
Sistem kendali Anti-windup Dengan Kontroler PID Pada Sistem Yang Mempunyai Dead Time

*Alfano Yoga Pangestu**¹, *Suyanta*², *Joko Prasajo*³
Program Studi Teknik Elektro, Fakultas Teknik dan Perancangan
Institut Teknologi Nasional Yogyakarta
e-mail : alfanoyy21@gmail.com

Abstrak

Kontroler PID (Proportional-Integral-Derivative) merupakan salah satu metode pengendalian yang paling banyak digunakan dalam industri karena kesederhanaannya serta kemampuannya dalam menangani berbagai jenis proses. Namun, dalam implementasinya, kontroler PID memiliki beberapa kelemahan, terutama saat berhadapan dengan keterbatasan aktuator seperti saturasi, yang dapat menyebabkan fenomena integral windup. Kondisi ini dapat menimbulkan respon sistem yang tidak diinginkan, seperti overshoot yang besar dan waktu penetapan yang lama, sehingga mengganggu kestabilan sistem. Penelitian ini bertujuan untuk menganalisis kinerja kontroler PID yang dilengkapi dengan mekanisme anti-windup pada sistem orde satu dengan waktu tunda (dead time). Metode yang digunakan adalah simulasi berbasis perangkat lunak MATLAB/Simulink. Proses penelitian mencakup pemodelan sistem, perancangan kontroler PID dengan anti-windup, serta analisis performa kinerja respon sistem. Hasil simulasi menunjukkan bahwa penerapan anti-windup secara signifikan mengurangi efek windup pada kontroler PID. Hal ini ditunjukkan dengan peningkatan kinerja sistem berupa waktu penetapan yang lebih cepat, pengurangan overshoot, dan kestabilan respon yang lebih baik dibandingkan sistem tanpa anti-windup.

Kata kunci : PID, Anti-windup, Dead time, Sistem orde satu, MATLAB/Simulink

Abstract

Controler PID (Proportional-Integral-Derivative) controller is one of the most widely used control methods in industry due to its simplicity and ability to handle various types of processes. However, in its implementation, the PID controller has several weaknesses, especially when dealing with actuator limitations such as saturation, which can cause integral windup phenomena. This condition can cause undesirable system responses, such as large overshoots and long settling times, thus disrupting system stability. This study aims to analyze the performance of a PID controller equipped with an anti-windup mechanism in a first-order system with a dead time. The method used is a simulation based on MATLAB/Simulink software. The research process includes system modeling, designing a PID controller with anti-windup, and analyzing the system's response performance. The simulation results show that the application of anti-windup significantly reduces the windup effect on the PID controller. This is indicated by improved system performance in the form of faster settling times, reduced overshoot, and better response stability compared to a system without anti-windup.

Keywords : PID, Anti-windup, Dead time, First order system, MATLAB/Simulink

1. PENDAHULUAN

Kontroler PID telah banyak digunakan untuk kendali industri pada setiap aplikasi proses pengendalian sejak lama. Menurut survei pada tahun 1989, 90% dari proses di industri menggunakan kontroler PID konvensional (Astrom dan Hagglun, 1995). Alasan mengapa Penggunaan kontroler PID di industri tersebar luas karena kesederhanaan dan kemudahan mengembalikan fitur daring kontroler PID (Astrom dan Hagglun, 2001). Meskipun banyak aspek dari sistem kendali dapat dipahami berdasarkan teori linier, beberapa efek taklinier harus diperhitungkan. Semua aktuator memiliki keterbatasan, misalnya motor memiliki kecepatan terbatas, katup tidak boleh lebih dari terbuka penuh, air pompa memiliki laju aliran maksimum dan sebagainya. Dalam beberapa kasus, seperti kendali dengan berbagai kondisi kerja, mungkin terjadi bahwa isyarat kendali dari kontroler mencapai batas aktuator. Ketika kondisi ini terjadi, kalang umpan-balik rusak karena aktuator akan tetap pada batasnya secara bebas dari keluaran proses. Kontroler PID yang memiliki proses pengintegrasian akan terus melakukan proses integrasi dan nilai tetapan integralnya mungkin menjadi sangat besar. Konsekuensinya adalah kemunculan lonjakan respon secara paksa dalam rentang waktu yang panjang dan hal tersebut memperburuk waktu penetapan respon dan kestabilan sistem.

Salah satu kelemahan kontroler PID adalah bahwa hal itu sangat tergantung pada parameter dari objek yang dikendalikan sedangkan kontrol proses industri sistem terdiri dari banyak kendala seperti taklinier, inersia, *lag*, *dead time* atau waktu tunda, parameter bervariasi terhadap dan banyak faktor lainnya.

Sistem yang memiliki *dead-time* dikategorikan sebagai sistem tak-linier. Pada batas-batas tertentu, kontroler PID dapat memperbaiki respon menjadi lebih baik. Proses akumulasi komponen integral yang dimiliki oleh kontroler PID dapat menimbulkan peristiwa *windup*, yaitu akumulasi kontroler integral di luar batas tanpa menghiraukan daerah saturasi elemen penggerak. *Windup* berdampak pada munculnya lonjakan respon secara paksa dalam rentang waktu yang panjang, hal tersebut memperburuk waktu penetapan respon dan kestabilan sistem.

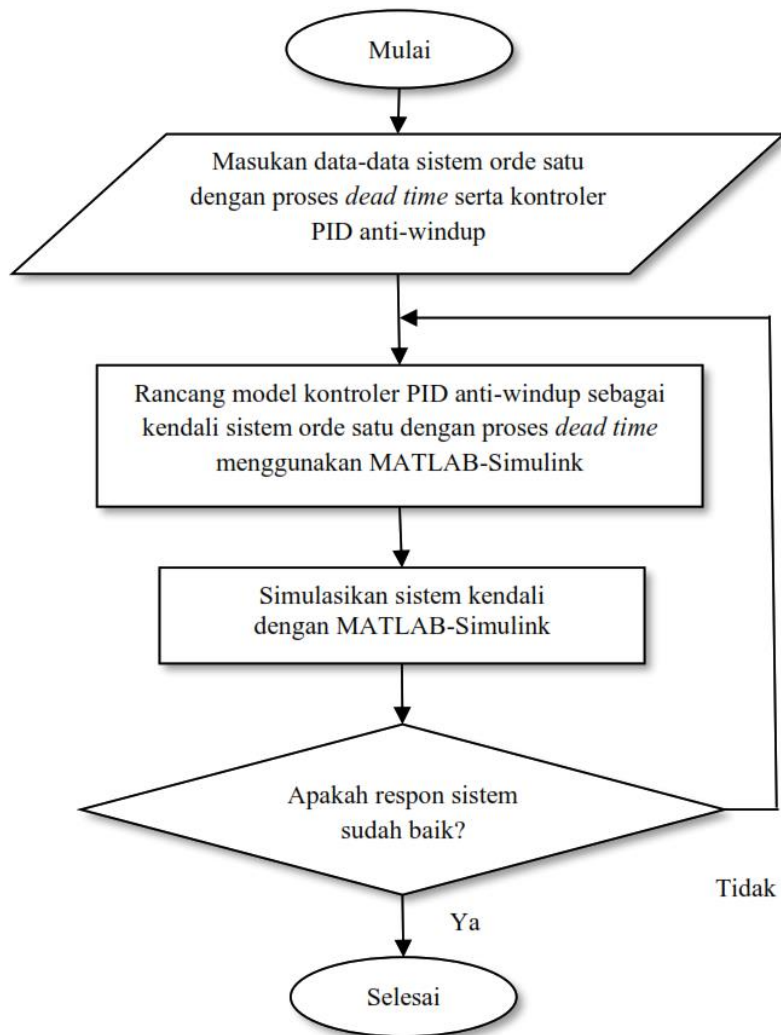
Penelitian ini akan melakukan simulasi sistem kontroler PID yang menerapkan *anti-windup* pada sistem orde satu yang mempunyai *dead-time* dengan perangkat lunak bantu yang digunakan adalah MATLAB/Simulink.

2. METODE PENELITIAN

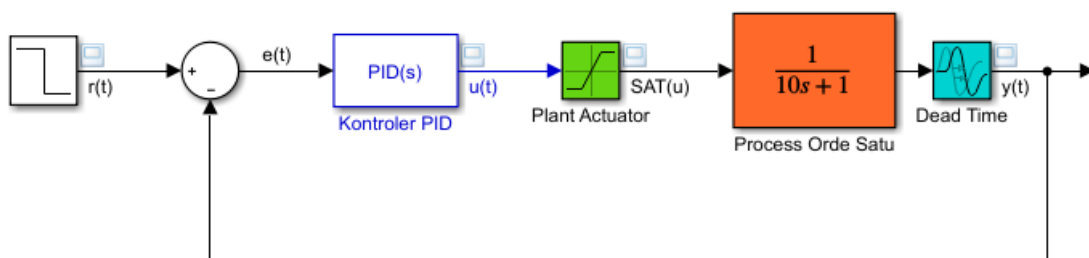
Metode penelitian yang akan digunakan pada penelitian skripsi ini bersifat eksperimental. Eksperimen penelitian ini berupa simulasi sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* dengan menggunakan perangkat lunak Matlab/Simulink. Penelitian eksperimental yaitu proses perancangan sistem penalaan, studi literatur dengan pengumpulan pustaka dari penelitian terdahulu, dan simulasi penelitian sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* dilakukan di Laboratorium Industri yang terletak di Institut Teknologi Nasional Yogyakarta.

Alat penelitian penalaan sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* menggunakan perangkat keras laptop HP Pavilion 14-ec0xxx dengan spesifikasi AMD Ryzen 5 with Radeon Graphics Processor (2.10 Ghz), RAM 16,00 GB, Internal Memory SSD 512 GB, Sistem operasi x64-bit, Windows 11 Home Single Language dan perangkat lunak matlab/simulink R2020a.

Tahapan penelitian ini memiliki beberapa tahapan, terlihat pada gambar 2.1. Tahapan pertama yaitu memasukan data-data sistem orde satu dengan proses *dead time* serta kontroler PID *anti-windup*. Tahapan Selanjutnya merancang model kontroler PID *anti-windup* sebagai kendali sistem orde satu dengan proses *dead time* menggunakan perangkat lunak MATLAB/Simulink. Kemudian mensimulasikan sistem kendali dengan MATLAB/Simulink. dan menganalisis perbedaan performa kinerja sistem kendali dengan *anti-windup* dan performa kinerja sistem kendali tanpa *anti-windup*.



Gambar 2.1 Tahapan penelitian



Gambar 2.2 Model sistem kontroler PID *anti-windup* dengan proses orde satu dan *dead time*

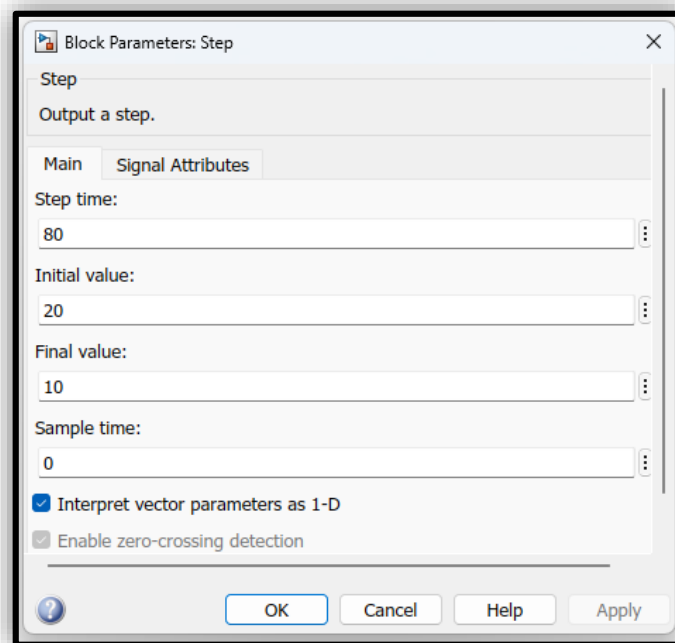
Pemodelan sistem orde satu dengan proses *dead time* diperlukan sebagai langkah awal perancangan sistem kontroler PID *anti-windup*. Sedangkan model sistem kontroler PID *anti-windup* yang diaplikasikan pada sistem orde satu dengan proses *dead time* dengan bantuan perangkat lunak MATLAB/Simulink terlihat pada Gambar 2.2.

3. HASIL DAN PEMBAHASAN

Setelah selesai melakukan seluruh perancangan model sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* yang sudah diuraikan pada metode penelitian, maka pada BAB ini akan membahas mengenai hasil dari perancangan model dan simulasinya. Pengujian simulasi dari penelitian ini dilakukan menggunakan perangkat lunak MATLAB/Simulink R2020a untuk menjalankan simulasinya.

Simulasi ini dilakukan dengan dua kali pengujian, yaitu pengujian simulasi sistem kendali tanpa menggunakan *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* dan pengujian simulasi sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time*. Hasil dari simulasi pengujian penelitian ini adalah berupa grafik yang dapat diamati melalui *scope* yang ada didalam program MATLAB/Simulink. Berikut ini adalah uraian hasil pengujian dari model simulasinya.

Masuk pada pengaturan pertama yaitu memasukan nilai setpoint pada blok parameter *step* yaitu step time 80, nilai setpoint 10 dan 20 yang terlihat pada gambar 3.1.

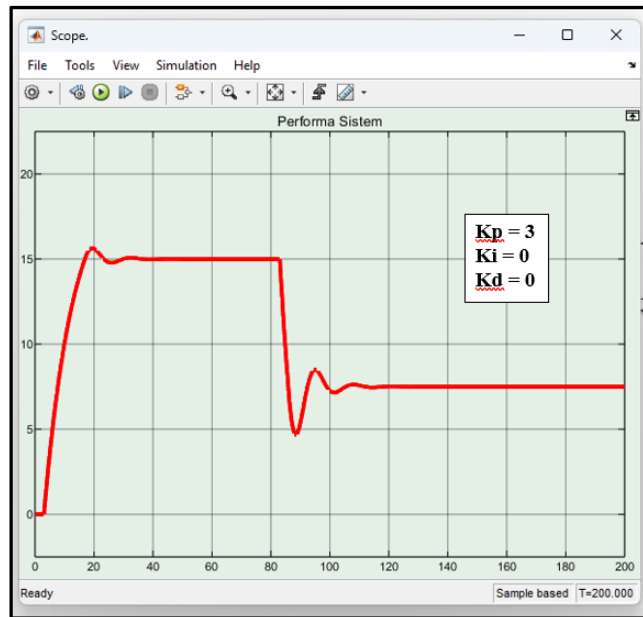


Gambar 3.1. masukan nilai step

Untuk menentukan parameter nilai konstanta kontroler PID, percobaan pertama dilakukan *tuning* secara manual *trial and error*, proses percobaan ini dilakukan dengan metode mencoba-coba (eksperimental) dengan cara mengubah masukan nilai konstanta proporsional (K_p), nilai konstanta integral (k_i), dan nilai konstanta derivatif (K_d) pada kontroler PID hingga menemukan nilai yang stabil.

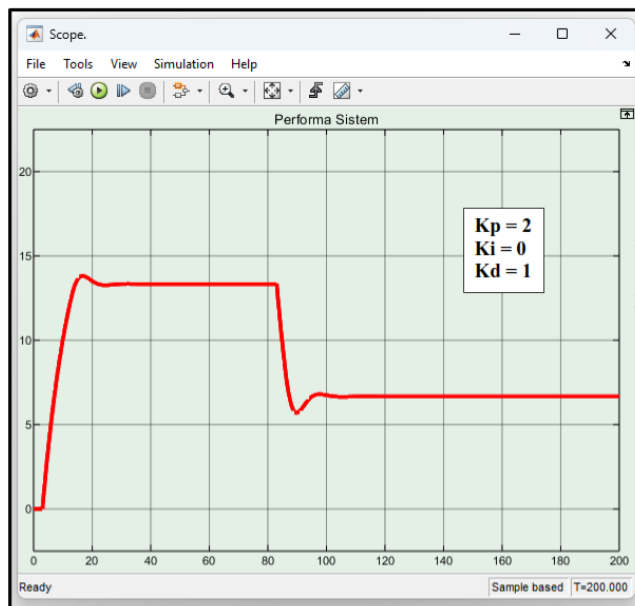
1. Langkah awal yaitu mencoba mengabaikan nilai konstanta intergral dan derivative dengan memberikan nilai nol. dan menggunakan nilai konstanta proporsional terlebih dahulu secara bertahap sampai sistem mulai beresilasi atau menunjukkan respon yang diinginkan tetapi masih belum stabil.

Gambar 3.2 menunjukkan sinyal keluaran kontroler PID ketika diberi nilai $K_p = 3$; $K_i = 0$; $K_d = 0$, sistem masih belum stabil, masih terjadi osilasi, dan belum mencapai *setpoint* yang diinginkan.



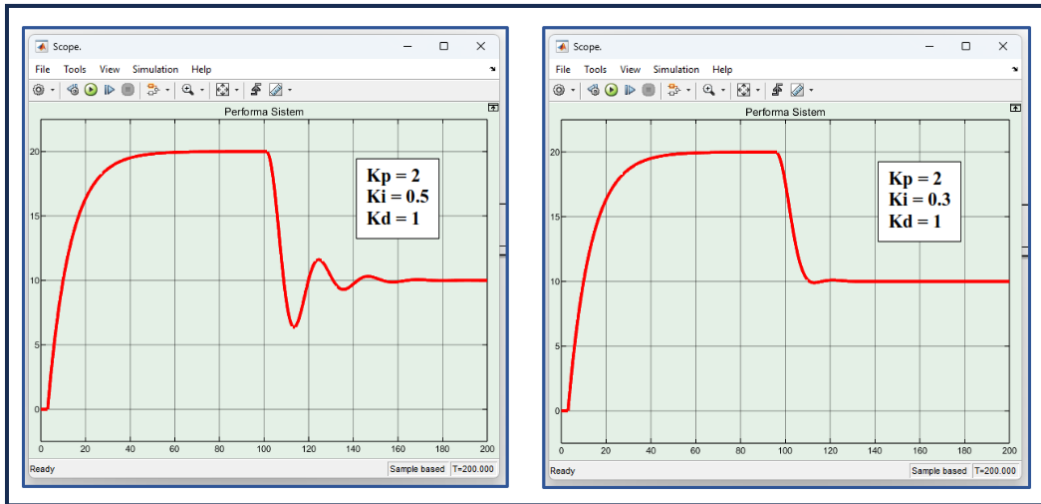
Gambar 3.2 Sinyal kontroler PID dengan nilai $K_p = 3$ $K_i = 0$ $K_d = 0$

- Setelah mendapatkan nilai konstanta proporsional yang cenderung masih belum stabil atau osilasi, selanjutnya menambahkan nilai konstanta derivatif (K_d) untuk meredam osilasi. Gambar 3.3 menunjukkan performa sistem kontroler PID dengan nilai konstanta $K_p = 2$; $K_i = 0$; $K_d = 1$ terlihat sistem masih belum stabil dan masih belum bisa mencapai *setpoint* tetapi ada nya penurunan osilasi.



Gambar 3.3 Sinyal kontroler PID dengan nilai $K_p = 2$ $K_i = 0$ $K_d = 1$

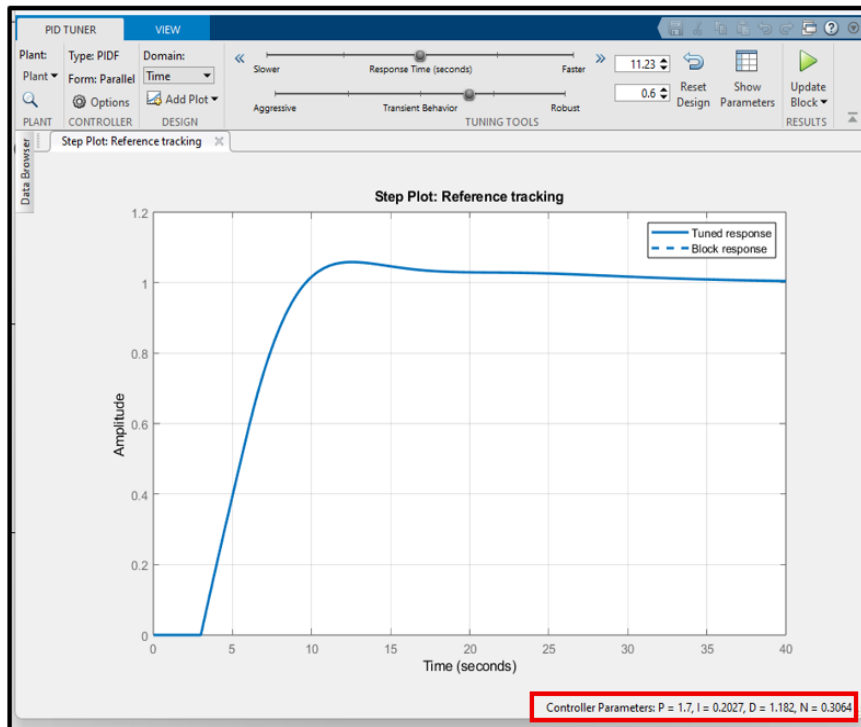
- Jika sistem sudah mulai stabil dan memiliki respon yang baik dengan konstanta proporsional (K_p) dan konstanta derivatif (K_d). Nilai konstanta Integral (K_i) dapat ditambahkan untuk menghilangkan *steady-state error* (kesalahan kondisi tunak).



Gambar 3.4 Perbandingan sinyal kontroler PID dengan nilai $K_p = 2$ $K_i = 0.5$ $K_d = 1$ dan $K_p = 2$ $K_i = 0.3$ $K_d = 1$

Gambar 3.4 menunjukkan Ketika nilai Parameter kontroler PID $K_d = 2$; $K_i = 0.5$; $K_d = 1$ sistem masih mengalami osilasi yang cukup lama setelah mencapai *setpoint* 10 berbanding dengan nilai $K_d = 2$; $K_i = 0.3$; $K_d = 1$ grafik sinyal menunjukkan sistem beroperasi lebih stabil. menjelaskan bahwa nilai konstanta integral dapat menghilangkan *steady-state error* dan meredamkan osilasi supaya sistem bisa mencapai *setpoint lebih cepat*.

Agar penelitian lebih optimal maka kontroler PID sudah di *tuning* dengan menggunakan *autotuning* PID dari MATLAB/*Simulink* dan menghasilkan parameter nilai nilai konstanta proporsional ($K_p = 1.7$) konstanta integral ($K_i = 0,2027$) dan konstanta derivatif ($K_d = 1,182$) yang dapat dilihat pada gambar 3.5.



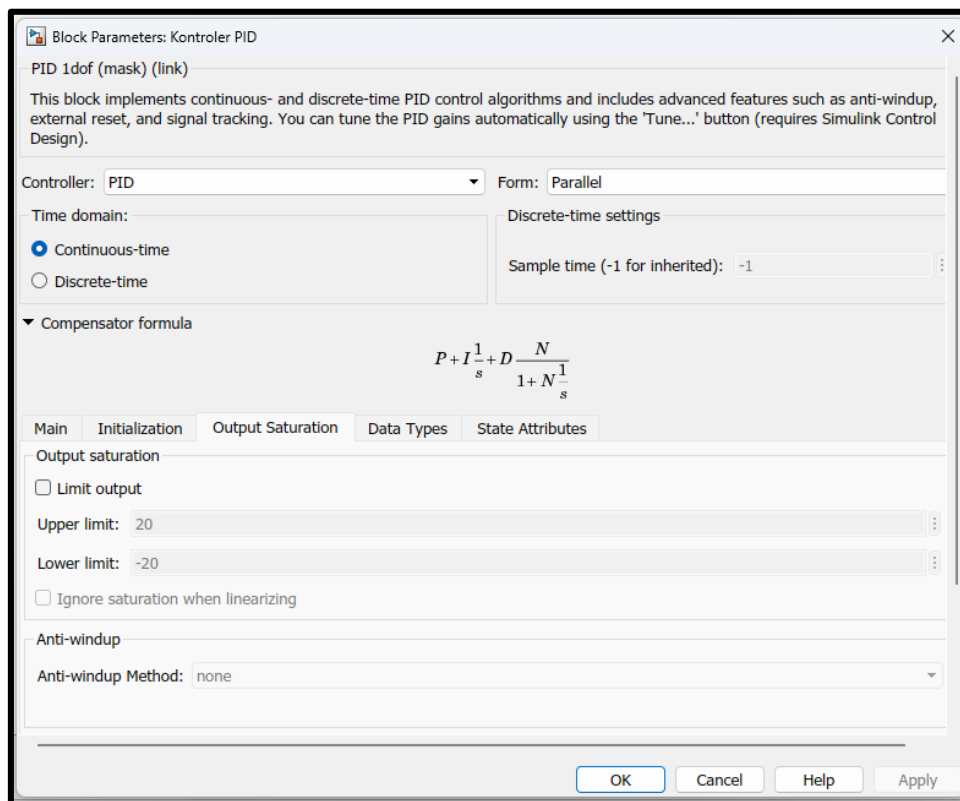
Gambar 3.5 PID tuner MATLAB/*Simulink*

Batas saturasi yang berlabel “Plant Actuator” pada blok diagram telah disetel dengan batas saturasi masukannya [20, -20]. Blok kontroler PID pada MATLAB/Simulink memiliki dua metode *anti-windup* bawaan yang memungkinkan blok kontroler PID memperhitungkan tentang masukan batas saturasiya. Penelitian Sistem kendali *anti-windup* pada kontroler PID untuk mengendalikan proses orde satu dengan waktu mati/tunda (*dead time*) 3 *second* dengan persamaan :

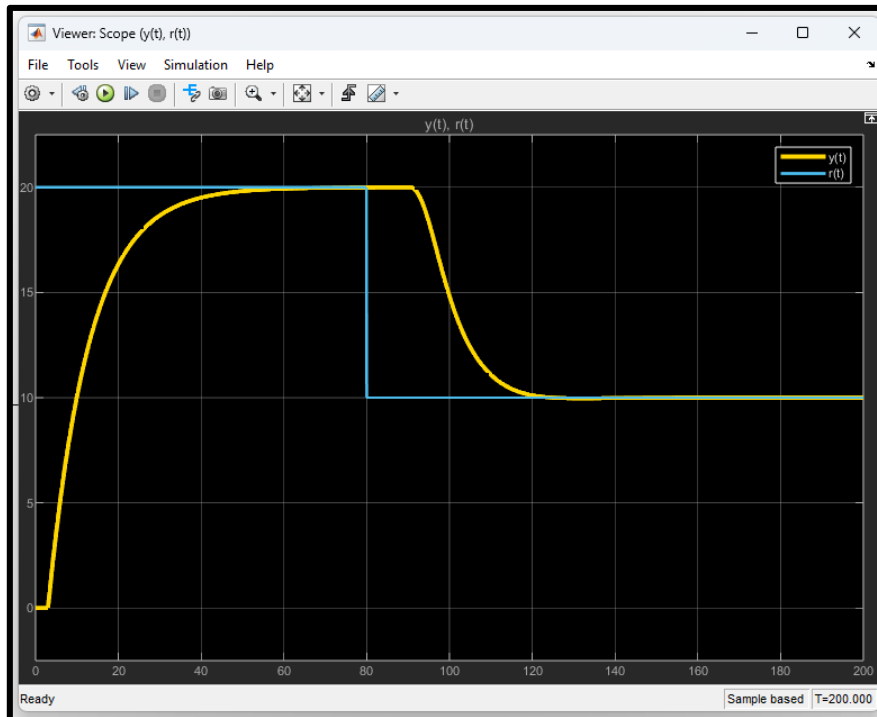
$$P(s) = \frac{1}{10s + 1} e^{-3s}$$

3.1 Kinerja Performa Sistem Tanpa Menggunakan Anti-Windup

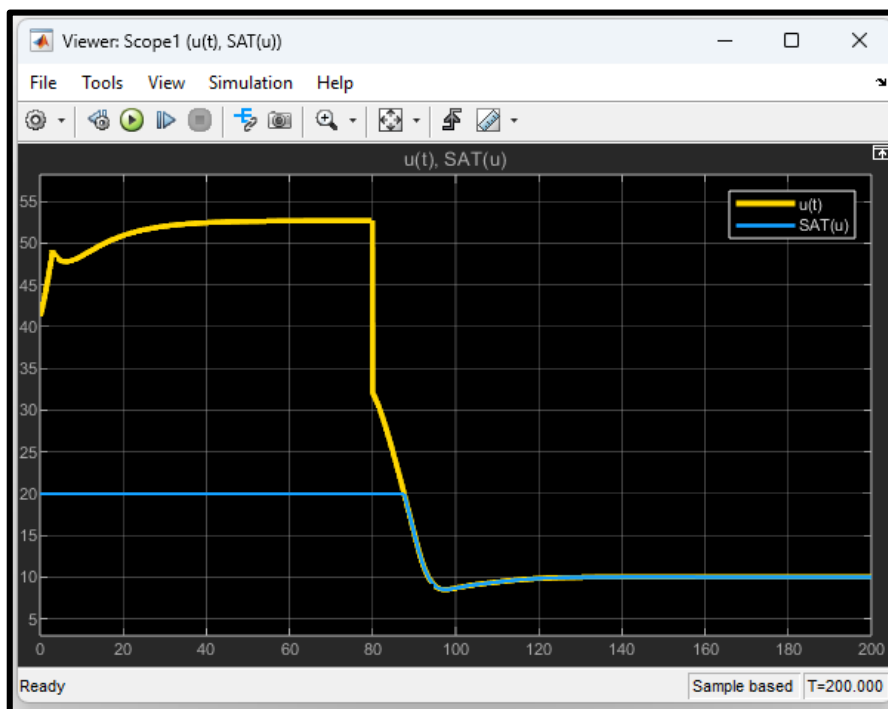
Pengujian pertama adalah menguji pengaruh saturasi ketika model saturasi tidak di pertimbangkan oleh kontroler PID dan tanpa mengaktifkan metode *anti-windup* seperti terlihat pada gambar 3.6. Simulasi sistem kendali tanpa menggunakan *anti-windup* dengan kontroler PID pada sistem proses orde satu yang mempunyai *dead time* menghasilkan hasil yang ditunjukkan pada grafik gambar 3.7 dan gambar 3.8.



Gambar 3.6 Batas saturasi blok kontroler PID dan *anti-windup* belum diaktifkan



Gambar 3.7 *Setpoint* dan *output* kontroler tanpa menggunakan *anti-windup*



Gambar 3.8 *Output* kontroler pada batas saturasi tanpa menggunakan *anti-windup*

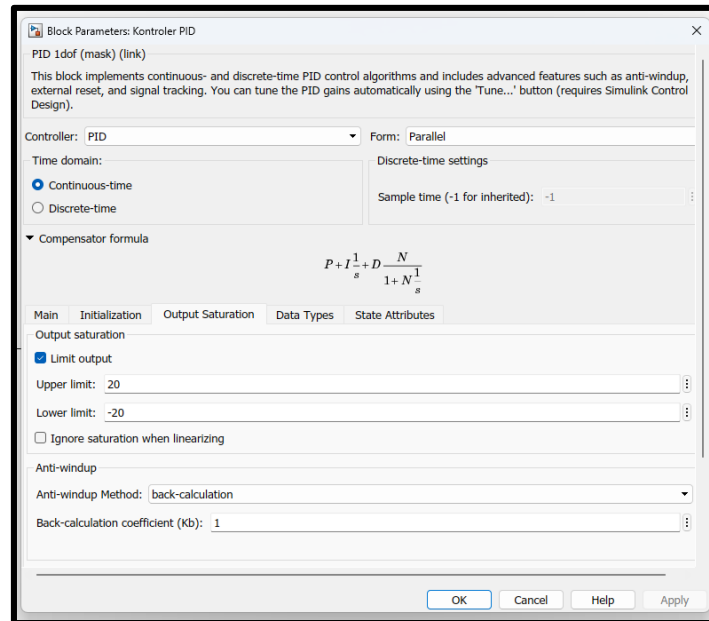
Pada gambar 3.7 dan gambar 3.8 menyoroti dua permasalahan yang muncul.

1. Dari grafik gambar 3.7 permasalahannya adalah ketika nilai *setpoint* 10 ada penundaan yang cukup lama sebelum *output* kontroler PID kembali ke dalam kisaran aktuator.
2. Permasalahan pada gambar 3.8 yaitu ketika nilai *setpoint* menjadi 20, sinyal kontrol PID mencapai kondisi sekitar 53 jauh diluar jangkauan aktuator. Suatu kondisi yang dikenal dengan istilah *windup*.

4.2 Kinerja Performa Sistem Menggunakan Anti-Windup

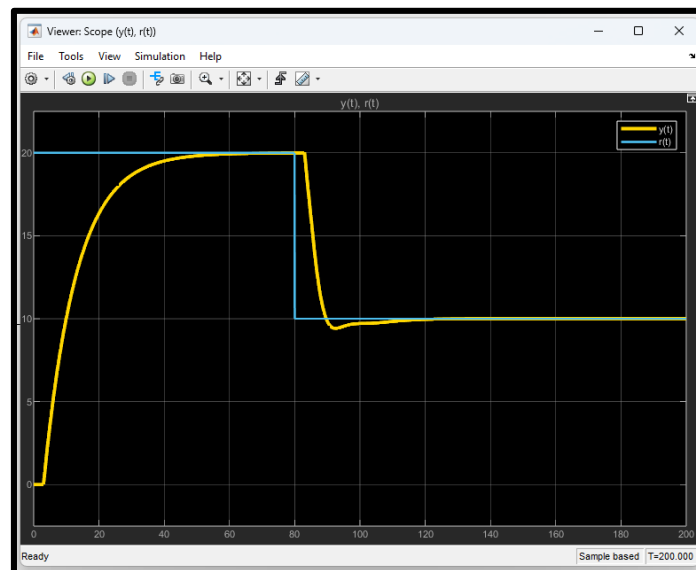
Untuk mengaktifkan metode *anti-windup*, buka tab “Output Saturation” pada blok kontroler PID, kemudian pilih “Limit Output”, dan masukan batas saturasinya [20 -20]. Kemudian pilih “Back-Calculation” dari menu *anti-windup method*. Terlihat pada gambar 3.9.

Metode anti-windup back-calculation adalah strategi yang digunakan dalam sistem kontrol PID untuk mencegah windup pada integral, yaitu kondisi di mana keluaran integrator pengontrol terus menumpuk saat aktuator mencapai batas saturasi, menyebabkan respons yang tidak diinginkan. Metode ini bekerja dengan menggunakan loop umpan balik internal yang melepaskan integrator internal pengontrol ketika mencapai batas saturasi, memungkinkan sinyal kontrol untuk kembali ke wilayah linier.

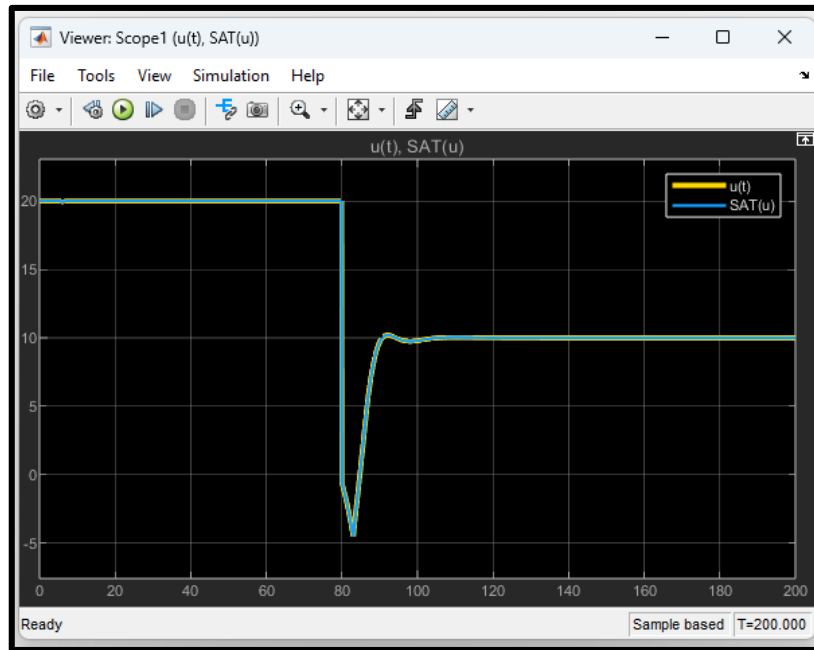


Gambar 3.9 Mengaktifkan metode *anti-windup* dengan *back-calculation*

Hasil dari kinerja performa setelah metode *anti-windup* diaktifkan dapat dilihat pada grafik gambar 3.10 dan gambar 3.11.



Gambar 3.10 Setpoint dan output kontroler menggunakan *anti-windup*

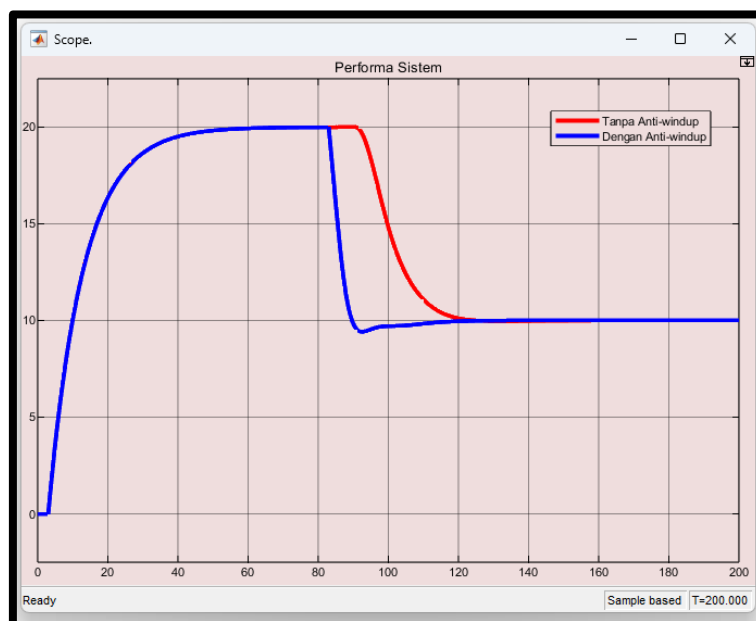


Gambar 3.11 *Output* kontroler dan batas saturasi dengan *anti-windup*

Pada grafik gambar 3.10 dan gambar 3.11 memperlihatkan hasil simulasi pemodelan sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai *dead time* setelah metode *anti-windup* diaktifkan.

- Gambar 3.10 memperlihatkan perubahan sinyal kontrol PID lebih cepat untuk kembali kedalam kisaran aktuator.
- Gambar 3.11 menunjukkan bahwa keluaran pengontrol $u(t)$ dan masukan saturasi $SAT(u)$ saling berhimpitan karena batas saturasi pada kontroler PID dan metode *anti-windup* diaktifkan.

Gambar 3.12 menunjukkan perbandingan hasil sinyal *output* kontroler dengan menggunakan *anti-windup* dan hasil sinyal tanpa menggunakan *anti-windup*.



Gambar 3.12 *Output* kontroler dengan dan tanpa *anti-windup*

4. KESIMPULAN

Setelah menganalisis hasil grafik dari simulasi sistem kendali *anti-windup* dengan kontroler PID pada sistem yang mempunyai dead time. Maka dapat ditarik kesimpulan :

- Menggunakan metode anti-windup pada kontroler PID dapat mempercepat sinyal output kontroler PID untuk kembali pada kisaran aktuator dan mempercepat dalam mencapai setpoint yang diinginkan.
- Metode *anti-windup back-calculation* dapat mencegah terjadinya efek *windup* yang dapat menyebabkan kinerja kontrol menjadi buruk, seperti overshoot yang berlebihan.
- Dengan menerapkan anti-windup, sistem kendali kontroler PID dapat bekerja lebih optimal.

5. SARAN

Dari penelitian yang dilakukan masih dirasa ada kekurangan sehingga kritik dan saran sangat dibutuhkan kedepannya. Saran untuk penelitian selanjutnya agar penelitiannya jauh lebih kompleks yaitu dengan melakukan pengujian dengan menggunakan lebih banyak lagi referensi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada kedua orang tua, saudara, pacar, bapak dosen pembimbing, dan teman-teman teknik elektro ITNY yang telah memberi dukungan, motivasi, doa, terhadap keberhasilan penelitian ini.

DAFTAR PUSTAKA

- Angel, L., Viola, J. dan Paez, M. (2019). Evaluation of the Windup effect in a Practical PID Controller for the Speed Control of a DC-Motor System. 2019 IEEE 4th *Colombian Conference on Automatic Control (CCAC)*, October 15-18, 2019. Diez Hotel, Medellin, Colombia.
- Astrom, K. J. dan Hagglund, T. (1995). *PID Controllers, Theory, Design and Tuning*, 2nd edition. Instrument Society of America.
- Astrom, K. J. dan Hagglund, T. (2001). The Future of PID Control. *Control Engineering Practice*, Vol 9, pp 1163-1175.
- Iskandar, R. F., Putra, R. dan Suhendi, A. (2009). Analisis Pengaruh Kontrol PI Dengan Integral *Anti-Windup* Sebagai Upaya Reduksi Lonjakan Respon pada Sistem Ruang Termal. *Jurnal Sistem Komputer*, Vol. 8, No. 1, April.
- Kusnadi, Aji A, dan Nanda, E. A. (2018). Analisa Metode PID pada Plant Water Heater dengan Karakteristik FOPDT (First Orde Plus Dead Time). *Teknik Elektro. UI Depok*. Vol. 3 : 151-157.
- Okelola, M. O., Aborisade1, D. O. dan Adewuyi, P. A. (2020). Performance and Configuration Analysis of Tracking Time Anti-windup PID Controllers. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)* Vol. 6, No. 2, December, pp. 20-29.
- Pratama, S. C., Susanto, E. dan Wibowo, A. S. (2016). Design and Implementation of Water Level Control Using Gain Scheduling PID Back Calculation Integrator Anti Windup. The 2016
-

International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), pp. 101-104.

Putra, A. Y. (2017). Optimasi Tuning Kontrol PID Pada Amine Regenerator Di Proses Sweetening Gas. Surabaya: Dapertemen Teknik Fisika. Institut Teknologi Sepuluh Nopember

Ribic, A. I. dan Matausek, M. R. (2016). Antiwindup PID Controller Design of TITO System Based on Ideal and Inverted Decoupling. 2016 International Symposium on Industrial Electronics (INDEL), Banja Luka, Bosnia and Herzegovina, 3-5 November.

Setiawan, I. (2008). Kontrol PID Untuk Proses Industri Beragam Struktur dan Metode Tuning PID praktis. PT Elek Media Komputindo. 978-979-27-4100-1.

Wahyudi, Wara B, dan Setiyono B. (2017). Kinerja Kontroler Internal Model Kontrol (IMC) Pada Plant First Orde Plus Dead Timr (FOPDT). Teknik Elektro, UNDIP Tembalang.Semarang. e-ISSN 2407-6422, 59.

Wahyungoro O., dan Arisman.. Simulasi Anti Integral Windup dengan Clamp Integrator untuk Pengendali PI Menggunakan Matlab 6.1. Teknik Elektro. UGM Yogyakarta.
