

Proses Simulasi Separator FWKO Berbasis Labview

Abiyu Ahmad Muhfadhhal¹, Almira Budiyanto², Setyawan Wahyu Pratomo³

¹ Program Studi Teknik Elektro, Universitas Islam Indonesia, Yogyakarta, Indonesia

² Program Studi Teknik Elektro, Universitas Islam Indonesia, Yogyakarta, Indonesia

³ Program Studi Teknik Elektro, Universitas Islam Indonesia, Yogyakarta, Indonesia

Kata kunci:

Separator, FWKO, Level Liquid.

Email penulis:

14524100@students.uii.ac.id

Abstract

Penelitian ini bertujuan memahami struktur kerja *separator* dan meningkatkan keamanan pada area kerja *separator*, sehingga kejadian seperti kebocoran, kebakaran dan ledakan tidak terulang. FWKO *separator* adalah sistem memisahkan air bebas (*free water*) dari liquid. FWKO dipasang pada aliran fluida sesudah *scrubber* yang memisahkan gas dari aliran minyak. Metodologi dari penelitian ini adalah perancangan sistem *simulation* dengan pengendali PID. Aplikasi yang digunakan untuk memproses data ini adalah LabView. Pengujian *simulation* menunjukkan tingkat *error* 0,351% dengan menggunakan parameter $P=10$, $I=0,05$ sehingga untuk selanjutnya konsep pengendalian tersebut dapat digunakan sebagai kontrol *simulation* pada *separator*. Pengujian respon menggunakan perhitungan nilai *error* sebagai selisih terhadap *set point*. Pengujian sistem kendali PID menunjukkan hasil pengendalian yang menggunakan parameter *Propotional (P)*, *Integral (I)*, *Derevative (D)* terhadap respon pengendalian dari selisih *error* yang diukur.

1. Pendahuluan

Free Water Knock Out merupakan kepanjangan dari FWKO. FWKO *separator phase* bertujuan untuk memisahkan air bebas (*free water*) dari minyak. FWKO dipasang pada aliran *fluida* sesudah *scrubber* yang memisahkan gas dari aliran minyak, sehingga tekanan operasi di FWKO tidak tinggi. Selain itu, kapasitas pemisahan gasnya pun tidak besar. Umumnya FWKO merupakan bejana tekan horizontal. Performa pemisahan FWKO biasanya diukur berdasarkan kandungan air dalam minyak.



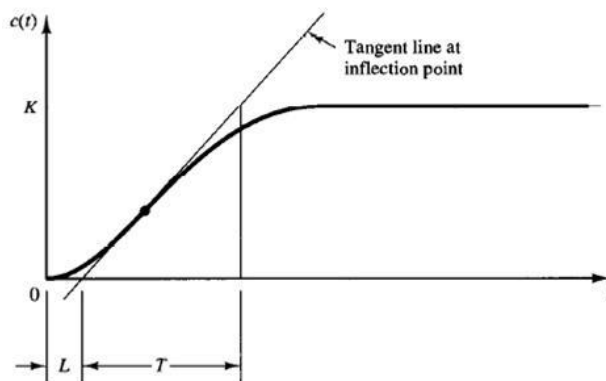
Gambar 1 Separator Tank FWKO

Pada Gambar 1 fluida dialirkan masuk ke FWKO, disini terjadi pemisahan antara minyak, air dan gas. Pemisahan dilakukan dengan mengatur ketinggian fluida di dalam *separator tank*, pada ketinggian 50% itu adalah ketinggian air yang di set kemudian pada ketinggian 50%-70% terjadilah pemisahan antara minyak air dan gas. Minyak didalam *separator* tersebut akan masuk ke penampungan minyak, ketinggian dari minyak dapat dilihat dari Glass Guage di *separator tank*. Gas yang dihasilkan dari proses pemisahan *fluida* tersebut akan dialirkan ke *line* gas yang kemudian terhubung ke PLTG milik perusahaan. Air dari pemisahan *fluida* akan dialirkan ke *skim tank* dimana nantinya air tersebut akan dipompakan ke sumur minyak dengan menggunakan pompa.

2. Metode Penelitian

2.1 Metode Ziegler-Nichols I

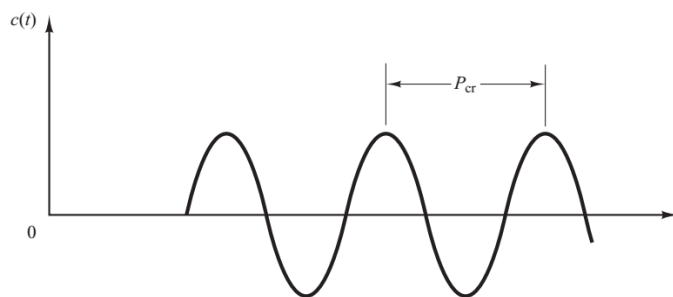
Metode Ziegler-Nichols I ini juga disebut dengan metode Ziegler-Nichols kalang terbuka. Langkah-langkah untuk menentukan nilai parameter proporsional (P), Integral (I), dan Derivative (D) pada metode ini, adalah dengan melihat *output* respon sistem ketika *plant* diberi *input unit step*, dan dilakukan dengan kondisi *plant* dalam keadaan kalang terbuka. Gambar 2 merupakan aturan *tuning* pada Metode Ziegler-Nichols I (Ogata, 2010).



Gambar 2 Metode Ziegler-Nichols I (Ogata, 2010)

2.2 Metode Ziegler-Nichols II

Metode Ziegler-Nichols II, juga disebut dengan metode Ziegler-Nichols kalang tertutup. Langkah-langkah untuk menentukan nilai parameter-parameternya dilakukan dengan cara kalang tertutup pada *plant* tersebut. Dengan memberi *input* dengan *unit step* dan hanya memberikan nilai K_p mulai dari 0 hingga nilai kritis K_{cr} , maka akan diperoleh nilai *output* yang berisolasi. Nilai dari K_{cr} ini akan divariasikan sampai *output* respon sistem mempunyai nilai amplitudo yang sama ketika berisolasi. Gambar merupakan aturan *tuning* pada Metode Ziegler-Nichols II (Ogata, 2010).



Gambar 3 Metode Ziegler-Nichols II (Ogata, 2010)

2.3 Metode Pengendalian PID

Pengendalian PID pada sistem *separator* ini menggunakan pengendali yang banyak dipakai dalam dunia industri. Pengendali akan memberikan input kepada *control valve* berdasarkan besar *error* yang diperoleh dari selisih *level liquid* terhadap *set point*. *Error* adalah perbedaan *set point* dengan *level actual*. Persamaan pengendali PID adalah :

$$mv(t) = K_p \left(e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{\partial e(t)}{\partial t} \right)$$

Keterangan:

$mv(t)$: *manipulated variable* keluaran dari pengontrol PID

K_p : penguatan proporsional

T_i : waktu integral

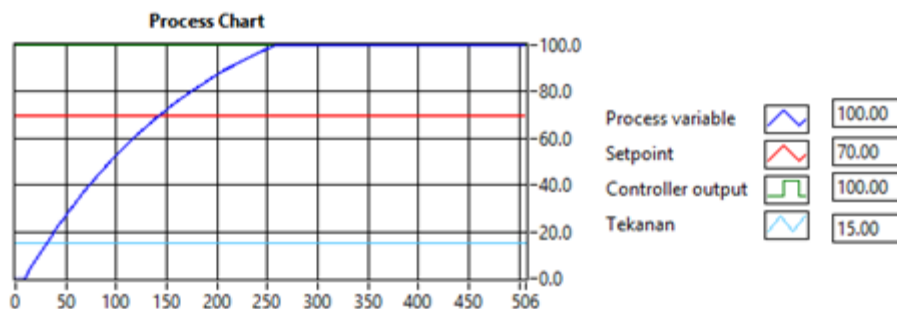
T_d : waktu derivative

$e(t)$: *error* = *setpoint* – keluaran

3. Hasil dan Pembahasan

3.1. Open Loop Control System

Open loop control system atau kontrol loop terbuka adalah suatu sistem yang keluarannya tidak mempunyai pengaruh terhadap aksi pengendalian. Sistem kontrol terbuka keluarannya tidak dapat digunakan sebagai umpan balik dalam masukan. Pengujian respon kendali *open loop* untuk mendapatkan hasil yang diperlihatkan pada Gambar 4.



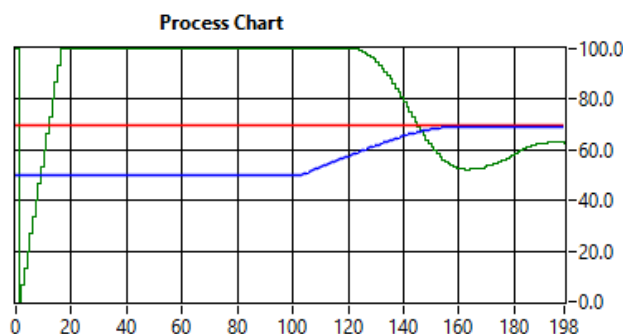
Gambar 4 Grafik Respon *Open Loop*

Grafik respon *open loop* menunjukkan terjadinya *overshoot* terhadap grafik pengendalian. Dimana terdapat *error* yang sangat besar dan tidak adanya aksi pengendalian pada *output* sehingga nilai *error* yang didapat semakin membesar. Kelemahan dalam *open loop* ini kualitas dari *output* tidak dapat dipastikan sama dengan *set point*. Akibatnya, diperlukan pengaturan yang terus menerus hingga mendapat nilai yang diharapkan.

Pada sistem *open loop*, output tidak dapat dibandingkan dengan *set point*. Sehingga, untuk setiap *set point* tidak dapat melakukan pengendalian dengan nilai yang ditentukan. System control *open loop* dapat digunakan hanya jika diperlukan sebuah pengendalian yang *output* tidak menjadi gangguan pada sistem lainnya memerlukan. Sebagai contohnya pada control *safety valve*.

3.2 Close Loop Control System

Pengujian respon kendali *closed loop* untuk mendapatkan hasil yang sesuai dengan *setpoint* diperlihatkan pada Gambar 5



Gambar 5 Grafik respon *closed loop*

Pada Gambar 5 menjelaskan grafik respon dari pengendali dimana terdapat 4 indikator yang menjelaskan *process variable*, *setpoint*, *controller output*, dan *pressure*. Line ungu menjelaskan *process variable*, line merah menunjukkan *setpoint*, line hijau menjelaskan *controller output*, dan line biru menjelaskan *pressure*. Sebuah pengendalian pada *control simulation* ini sudah cukup baik dimana dijelaskan pada grafik respon, selisih atau jarak antara *setpoint* dengan *process variable* berjarak 0,70 dimana hasil tersebut didapat dari *setpoint* dikurang *process variable*.

3.3 Pengujian Sistem Kendali PID

Pengujian pada kendali PID yang merupakan kontroler untuk menentukan ketepatan suatu sistem instrument dengan karakteristik adanya umpan balik/*feedback* pada sistem tersebut. Komponen pengendali PID antara lain komponen proposional, komponen integral dan komponen *derivative*. Komponen tersebut dapat dipakai semua atau hanya beberapa.

3.3.1 Pengujian Parameter P

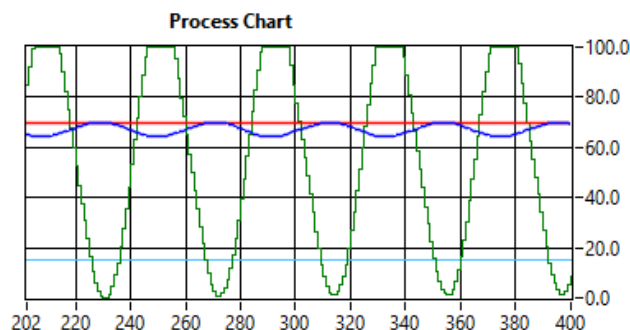
Pengujian parameter *Proporsional* (P) dengan *setpoint* 70 pada separator untuk melihat respon. Berikut adalah hasil pengujian dengan parameter P ditampilkan pada Table 1.

Tabel 1 Pengujian Parameter P

Percobaan	Parameter PID			Level Tank	Setpoint	Controller output	Tekanan
	P	I	D				
1	1	0	0	36,35	70	33,66	15
2	5	0	0	60,03	70	49,81	15
3	10	0	0	64,68	70	53,38	15
4	20	0	0	69,3	70	8,42	15

Dari Table 1 dapat dilihat pengaruh terhadap perubahan nilai dari parameter P. Percobaan 1 nilai dari *level tank* separator adalah 36,35 nilai tersebut didapat dari respon *plant* tidak sampai pada *setpoint*. Hal tersebut disebabkan pengali dari nilai *error* tersebut kurang besar sehingga respon tidak mencapai nilai *setpoint*, maka diperlukan penambahan nilai P lebih besar.

Pengaruh parameter P adalah memperkecil konstanta waktu sehingga sistem menjadi lebih sensitif dan mempunyai respons yang lebih cepat. Dengan mengubah-ubah besaran K_p , maka akan mempengaruhi *offset* atau *steady state error*. Pada saat nilai K_p besar maka akan menyebabkan *offset* semakin kecil. Akan tetapi disisi lain, pengontrol P tidak akan melakukan aksi kontrolnya apabila tidak ada sinyal kesalahan $e(t)$.



Gambar 6 Grafik percobaan P = 20

Pada Gambar 6 merupakan grafik dari percobaan 4 nilai P yang diberikan pada pengendali adalah 20 nilai tersebut dapat memperlihatkan respon yang mencapai *setpoint*. Grafik menunjukkan berhasil tetapi hal tersebut berjalan dengan *steady*. Kejadian tersebut dapat dikatakan baik dalam percobaan pada parameter P.

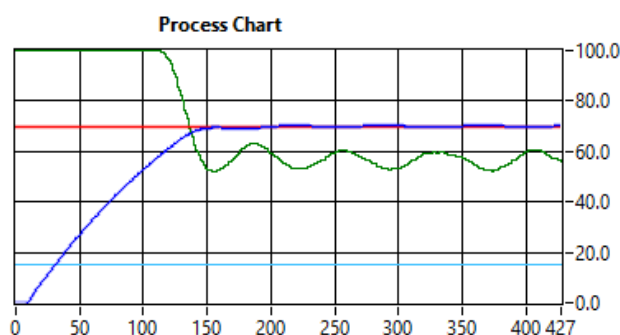
3.3.2 Pengujian Parameter PI

Pengujian parameter PI merupakan penambahan komponen *Integral* (I) kedalam parameter *Proporsional* (P) pada percobaan sebelumnya. Parameter *Integral* (I) adalah penjumlahan dari *error* sebelumnya. Proses penjumlahan akan berlangsung sampai nilai dari output sama dengan *setpoint*. Penggunaan komponen I pada pengujian berfungsi untuk mengurangi *overshoot* serta memperkecil osilasi. Berikut adalah Table 2 hasil pengujian parameter PI.

Tabel 1 Pengujian Parameter PI

Percobaan	Parameter PID			Process variable	Setpoint	Controller output	Tekanan
	P	I	D				
1	20	0,5	0	57,54	70	51,46	15
2	20	0,05	0	71,13	70	11,26	15
3	10	0,5	0	57,72	70	52,81	15
4	10	0,05	0	70,18	70	56,04	15

Dari Table 2 dapat dilihat pengaruh penambahan komponen I berdampak pada semakin besarnya *rise time*. Perbedaan tersebut dapat dilihat dari hasil percobaan P dibandingkan dengan percobaan PI. Pada pengujian ini didapatkan hasil yang cukup baik, nilai respon yang dihasilkan cukup mendekati dari *setpoint*. Hasil tersebut dapat dilihat pada Gambar 7.



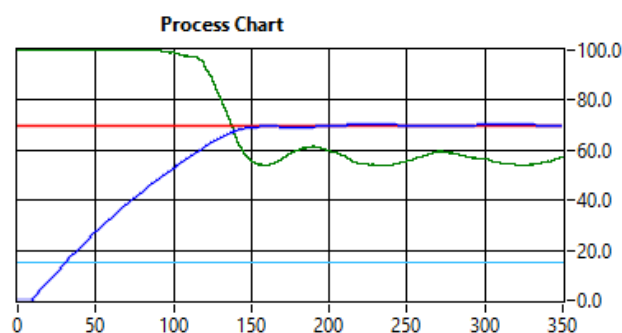
Gambar 7 Grafik percobaan P = 10 dan I = 0,05

Respon grafik pada Gambar 7 tersebut didapat saat nilai P = 10 dan I = 0,05. Grafik pada gambar menunjukkan tidak adanya overshoot dan nilai *output* dari pengendali berhasil *steady* di nilai *setpoint* dengan *error* sekitar 1%. Percobaan ke 4 ini dapat dikatakan cukup berhasil dengan respon grafik demikian. Pada pengujian ini dapat disimpulkan bahwa penambahan komponen I dapat menghilangkan *error steady state*, tetapi berdampak bertambahnya nilai dari *rise time*.

3.3.3 Pengujian Parameter PID

Pengujian PID terdiri dari tiga komponen, yaitu komponen *Proporsional* (P), komponen *Integral* (I) dan komponen *Derivatif* (D). Ketiga komponen ini saling melengkapi satu sama lain, sehingga kelemahan-kelemahan pada salah satu komponen dapat ditutupi oleh komponen yang lain. Komponen I dan D tidak dapat berdiri sendiri dan selalu dikombinasikan dengan komponen P, menjadi pengontrol PI atau PID. Pengontrol PID akan mengeluarkan aksi pengendalian dengan membandingkan kesalahan atau *error* yang merupakan selisih dari *process variable* dan *setpoint*, selisih akan digunakan sebagai masukan pengendali untuk mengeluarkan sinyal kendali ($u(t)$).

Pada pengujian parameter PID respons yang dihasilkan oleh plant ditampilkan pada Gambar 8.



Gambar 8 Grafik percobaan P = 10, I = 0,05 dan D = 0,0001

Penambahan komponen *Derivatif* (D) pada Gambar 8 hasil respon yang diberikan tidak memberikan pengaruh yang cukup besar kepada grafik. Pada saat nilai D diberikan lebih besar akibatnya menimbulkan *noise* cukup mengganggu terhadap performa. Pada *control simulation level liquid separator free water knock-out* ini penambahan komponen Derivatif (D) tidak dilakukan dikarenakan hasil yang didapat menurun dari performa pengujian sebelumnya yang hanya menggunakan parameter PI. Jadi pengendali separator hanya memilih menggunakan parameter PI.

4. Kesimpulan

Berdasarkan hasil dan pembahasan mengenai Proses *Control Simulation Level Liquid Separator Free Water Knock-Out Central Facility* Berbasis Labview, dapat diambil kesimpulan sebagai berikut:

1. *Process control simulation level liquid separator free water knock-out* didapatkan hasil *error* sebesar 0,351% dengan menggunakan parameter $P = 10$, $I = 0,05$ dan $D = 0$. Parameter PI tersebut dipilih sebagai pengendalian yang terbaik dalam proses pengujian ini. Dengan performa yang baik ini diharapkan dapat membantu operator dan *user* dalam sistem monitoring. Dengan begitu implementasi kendali PID sudah berhasil dengan *overshoot* sebesar 0% dan *steady state error* 0%.
2. Proses monitoring yang tepat agar pemisahan *fluida* dan gas berkeja optimal secara baik yaitu dengan adanya indikator pendukung yang memberikan informasi proses yang terjadi didalam *separator*. Pada *separator FWKO* di PT. Pertamina EP Asset 1 Field Jambi sistem yang digunakan masih manual, dimana *control valve* masih menggunakan tenaga manusia untuk menggerakkan. Oleh karena itu dibutuhkan *control simulation level liquid separator free water knock-out* ini. Dengan demikian informasi dari tekanan, *level* air dan *level* minyak dapat termonitor keadaanya.
3. Sistem kendali yang dibutuhkan agar pemisahan *fluida* dan gas optimal pada *control simulation level liquid separator free water knock-out* ini adalah dengan menggunakan pengendali PID. Dimana dengan pengendali PID akan mengeluarkan aksi kontrol dengan membandingkan kesalahan atau *error* yang merupakan selisih dari *process variable* dan *setpoint*, yang akan digunakan sebagai masukan pengontrol untuk mengeluarkan sinyal kontrol.
4. Pada hasil pengujian *close loop* nilai kendali yang dihasilkan simulasi memiliki *error* yang sangat kecil sebesar 0,351%.

Daftar Pustaka

- W. E. Odiete and J. C. Agunwamba, "Residence Time versus Aspect Ratio of Conventional Oil / Water Separators Aspect Ratio vrs Residence Time," *Int. J. Eng. Trends Technol.*, vol. 32, no. 1, pp. 2015–2017, 2016.
- Drn/dis, "Pipa Sumur PT Geo Dipa Energy Meledak Hebat Satu Tewas, Lima Luka-Luka_Radar Banyumas," *Radar Banyumas*, 2016. [Online]. Available: <http://radarbanyumas.co.id/pipa-sumur-pt-geo-dipa-energy-meledak-hebat-satu-tewas-lima-luka-luka/>. [Accessed: 27-Jan-2018].
- W. E. Odiete and J. C. Agunwamba, "Effect Of Aspect Ratio On The Oil Separation Efficiency Of Conventional Oil/Water Separators," *Int. J. Sci. Eng. Res.*, vol. 7, no. 3, pp. 840–847, 2016.
- M. Ahmed Mahmoud, "Hydrodynamic Separator Unit for R[1] M. Ahmed Mahmoud, 'Hydrodynamic Separator Unit for Removal and Recovery Oil from Wastewater,' *J. Pet. Environ. Biotechnol.*, vol. 7, no. 2, pp. 2–4, 2016.emoval and Recovery Oil from Wastewater," *J. Pet. Environ. Biotechnol.*, vol. 7, no. 2, pp. 2–4, 2016.
- L. A. Sabri and H. A. Al-mshat, "Implementation of Fuzzy and PID Controller to Water Level System using LabView," *Int. J. Comput. Appl.*, vol. 116, no. 11, pp. 11–15, 2015.
- G. N. Sapkale, S. M. Patil, U. S. Surwase, and P. K. Bhatbhage, "Supercritical Fluid Extraction," *Int. J. Chem. Sci.*, vol. 8, no. 2, pp. 729–743, 2010.
- A. I. Neamah, "Separation of the Petroleum System," *Hilltop Rev.*, vol. 7, no. 1, 2014.
- K. Ogata, *Modern Control Engineering*, 5th ed., vol. 17, no. 3. Prentice Hall, 2010.
- N. Instruments, "LabVIEW Graphical Programming Course," *LabVIEW Learn.*, p. 287, 2007.
- D. Seborg, T. Edgar, D. Mellicamp, and F. Doyle III, "Process Dynamics and Control," *John Wiley Sons*, p. 595, 2011