperti terlihat pada Gambar 4.6.

4.1.3 Pengaruh nilai faktor bobot perubahan sinyal kendali (R) pada hasil pengendali MPC

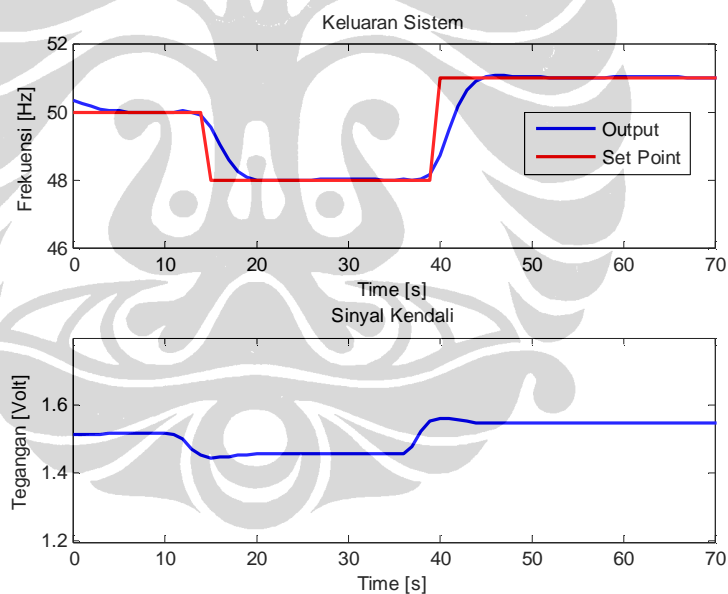
Faktor bobot R digunakan untuk mengendalikan perubahan sinyal kendali. Untuk melihat pengaruh faktor bobot R pada hasil pengendali, dilakukan uji coba pada sistem dengan membuat nilai diagonal matriks R berbeda-beda untuk setiap percobaan, yaitu 10, 100, dan 1000 sedangkan nilai parameter pengendali lainnya dibuat tetap, $H_p = 9$, $H_u = 4$ dan $Q = I_{H_p}$



Gambar 4.7 Keluaran dan sinyal kendali sistem dengan nilai sinyal kendali $R=10$



Gambar 4.8 Keluaran dan sinyal kendali sistem dengan nilai sinyal kendali $R=100$



Gambar 4.9 Keluaran dan sinyal kendali sistem dengan nilai sinyal kendali $R=1000$

Dari hasil simulasi pada Gambar (4.7), (4.8), (4.9) terlihat bahwa semakin besar nilai matriks R , perubahan sinyal kendali menjadi semakin ditekan sehingga keluaran sistem menjadi semakin halus.

Dari hasil simulasi dengan penalaan yang berbeda-beda didapatkan kombinasi parameter terbaik untuk sistem PLTMH, yaitu $H_p = 9$, $H_u = 1$, $\underline{Q} = \underline{I}_{HP}$ dan $\underline{R} = \underline{I}_{HU}$.

4.2 Simulasi dengan pengendali MPC tanpa *constraint* dan observer

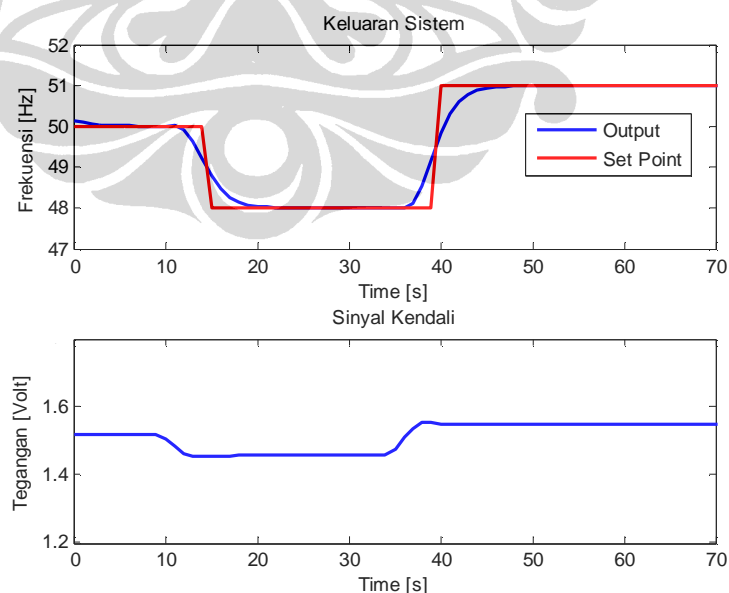
Full-order state observer digunakan untuk mengestimasi variabel keadaan sistem sehingga dapat menggantikan keterbatasan *hardware* seperti sensor. Simulasi pengendali MPC tanpa *constraint* dengan observer menggunakan kombinasi parameter terbaik yang didapat dari hasil simulasi pada langkah sub bab (4.1). Nilai *pole* dari sistem yaitu :

$$\begin{aligned} I_1 &= 0.6238 & I_3 &= 0.0000 \\ I_2 &= 0.0015 & I_4 &= 0.2374 \end{aligned}$$

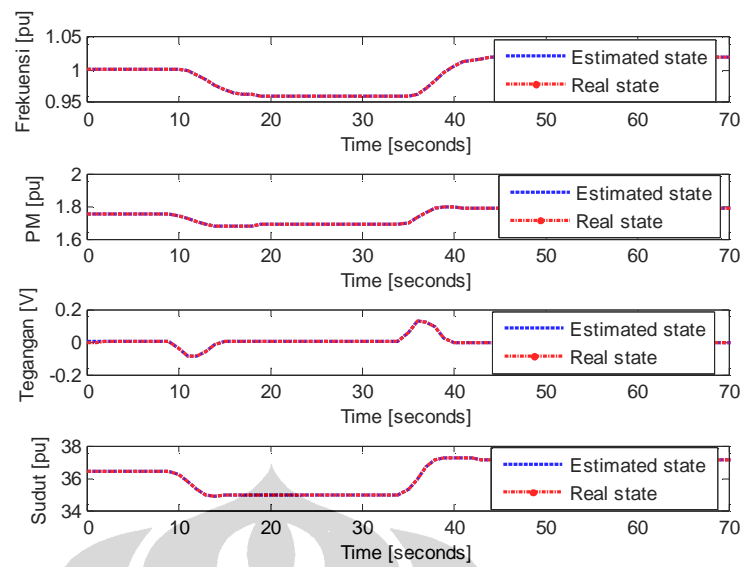
Untuk melihat pengaruh pergeseran *pole* pada hasil pengendali, dilakukan uji coba pada sistem dengan membuat nilai *pole* digeser ke kiri atau mendekati nol agar sistem yang diestimasi menjadi lebih stabil dan hanya nilai *pole* yang besar yang digeser.

Pada simulasi ke-1, nilai *pole* yang diinginkan adalah sebagai berikut:

$$\begin{aligned} m_1 &= I_1 - 0.1 = 0.5238 & m_3 &= I_3 = 0.0000 \\ m_2 &= I_2 = 0.0015 & m_4 &= I_4 - 0.1 = 0.1372 \end{aligned}$$



Gambar 4.10 Keluaran sistem dan sinyal kendali MPC tanpa *constraint* + observer



Gambar 4.11 Perbandingan Estimasi *state* dan aktual *state*

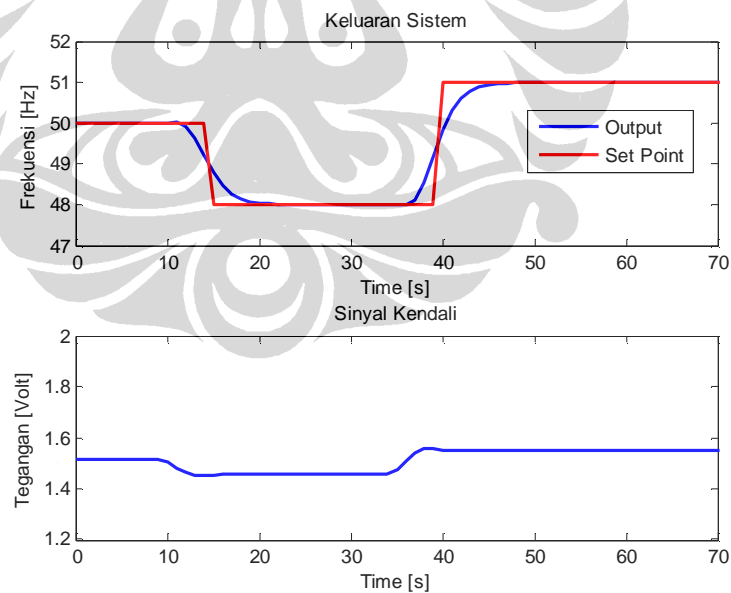
Simulasi ke-2, nilai *pole* yang diinginkan adalah sebagai berikut:

$$m_1 = I_1 - 0.3 = 0.3238$$

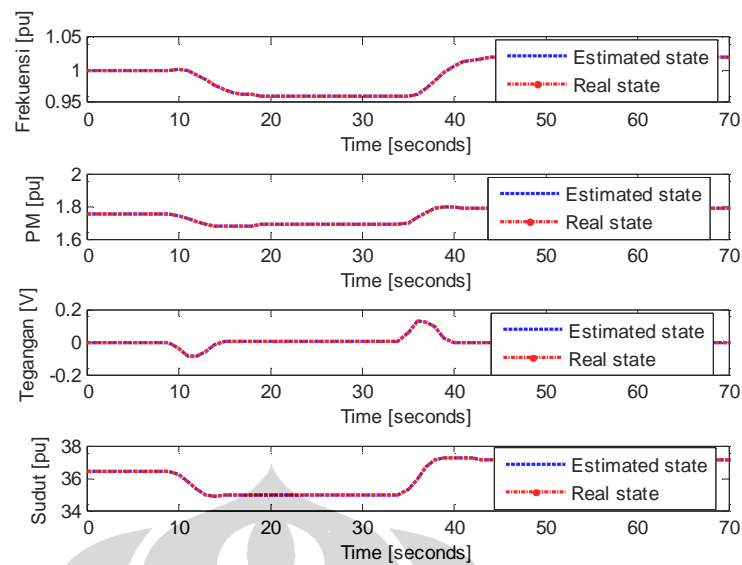
$$m_3 = I_3 = 0.0000$$

$$m_2 = I_2 = 0.0015$$

$$m_4 = I_4 - 0.2 = 0.0372$$



Gambar 4.12 Keluaran sistem dan sinyal kendali MPC tanpa *constraint* + observer



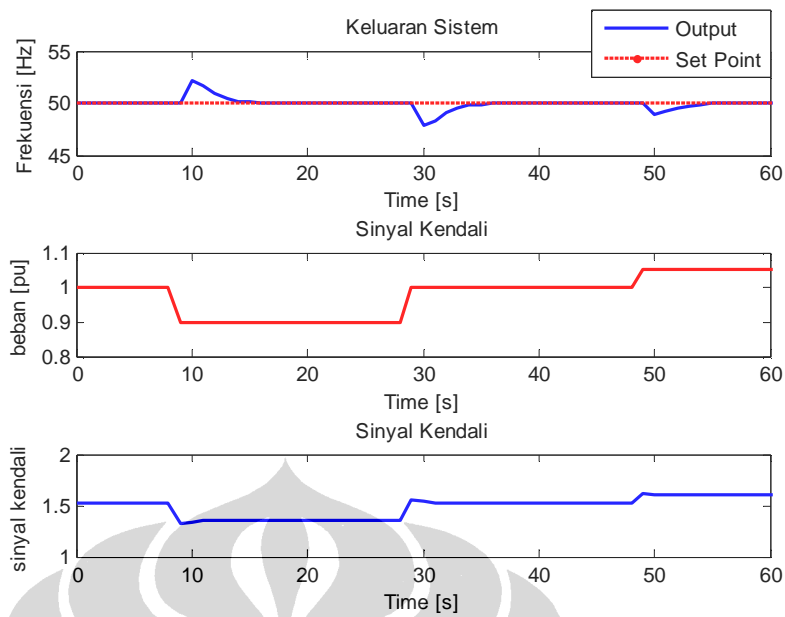
Gambar 4.13 Perbandingan estimasi *state* dan aktual *state*

Dari Gambar (4.10) dan (4.12) dapat dilihat bahwa jika *pole* digeser semakin mendekati nol maka keluaran sistem menjadi lebih baik. Hasil *state* estimasi juga semakin mendekati nilai *state* aktual nya.

4.3 Pengujian pengendali MPC tanpa *constraint* dengan perubahan beban

Pengujian ini dilakukan dengan memberikan nilai *set point* tetap yaitu 50 Hz dan pemberian gangguan berupa perubahan beban pada waktu pencuplikan tertentu. Tujuan pengujian ini untuk mengetahui keandalan dari pengendali dan kecepatan respon sistem terhadap perubahan beban. Parameter pengendali MPC yang digunakan yaitu $H_p = 9$, $H_u = 1$, $\underline{Q} = \underline{I}_{Hp}$ dan $\underline{R} = \underline{I}_{Hu}$.

Diasumsikan sistem telah bekerja pada kondisi *steady state*, perubahan beban adalah 5% dan 10 % atau 0.05 pu dan 0.1 pu dari total beban.



Gambar 4.14 Keluaran sistem dengan perubahan beban pada pengendali MPC tanpa *constraint*

Gambar 4.14 adalah respon keluaran sistem dengan perubahan beban. Dari Gambar tersebut dapat dilihat bahwa pada $t = 10$ detik terjadi perubahan beban berupa penurunan beban sebesar $0.1 \text{ pu} = 2.5 \text{ kW}$, maka frekuensi naik sampai 52 Hz. Karena nilai frekuensi yang dihasilkan tidak sama dengan nilai frekuensi pada *set point* maka sinyal *error* akan dikirim ke kontrol MPC dan kontrol MPC akan memberikan sinyal kendali kepada *gate* servomotor untuk memperkecil bukaan *gate*. Setelah mengalami perubahan beban, waktu yang diperlukan untuk kembali ke keadaan *steady state* adalah 5 detik.

Pada $t = 50$ detik terjadi perubahan beban berupa kenaikan beban sebesar $0.05 \text{ pu} = 1.25 \text{ kW}$, maka frekuensi akan turun menjadi 49.4 Hz. Karena nilai frekuensi yang dihasilkan tidak sama dengan nilai frekuensi pada *set point* maka sinyal *error* akan dikirim ke kontrol MPC dan kontrol MPC akan memberikan sinyal kendali kepada *gate* servomotor untuk memperbesar bukaan *gate*. Setelah mengalami kenaikan beban, waktu yang diperlukan untuk kembali ke keadaan *steady state* adalah 5 detik.

4.4 Pengujian pengendali MPC tanpa *constraint* + Observer dengan perubahan beban

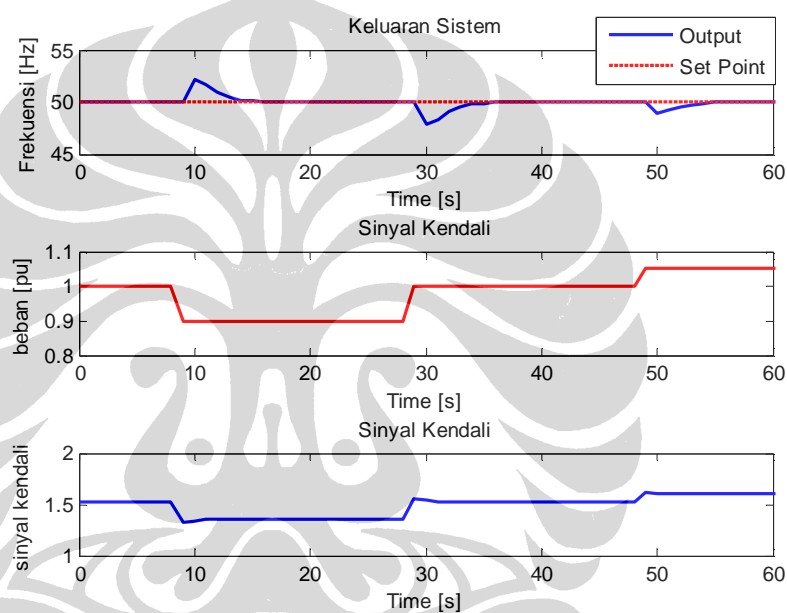
Pengujian pengendali MPC tanpa *constraint* + observer sama seperti pada langkah pengujian sub bab (4.3). Parameter pengendali MPC yang digunakan yaitu $H_p = 9$, $H_u = 1$, $Q = I_{Hp}$ dan $R = I_{Hu}$. Pole yang digunakan yaitu:

$$m_1 = I_1 - 0.3 = 0.3238$$

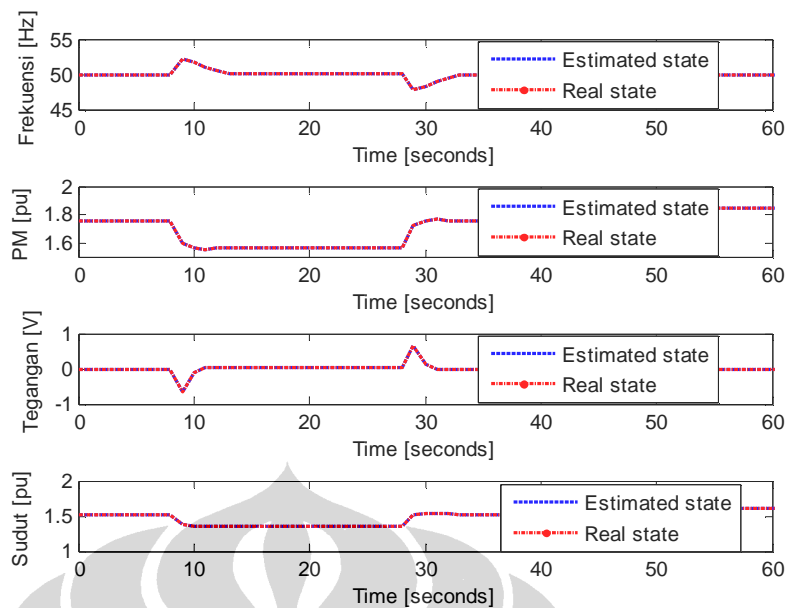
$$m_2 = I_2 = 0.0015$$

$$m_3 = I_3 = 0.0000$$

$$m_4 = I_4 - 0.2 = 0.0372$$



Gambar 4.15 Keluaran sistem dengan perubahan beban pada pengendali MPC tanpa *constraint* + observer



Gambar 4.16 Perbandingan Estimasi *State* dan Aktual *state*

Hasil yang diperoleh dari pengujian pengendali MPC tanpa *constraint* + observer sama dengan hasil pengujian pengendali MPC tanpa *constraint* tanpa observer. Dan estimasi *state* sudah menunjukkan kesesuaian dengan *real state*. Dari Gambar 4.16 dapat dilihat bahwa pada $t = 10$ detik terjadi perubahan beban berupa penurunan beban sebesar $0.1 \text{ pu} = 2.5 \text{ kW}$, maka frekuensi naik sampai 52 Hz. Karena nilai frekuensi yang dihasilkan tidak sama dengan nilai frekuensi pada *set point* maka sinyal *error* akan dikirim ke kontrol MPC dan kontrol MPC akan memberikan sinyal kendali kepada *gate* servomotor untuk memperkecil bukaan *gate*. Setelah mengalami perubahan beban, waktu yang diperlukan untuk kembali ke keadaan *steady state* atau 50 Hz adalah 5 detik.

4.5 Pengujian pengendali PID

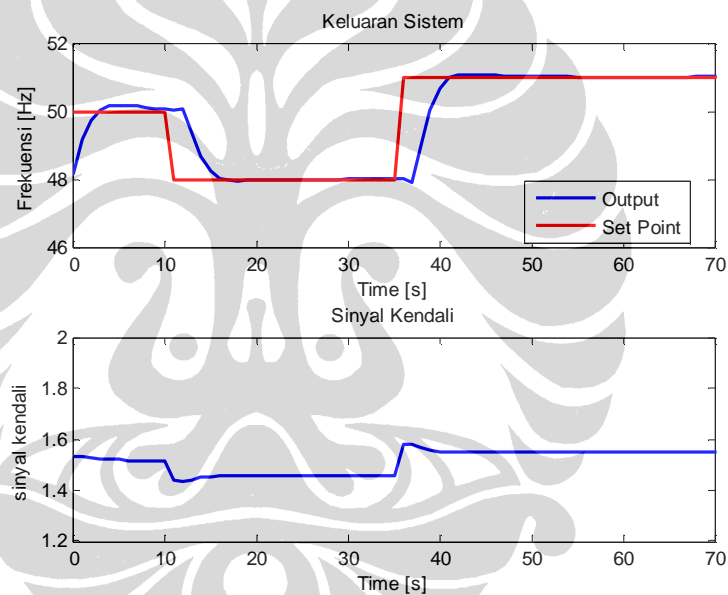
Pengendali yang banyak digunakan pada pembangkit listrik adalah pengendali PID. Pengendali PID mempunyai algoritma yang sederhana sehingga mudah diterapkan pada berbagai *plant*.

Pengujian pengendali PID dilakukan untuk membandingkan kehandalan kinerja pengendalian antara pengendali MPC tanpa *constraint* dengan pengendali PID.

Dalam menentukan parameter pengendali P, I, dan D menggunakan Metode coba-coba (*trial and error*) sampai menghasilkan kombinasi parameter yang terbaik, yaitu:

$$K_p = 0.02, T_i = 2, T_d = 1$$

Pengujian pertama dilakukan dengan membuat variasi pada nilai *set point* dan diasumsikan beban tetap. Hasil simulasi keluaran sistem dan sinyal kendali menggunakan pengendali PID dapat dilihat pada gambar 4.17

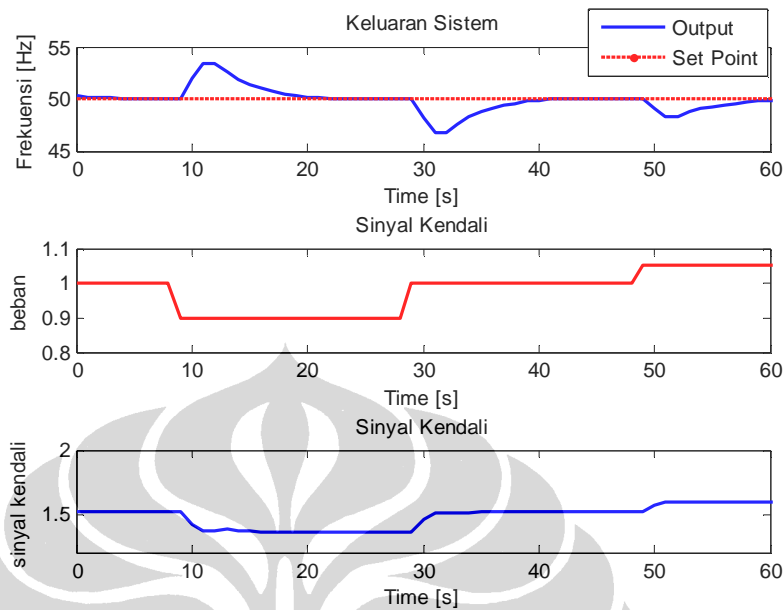


Gambar 4.17 Respon sistem dengan perubahan *set point*

Dari Gambar 4.17 dapat dilihat bahwa keluaran sistem mempunyai respon yang sangat lambat ketika terjadi perubahan *set point*. Hal ini terjadi karena pengendali PID tidak bisa memprediksi apa yang akan terjadi sehingga keluaran sistem terjadi setelah perubahan *set point*.

Pengujian kedua dilakukan dengan membuat nilai *set point* tetap yaitu 50 Hz. Perubahan beban berupa penurunan beban pada $t = 10$ detik sebesar 10 % dari

total beban dan kenaikan beban pada $t = 50$ detik sebesar 5 % dari total beban. Hasil simulasi keluaran sistem dan sinyal kendali dapat dilihat pada Gambar 4.18.

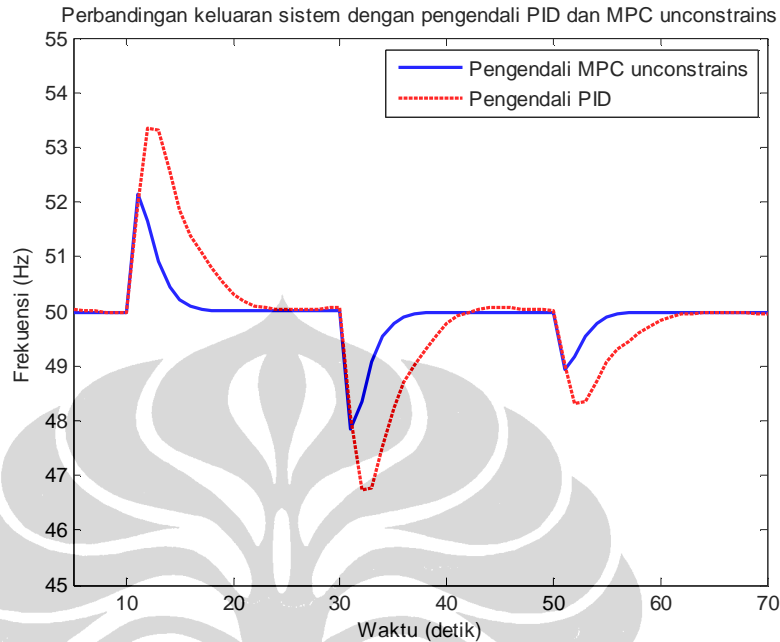


Gambar 4.18 Respon sistem dengan perubahan beban

Gambar 4.18 adalah respon keluaran sistem pada pengendali PID dengan perubahan beban. Dari Gambar tersebut dapat dilihat bahwa pada $t = 10$ detik terjadi perubahan beban berupa penurunan beban sebesar $0.1 \text{ pu} = 2.5 \text{ kW}$, maka frekuensi naik sampai 54 Hz. Karena nilai frekuensi yang dihasilkan tidak sama dengan nilai frekuensi pada *set point* maka sinyal *error* akan dikirim ke kontrol MPC dan kontrol MPC akan memberikan sinyal kendali kepada *gate* servomotor untuk memperkecil bukaan *gate*. Setelah mengalami perubahan beban, waktu yang diperlukan untuk kembali ke keadaan *steady state* atau 50 Hz adalah 10 detik.

Pada $t = 50$ detik terjadi perubahan beban berupa kenaikan beban sebesar $0.05 \text{ pu} = 1.25 \text{ kW}$, maka frekuensi akan turun menjadi 47 Hz. Karena nilai frekuensi yang dihasilkan tidak sama dengan nilai frekuensi pada *set point* maka sinyal *error* akan dikirim ke kontrol MPC dan kontrol MPC akan memberikan sinyal kendali kepada *gate* servomotor untuk memperbesar bukaan *gate*. Setelah mengalami kenaikan beban, waktu yang diperlukan untuk kembali ke keadaan *steady state* atau 50 Hz adalah 10 detik.

Perbandingan antara respon keluaran sistem dengan pengendali MPC tanpa *constraint* dengan pengendali PID dapat dilihat pada Gambar 4.19.



Gambar 4.19 Perbandingan keluaran sistem dengan pengendali PID dan MPC tanpa *constraint*

Pada Gambar 4.19 dapat dilihat perbandingan kehandalan kinerja sistem dengan pengendali PID dan pengendali MPC tanpa *constraint*. Ketika terjadi penurunan beban sebesar 10% dari total beban pada $t=10$ detik, jika menggunakan pengendali PID maka keluaran sistem lebih tinggi yaitu 53.5 Hz bila dibandingkan jika menggunakan pengendali MPC tanpa *constraint* yaitu 52 Hz. Waktu yang diperlukan untuk kembali ke keadaan steady state atau 50 Hz setelah terjadi perubahan beban jika menggunakan pengendali PID akan lebih lama yaitu 10 detik bila dibandingkan jika menggunakan pengendali MPC tanpa *constraint* yaitu 5 detik.