

Pengendalian Frekuensi dengan Menggunakan Kontrol *Fuzzy* Prediktif pada Simulator Plant Turbin – Generator pada PLTU

Dyah Putri Anggriani, Rusdhianto Effendie

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

E-mail: ditto@ee.its.ac.id

Abstrak—Sistem pembangkit dirasakan sangat perlu guna memenuhi kebutuhan tenaga listrik yang semakin meningkat, kestabilan sangat dibutuhkan pada proses pembangkit sehingga sistem pengendalian digunakan untuk menjaga variabel proses tersebut tetap stabil. Salah satunya adalah dengan melakukan pengendali frekuensi pada turbin-generator suatu pembangkit listrik, contohnya PLTU (Pembangkit Listrik Tenaga Uap). Frekuensi dari turbin uap harus dijaga kestabilannya agar keluaran daya listrik di generator berjalan dengan baik. Fluktuasi frekuensi adalah salah satu kendala penyampaian daya listrik ke beban, juga waktu kembali yang tidak segera ke kondisi normal akan mengakibatkan kerusakan pada sistem seperti patahnya poros turbin-generator dan kemungkinan terjadi gangguan pada jaringan listrik, sehingga perlu dilakukan pengaturan laju aliran uap yang masuk ke turbin. Kontroler yang digunakan untuk menjaga perubahan frekuensi adalah kontrol *fuzzy* prediktif, dengan penambahan *gain* K1 pada kontrol *fuzzy* prediktif sebesar 42.35 yang bekerja secara *successive* kontroler ini dapat mengurangi *error* sebesar 1,04% jika sistem hanya menggunakan kontroler *fuzzy* pada saat terjadi perubahan beban.

Kata Kunci—Frekuensi, Turbin, *Fuzzy* Prediktif

I. PENDAHULUAN

KEBUTUHAN akan penggunaan daya listrik saat ini sangat diperlukan mengingat banyak negara diprediksi dalam keadaan krisis energi listrik dimasa mendatang. Saat ini pemerintah merencanakan pembangunan pembangkit listrik untuk mengurangi krisis energi listrik tersebut. Oleh karena itu, pembelajaran mengenai pembangkit listrik perlu dilakukan, metode simulasi merupakan salah satu metode pembelajaran yang efektif dan efisien. Dari pembelajaran simulasi dapat diwujudkan pembuatan simulator dari sistem pembangkit listrik. Gangguan sistem tenaga listrik mungkin bisa terjadi dan menjadi permasalahan serta kendala penyaluran daya listrik ke beban. Contoh gangguannya adalah terjadinya fluktuasi frekuensi sistem tenaga listrik yang disebabkan perubahan beban.

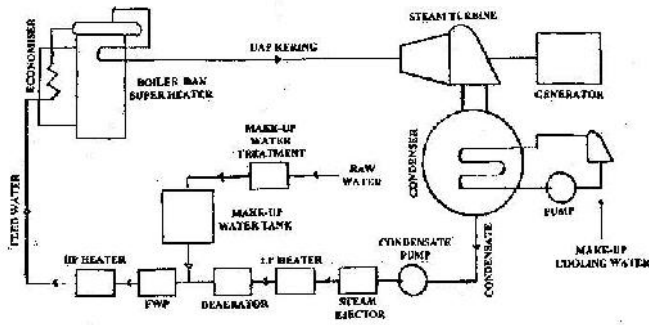
Fluktuasi frekuensi seharusnya berada pada batas toleransi yang sudah ditetapkan, dan kembali kepada frekuensi normalnya dengan segera. Fluktuasi frekuensi yang berada diatas batas toleransi merupakan sebuah permasalahan yang kerap terjadi. Selain itu, waktu kembali fluktuasi frekuensi yang tidak segera ke kondisi normal akan mengakibatkan kerusakan pada sistem seperti patahnya poros turbin-generator dan kemungkinan terjadi gangguan pada jaringan listrik. Fluktuasi frekuensi ini erat kaitannya dengan perubahan kecepatan putar pada turbin dan generator, dikarenakan perubahan permintaan daya beban. Perubahan kecepatan turbin-generator dilakukan dengan mengikuti nilai perubahan beban, melalui pengaturan laju aliran uap sebagai *input* dari pemodelan ini. sehingga akan terlihat perubahan nilai frekuensi yang dihasilkan turbin-generator. Kontrol yang digunakan untuk pengendalian pada sistem ini menggunakan kontrol *fuzzy* prediktif.

II. TEORI DASAR

A. Pembangkit Listrik Tenaga Uap (PLTU) [1]

PLTU merupakan suatu sistem pembangkit tenaga listrik yang mengkonversikan energi kimia listrik dengan menggunakan uap air sebagai fluida kerjanya, yaitu dengan memanfaatkan energi kinetik uap untuk menggerakkan poros sudu-sudu turbin. Sudu-sudu turbin menggerakkan poros turbin, untuk selanjutnya poros turbin menggerakkan generator. Dari generator inilah kemudian dibangkitkan energi listrik. Energi listrik yang dihasilkan akan mensuplai alat-alat yang disini disebut beban.

Prinsip kerja dari PLTU berdasarkan pada Gambar 1 adalah menggunakan siklus air-uap sistem tertutup, dimana air dari *condenser* atau air hasil dari proses pengkondensasian. Di sini air dipanaskan kembali kemudian dimasukkan ke *deaerator* untuk menghilangkan gas udara (oksigen), kemudian air dipompa oleh *boiler feedwater pump* ke *economizer*. Dari *economizer* yang selanjutnya dialirkan ke pipa down comer untuk dipanaskan pada *wall tubes* yang berada di *boiler*.

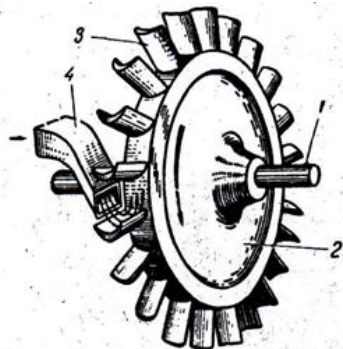


Gambar 1. Siklus sederhana PLTU

B. Turbin Uap [2]

Turbin uap adalah suatu penggerak mula yang mengubah energi potensial uap menjadi energi kinetik dan energi kinetik ini selanjutnya diubah menjadi energi mekanis dalam bentuk putaran poros turbin. Poros turbin, langsung atau dengan bantuan roda gigi reduksi, dihubungkan dengan mekanisme yang digerakkan. Tergantung pada jenis mekanisme yang digerakkan, turbin uap dapat digunakan pada berbagai bidang industri, untuk pembangkit tenaga listrik dan untuk transportasi.

Turbin uap terdiri dari beberapa bagian utama seperti: rumah turbin (*casing*), bagian yang berputar (*rotor*), sudu-sudu yang dipasang pada rotor maupun *casing* dan bantalan untuk menyanggah rotor.



Gambar 2. Konstruksi sederhana turbin

Pada Gambar 2 bagian yang ditandai dengan nomor 1 adalah poros, komponen utama tempat dipasangnya cakram-cakram sepanjang sumbu dimana poros berhubungan langsung dengan generator. Nomor 2 disebut dengan cakram, tempat sudu-sudu dipasang secara radial pada poros sedangkan sudu-sudu turbin diberi nomor 3 berfungsi sebagai menerima gaya dari energi kinetik uap melalui nosel. Nomer 4 adalah nosel yang berfungsi sebagai media ekspansi uap yang merubah energi potensial menjadi energi kinetik.

C. Logika Fuzzy [3]

Professor Zadeh (California Univ.) diakui sebagai ilmuwan pemrakarsa konsep himpunan fuzzy pada tahun 1965 yang telah menjabarkan perhitungan matematik untuk menggambarkan ketidak-jelasan atau kesamaran dalam bentuk variable linguistik. Ide klasik yang menggabungkan pendekatan kualitatif dengan kuantitatif.

Prinsip dasar dan persamaan matematika dari teori himpunan fuzzy, adalah sebuah teori pengelompokan objek dalam batas yang samar. Pertama, kita harus melihat himpunan fuzzy sebagai suatu generalisasi dari himpunan crisp klasik dengan memperluas jelajah fungsi anggota (atau fungsi karakteristik) dari [0,1] sampai semua angka real pada interval [0,1].

Fuzifikasi

Fuzifikasi merupakan proses pemetaan nilai-nilai input (*crispinput*) yang berasal dari sistem yang dikontrol (besaran non fuzzy) ke dalam himpunan fuzzy menurut fungsi keanggotaannya. Himpunan fuzzy tersebut merupakan *fuzzyinput* yang akan diolah secara fuzzy pada proses berikutnya. Untuk mengubah *crispinput* menjadi *fuzzyinput*, terlebih dahulu harus menentukan *membership function* untuk tiap *crispinput*, kemudian proses fuzifikasi akan mengambil *crispinput* dan membandingkan dengan *membership function* yang telah ada untuk menghasilkan *fuzzyinput*.

Fuzzy Rule Base [4]

Pada tahap *fuzzy rule base*, diproses hubungan antara nilai-nilai input (*crispinput*) dan nilai-nilai output (*crispoutput*) yang dikehendaki dengan aturan-aturan (*rules*). Aturan ini nantinya yang akan menentukan respon sistem terhadap berbagai kondisi *setting point* dan gangguan yang terjadi pada sistem. Aturan yang banyak digunakan adalah :
Format aturan IF-THEN

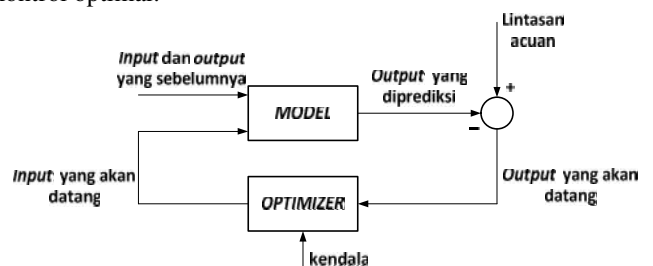
IF Err is Normally big and AEn is Normally big
THEN output is Normally big

Defuzifikasi

Pada tahap ini dilakukan pemetaan bagi nilai-nilai *fuzzyoutput* yang dihasilkan pada tahap *fuzzyrule base* ke nilai-nilai output kuantitatif yang sesuai dengan sistem yang diharapkan. Ada berbagai metode untuk melakukan proses defuzifikasi pada komposisi aturan Mamdani.

D. Kontrol Prediktif [5]

Model prediktif kontrol adalah teknik yang berfokus pada pembuatan kontroler yang dapat menyesuaikan aksi kontrol sebelum perubahan pada keluaran setpoint sebenarnya terjadi, kemampuan prediktif ketika dikombinasikan dengan operasi umpan-balik memungkinkan kontroler untuk melakukan penyesuaian lebih *smoother* dan dekat pada nilai dari kontrol optimal.



Gambar 3. Blok diagram kontrol prediktif

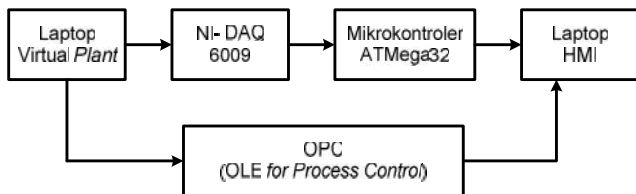
Blok diagram dari kontrol prediktif ditunjukkan oleh Gambar 3, dimana *input* dan *output* sebelumnya masuk ke

dalam blok model untuk mendapatkan *output* yang akan datang. *Output* tersebut akan di optimalisasikan, agar dapat meminimumkan *error* dan memaksimalkan sinyal kontrol dari gangguan sehingga menghasilkan *input* yang akan datang.

III. PERANCANGAN SISTEM

A. Gambaran Umum Perancangan Sistem

Bab ini membahas mengenai perancangan sistem secara detail meliputi perancangan perangkat keras yang terdiri dari perancangan sistem minimum mikrokontroler ATmega32, NI-DAQ, dan penggabungan elemen - elemen pembangun sistem, sedangkan untuk perancangan perangkat lunak antara lain perancangan simulator *plant* turbin-generator berupa *virtual plant* dan HMI yang keduanya dibangun dengan *software* LabVIEW 8.6. Untuk komunikasi pengiriman data serial menggunakan *software* Toolbox Omniserver dan KEPServerEx.

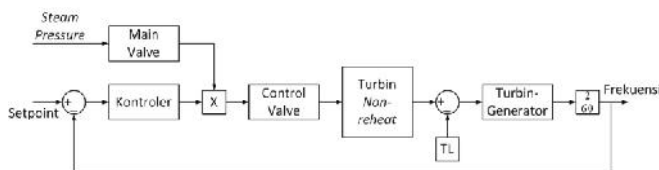


Gambar 4. Blok diagram sistem

Arsitektur yang akan dibangun ditunjukkan oleh Gambar 4, dimana laptop virtual *plant* (laptop yang dipergunakan sebagai virtual *plant*) dihubungkan ke laptop HMI (laptop yang dipergunakan sebagai HMI) melalui NI-DAQ dan mikrokontroler sedangkan *software* OPC (*OLE for Process Control*) digunakan untuk mengirimkan data tambahan dari laptop virtual *plant* ke laptop HMI secara serial.

B. Pemodelan Virtual Plant Turbin-Generator

Plant yang digunakan pada tugas akhir ini adalah turbin-generator yang digunakan untuk pembangkit listrik tenaga uap, dengan menggunakan energi solar. Putaran yang dihasilkan turbin untuk memutar generator mencapai 3000 RPM, sehingga menghasilkan frekuensi 50 Hz.



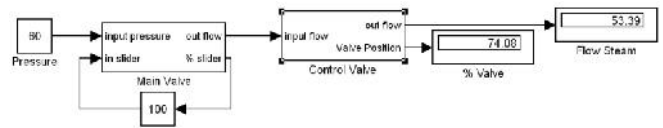
Gambar 5. Blok diagram turbin-generator

Gambar 5 menunjukkan blok diagram turbin-generator. Kontroler yang digunakan berfungsi untuk menjaga agar *output* frekuensi tetap berada dinilai 50 Hz walaupun terjadi perubahan beban pada TL (torsi lawan). Torsi lawan ini dihasilkan oleh poros generator bila terjadi perubahan pada daya beban, karena poros turbin berhubungan langsung dengan turbin.

C. Pemodelan Control Valve

Steam yang dihasilkan boiler sebelum menggerakkan turbin akan melewati 2 *valve* utama yaitu *main valve* yang

dioperasikan secara manual hanya pada saat *starting* dan *control valve*. Pada Gambar 6 terlihat *pressure* yang masuk ke *main valve* bernilai konstan yaitu 60 kg/cm².

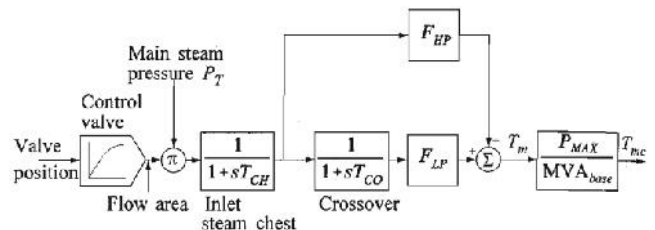


Gambar 6. Blok diagram control valve

Saat awal, laju aliran *steam* akan dikendalikan oleh *manual valve* dan akan terbuka 100% hingga kecepatan turbin mencapai 3000 RPM, selanjutnya akan dikendalikan sepenuhnya oleh *control valve*. Bukan dari *control valve* sebanding dengan kecepatan turbin yaitu sekitar 74% untuk putaran 3000 RPM.

D. Pemodelan Turbin

Plant yang digunakan pada tugas akhir ini adalah dalam bentuk *virtual*, yaitu *virtual plant* turbin-generator yang diadaptasi dari buku Prabha Kundur dan ditunjukkan oleh Gambar 7.



Gambar 7. Blok diagram turbin

Turbin yang digunakan adalah jenis *non-reheat* dimana uap yang masuk ke turbin tidak mengalami pemanasan ulang untuk menggerakkan turbin itu sendiri. Hanya saja uap yang masih panas akan dimanfaatkan sebagai pemanas (*preheating*) air yang akan diuapkan di dalam *boiler*.

E. Pemodelan Mekanika Generator

Bentuk pemodelan diadaptasi dari buku Prabha Kundur dengan bentuk *transfer function*.

$$T_f(s) = \frac{1}{(Js + B)} = \frac{0.294}{2.94s + 1} = \frac{1}{10s + 3.4}$$

Dimana J adalah inersia dan B adalah gesekan (*damper*) dari turbin-generator.

F. Perancangan Komunikasi Data

Pengiriman data dari laptop virtual *plant* ke HMI menggunakan NI-DAQ yang sebelumnya dikonversi dahulu menjadi 0-5 volt karena kemampuan NI-DAQ membaca tegangan pada range tersebut, kemudian dikirimkan ke mikrokontroler dan diolah yang selanjutnya akan dikirimkan ke HMI melalui kabel serial.

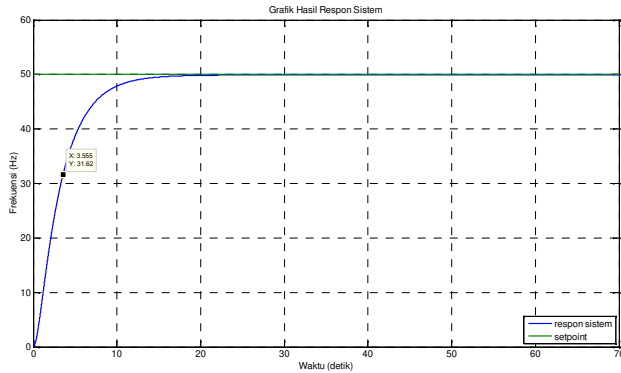
G. Identifikasi Sistem

Dalam merancang suatu sistem yang mampu dikendalikan, maka terlebih dahulu perlu dilakukan proses identifikasi terhadap *plant*, sehingga dapat diketahui karakteristik dari *plant*.

Proses identifikasi dilakukan dengan menggunakan metode loop terbuka dengan bebannominal, maksimal dan

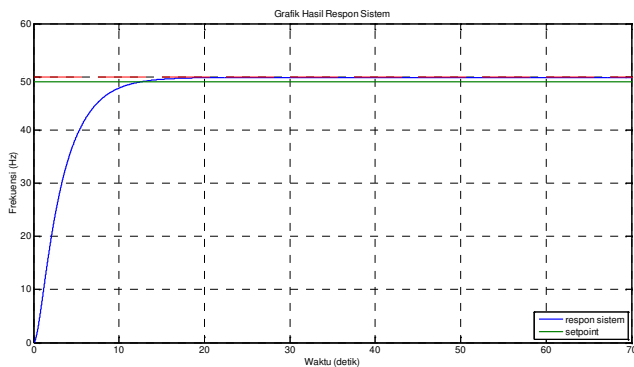
minimal secara simulasi dengan bantuan software MATLAB, dengan fungsi alih *plant* telah didapatkan sebelumnya.

Grafik hasil respon dari pemberian sinyal *step* pada *plant* untuk frekuensi saat beban nominal ditunjukkan pada Gambar 8, dimana respon mencapai nilai *setpoint* 50 Hz dengan $\tau = 3,5$ detik.



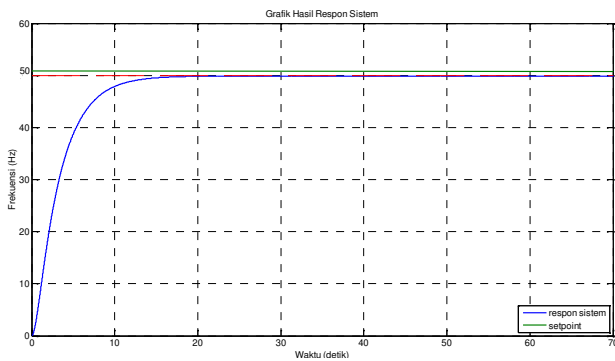
Gambar 8. Respon *output* frekuensi saat beban nominal

Gambar 9 menunjukkan grafik respon *output* frekuensi saat beban minimal. Saat beban tersebut respon melebihi *setpoint* yang diinginkan yaitu bernilai 53 Hz dengan $\tau = 3,8$ detik.



Gambar 9. Respon *output* frekuensi saat beban minimal

Gambar 10 adalah saat sistem diberi beban maksimal, dimana respon dari output hanya mencapai nilai 48.5 Hz dari 50 Hz dengan $\tau = 3,4$ detik.



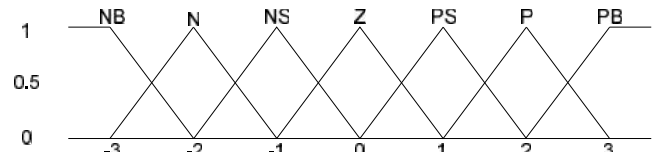
Gambar 10. Respon *output* frekuensi saat beban maksimal

H. Perancangan Kontrol Fuzzy

Untuk merancang suatu kontrol fuzzy, maka dilakukan tahap seperti :

1. Fuzifikasi

Jenis fungsi keanggotaan yang digunakan ada dua jenis, yaitu segitiga dan trapesium. Fungsi trapezium dipilih untuk mendefinisikan derajat keanggotaan yang sama dengan satu pada banyak nilai variabelnya, sedangkan fungsi segitiga dipilih karena memiliki nilai yang terbatas dan derajat keanggotaannya bernilai satu pada satu titik tertentu. Besar linguistik term input *error* (e) dan delta *error* (e) bernilai sama. Ilustrasi dari fungsi keanggotaan yang digunakan pada kontroler ini ditunjukkan pada Gambar 11.



Gambar 11. Membership function *error* (e) dan delta *error* (e).

2. Fuzzy Rule Base

Aturan dasar dirancang berdasarkan teori bukaan *control valve* yang tepat saat sistem dalam kondisi berbeban. Pada kondisi beban nominal maka laju uap yang dialirkan oleh *control valve* sebesar $\pm 53,3$ Kg/s, sedangkan bila sistem diberi beban maksimal atau minimal maka *control valve* harus dapat menyesuaikan laju uap agar sistem tidak mengalami gangguan. Dengan mengacu teori tersebut maka dirancang sebuah *rule base* yang merepresentasikan perubahan dari bukaan *control valve* seperti pada Tabel 1

Tabel 1. Rule base untuk kontrol fuzzy

		delta error						
		NB	N	NS	Z	PS	P	PB
error	NB	NB	NB	N	N	NS	NS	Z
	N	NB	N	N	NS	NS	Z	PS
	NS	N	N	NS	NS	Z	PS	PS
	Z	N	NS	NS	Z	PS	PS	P
	PS	NS	NS	Z	PS	PS	P	P
	P	NS	Z	PS	PS	P	P	PB
	PB	Z	PS	PS	P	P	PB	PB

3. Defuzifikasi

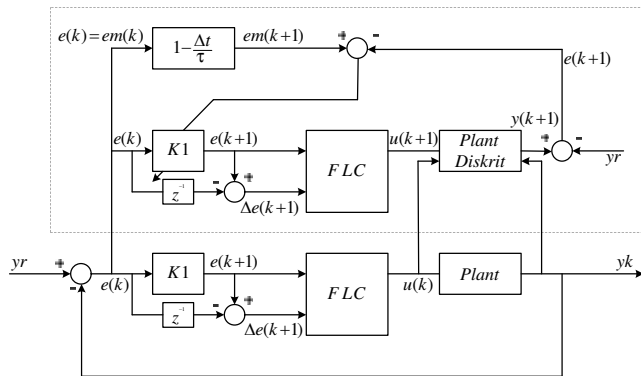
Proses defuzifikasi digunakan untuk mendapatkan nilai keluaran, dari hasil proses *rule base* didapatkan fungsi keanggotaan yang menjadi rentang nilai output kontroler. Metode defuzifikasi pada kontroler ini menggunakan jenis MOM (*Maximum of Minimum*), maka nilai output dapat dihitung dengan Persamaan 3.1

$$z = \sum_{j=1}^l \frac{v_j}{j} \dots\dots\dots (2)$$

- dengan,
- z = nilai keluaran
- v_j = nilai masukan
- j = banyaknya nilai maksimum

I. Perancangan Kontrol Prediktif

Kontrol prediktif yang dibangun pada sistem ini memprediksikan satu langkah ke depan dimana blok diagram untuk kontrol fuzzy prediktif ditunjukkan oleh Gambar 12, kontroler tersebut memprediksikan nilai dari output $y(k+1)$. Nilai *error* yang masuk ke FLC (*Fuzzy Logic Controller*) diberikan variabel tambahan yaitu $K1$ yang nilainya berubah-ubah sesuai dengan perubahan yang terjadi pada output dari *plant* diskrit. Hal ini dikarenakan nilai $e(k)$ harus sama dengan $em(k)$.



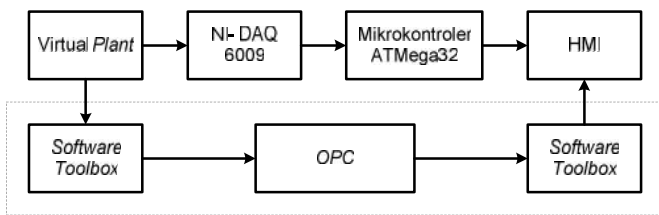
Gambar 12. Blok diagram kontrol prediktif

IV. HASIL SIMULASI DAN IMPLEMENTASI

Bab ini menjelaskan mengenai proses implementasi kontroler terhadap sistem perancangan dan *plant* dengan cara mengintegrasikan sistem, kemudian dianalisa dari beberapa hasil yang diperoleh.

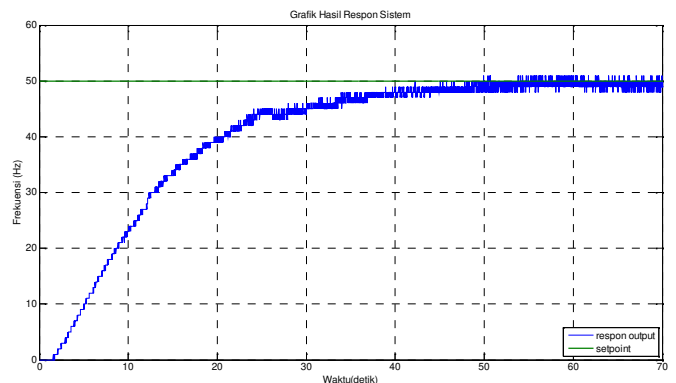
A. Implementasi Komunikasi NI-DAQ – Mikrokontroler

Pada tahap ini, implementasi dilakukan dengan mengintegrasikan antara *virtual plant* dengan HMI dengan menggunakan NI-DAQ yang terhubung oleh mikrokontroler ATmega32 pada dua komputer secara terpisah yang nantinya menggunakan media komunikasi *serial*. Gambar integrasi sistem tampak pada Gambar 13



Gambar 13. Arsitektur Sistem

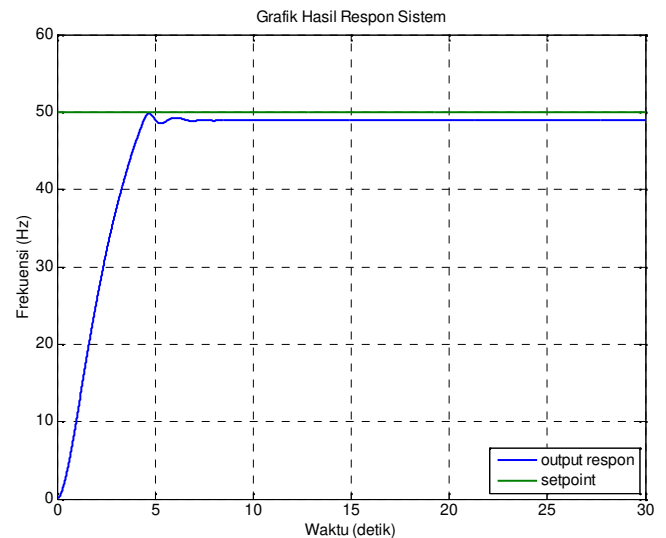
Hasil respon frekuensi yang dikirimkan virtual plant ke HMI dapat dilihat pada Gambar 14. Dimana respon memiliki banyak *ripple*, hal ini disebabkan oleh *clock* yang dimiliki NI-DAQ dan mikrokontroler berbeda. *Ripple* mungkin juga disebabkan karena nilai output dari virtual plant dikonversi ke dalam bentuk tegangan 0-5 volt, kemudian nilai yang telah dikonversi tersebut akan dikonversi kembali dalam frekuensi.



Gambar 14. Respon output frekuensi di HMI pada beban nominal

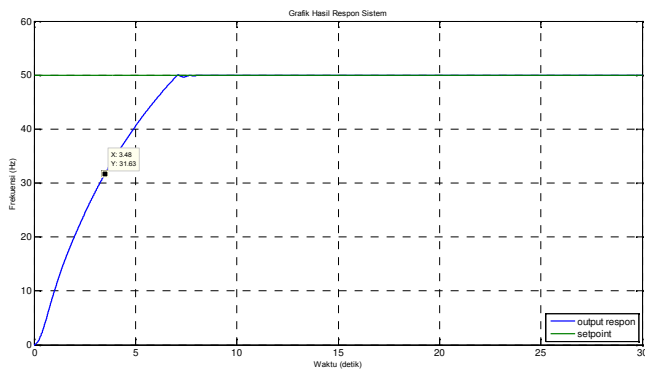
B. Hasil Perancangan Kontroler Pada Virtual Plant

Respon *output* sistem saat dalam keadaan beban minimum, nominal maupun maksimum telah dibahas pada bab sebelumnya, dimana saat terjadi pembebanan pada generator maka frekuensi yang dihasilkan oleh turbin ikut mengalami perubahan. Pengujian kontroler dilakukan dua kali untuk melihat perbandingan saat sistem dengan kontroler *fuzzy* dan kontroler *fuzzy* prediktif dengan penambahan *gain* secara *successive*. Pada Gambar 15 adalah saat sistem hanya menggunakan kontrol *fuzzy*, saat terjadi perubahan beban baik saat beban maksimal maupun minimal. Output frekuensi dari sistem tidak dapat kembali ke keadaan semula dan terdapat nilai error sebesar 1.04%, hal ini disebabkan terjadi perubahan parameter dari *plant*. Oleh karena itu, dilakukan pengujian pada sistem dengan menggunakan kontrol *fuzzy* prediktif.

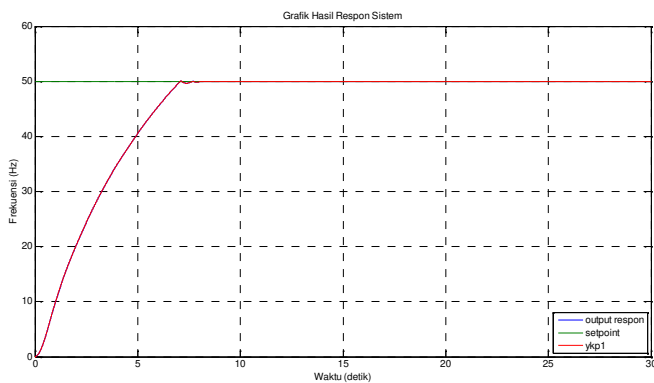


Gambar 15 Respon *output* frekuensi dengan kontrol *fuzzy*

Pada pengujian *plant* dengan kontroler *fuzzy* prediktif, sistem selalu dapat kembali ke keadaan semula, respon *output* $y(k+1)$ dapat dilihat pada Gambar 16 dimana nilai τ yang dimiliki oleh respon ini adalah 3.5 detik. Pada kontroler *fuzzy* prediktif diberi *gain* $K1$ yang nilainya akan selalu berubah mengikuti perubahan beban. *Gain* $K1$ akan berkurang bila nilai $e(k+1) < e_m(k+1)$, sebaliknya bila $e(k+1) > e_m(k+1)$ maka *gain* $K1$ akan bertambah agar nilai $e(k+1) = e_m(k+1)$

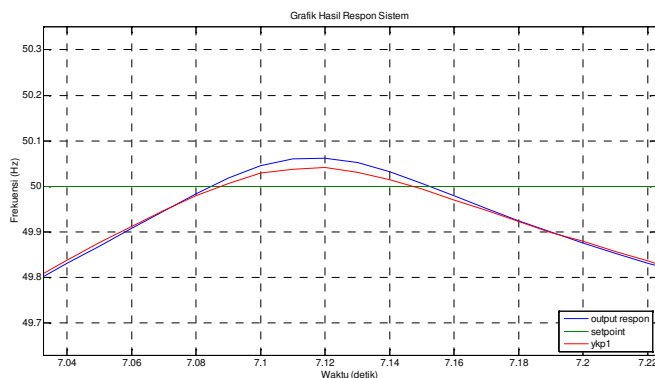


Gambar 16 Respon *output* frekuensi $y(k+1)$



Gambar 17 Penggabungan respon *output* frekuensi

Gambar 17 adalah respon frekuensi $y(k+1)$ saat digabung dengan respon frekuensi $y(k)$, terlihat tidak terlalu banyak perbedaan pada respon. Namun, apabila gambar diperbesar maka akan terlihat bahwa kontroler prediktif bekerja pada sistem ini. Hal tersebut diperlihatkan oleh Gambar 18, untuk garis yang berwarna biru adalah respon *output* sistem dan garis berwarna merah adalah respon *output* frekuensi $y(k+1)$ dimana garis tersebut selalu memprediksi satu langkah kedepan yang dihasilkan oleh *output* frekuensi $y(k)$.



Gambar 18. Perbedaan respon frekuensi $y(k+1)$ dengan $y(k)$

V. KESIMPULAN

Berdasarkan hasil percobaan yang telah dilakukan pada pengerjaan tugas akhir ini, maka dapat diambil beberapa kesimpulan antara lain:

1. *Simulator plant* turbin-generator dapat bekerja dengan baik walaupun dalam keadaan berbeban, respon selalu mencapai nilai *set point* yang diinginkan yaitu 50 Hz.
2. Penambahan *gain* K1 sebesar 42.35 yang bekerja secara *successive* pada kontroler *fuzzy* prediktif dapat memperbaiki error pada kontroler *fuzzy*
3. Jika sistem hanya menggunakan kontrol *fuzzy* sebagai pengendali frekuensi saat terjadi perubahan beban, maka *error* yang dapat direduksi oleh kontroler *fuzzy* prediktif adalah sebesar 1.04 %.

Diharapkan untuk tugas akhir yang akan datang menggunakan *plant* yang nyata tidak dalam bentuk virtual, agar dapat dilakukan identifikasi dan mendapatkan nilai parameter yang lebih baik.

DAFTAR PUSTAKA

- [1] Pudjanarsa, Astu, "Mesin Konversi Energi", Penerbit ANDI, Yogyakarta, 2008.
- [2] Abduh, Syamsir, "Teknik Tegangan Tinggi", Penerbit Salemba Teknika, Jakarta, 2001.
- [3] Robandi, I., "Desain Sistem Tenaga Modern; Optimisasi, Logika Fuzzy, dan Algoritma Genetika", Penerbit ANDI, Yogyakarta, 2006.
- [4] Pasino, K. M., dan Yurkovich, S. , "Fuzzy Control", Addison Wesley Longman, California, 1998.
- [5] S. Ronald, "Predictive Control", Prentice Hall, Department of Electrical Engineering, Netherlands, 1991.
- [6] Ogata, Katsuhiko, "Teknik Kontrol Automatik"(Sistem Pengaturan), Penerbit Erlangga, Bandung 1970.