

# PERANCANGAN AIR TO FUEL RATIO (AFR) CONTROLLER BERBASIS PID ADAPTIF PADA MESIN BENSIN 4 LANGKAH

Dinar Aji Pinunjul<sup>\*)</sup>, Wahyudi dan Sudjadi

Departemen Teknik Elektro, Universitas Diponegoro  
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

<sup>\*)E-mail: dinarajipinunjul@gmail.com</sup>

## Abstrak

*Air Fuel Ratio* (AFR) atau rasio pembakaran udara dengan bahan bakar pada sebuah kendaraan mesin bensin 4 langkah merupakan salah satu variabel penting yang menentukan kesempurnaan pembakaran internal mesin. Pengontrolan AFR pada mesin bensin 4 langkah diharapkan menghasilkan rasio udara dengan bensin pada rasio tertentu agar kriteria pembakaran sempurna atau *stoichiometric* terpenuhi pada saat mesin beroperasi. Nilai pembakaran sempurna adalah saat AFR bernilai 14,67. Dikarenakan spark ignition engine merupakan sistem yang bersifat non linier maka diperlukan kontroler yang handal yang dapat mempertahankan nilai AFR salah satunya yaitu kontroler PID adaptif. Berdasarkan hal tersebut, dirancang kontroler AFR pada model mesin bensin 4 langkah menggunakan metode kontrol PID adaptif algoritma Dahlin. Kontroler PID adaptif digunakan untuk mengontrol jumlah bensin yang akan masuk ke ruang pembakaran dari sistem agar AFR berada pada nilai yang diinginkan yaitu 14,67 untuk mesin berbahan bakar bensin. Semakin kecil IAE dan ITAE dari AFR sistem atau mendekati nilai 14,67 maka dapat dikatakan kontroler berfungsi dengan baik. Berdasarkan implementasi kontroler PID adaptif pada mesin bensin 4 langkah dihasilkan nilai rata-rata IAE dan ITAE dari AFR sebesar 0,4423 dan 0,4699.

*Kata kunci: air fuel ratio, mesin bensin 4 langkah, stoichiometric, Dahlin, IAE, ITAE*

## Abstract

Air Fuel Ratio of combustion in a spark ignition engine vehicle is one of the important variables that determine the perfection of the internal combustion engine. Controlling of air fuel ratio in spark ignition engine was expected to producing air to gasoline ratio perfect combustion or stoichiometric was satisfied when the engine is operating. Perfect combustion occurs when AFR value is 14.67. Because of the spark ignition engine is a non linear system so controller will require a reliable one that is an Adaptive PID controller. Accordingly, the air fuel ratio controller designed on the model of spark ignition engines using Adaptive PID controller with Dahlin's algorithm. Adaptive PID controller will be used to control the amount of fuel that will going into the combustion chamber so that air fuel ratio is at the desired value is 14.67 for gasoline engine. The smaller of the IAE and ITAE of AFR or approach 14.67 for the value so controller could be said to worked properly. Based on the implementation of Adaptive PID controller on spark igniton engine was obtained the smallest IAE and ITAE average of AFR that is 0.4402 and 0.4499.

*Keywords: air fuel ratio, spark ignition engine, stoichiometric, Dahlin, IAE, ITAE*

## 1. Pendahuluan

Emisi dari kendaraan bermotor merupakan salah satu penyebab utama dalam pencemaran udara. Gas buang *spark ignition engine* mengandung NO, NO<sub>x</sub>, CO, dan hidrokarbon yang belum terbakar secara sempurna [1]. Pengontrolan AFR dalam sistem pembakaran bahan bakar pada mesin perlu dilakukan karena kondisi pemanasan global yang semakin memburuk, pencemaran lingkungan serta semakin meningkatnya produksi dan pemakaian alat transportasi (motor bakar), sehingga diharapkan dapat menekan angka pencemaran udara dan pemanasan global. Untuk memenuhi hemat energi (*low fuel consumption*) dan ramah lingkungan (*low emission*) dapat dilakukan dengan

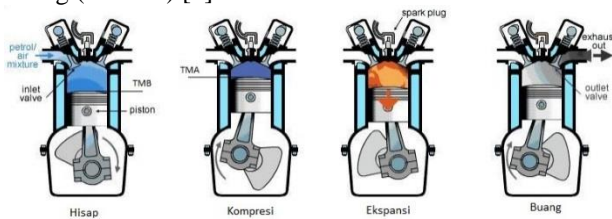
cara memperbaiki proses pembakaran, yaitu mengatur *Air to Fuel Ratio* (AFR) sehingga mendekati proses pembakaran yang ideal atau sesuai dengan kebutuhan pengendara [2]. Secara umum nilai AFR dijaga tetap berada pada nilai ideal atau stoikiometri sebesar 14,67 [3]. Analoginya, 1 kilogram bensin atau *gasoline* akan habis terbakar jika dicampur dengan 14,67 kilogram udara dalam prosesnya. Variasi nilai AFR kurang dari 1% dari 14,67 disebut dengan campuran kaya (*Rich Fuel*), sedangkan variasi nilai AFR lebih dari 1% dari 14,67 disebut dengan campuran miskin (*Lean Fuel*) [4]. Beberapa penelitian tentang AFR mesin bisa menjadi referensi dalam membuat sebuah desain kontrol AFR pada mesin pembakaran internal. Penelitian mengenai AFR dilakukan oleh Reactor

dkk untuk mendesain kontrol AFR pada *spark ignition engine* dengan metode kontrol *Hybrid Fuzzy PI* [5]. Triwiyatno dkk melakukan perancangan kontrol PID berbasis ANFIS untuk AFR pada model mesin bensin 4 langkah [6]. Ahmed dkk melakukan perancangan kontroler *Sliding Mode Controller* pada mesin bensin 4 langkah [7]. Penelitian tersebut mengontrol AFR dalam keadaan ideal sehingga nilai AFR sesuai stoikiometri. Pada Penelitian ini dirancang sebuah kontroler PID Adaptif dengan menggunakan metode PID Dahlin untuk melakukan *tuning* parameter kontrol proporsional, integral, dan derivatif untuk mengatur AFR pada kondisi ideal pada model mesin bensin 4 langkah menggunakan simulasi MATLAB yang secara tidak langsung akan mengurangi kadar emisi gas buang dan mengoptimalkan konsumsi bahan bakar sehingga menjadi teknologi yang ramah lingkungan.

## 2. Metode

### 2.1. Mesin Bensin 4 Langkah

Mesin bensin atau mesin Otto dari Nikolaus Otto adalah suatu tipe mesin pembakaran dalam yang menggunakan nyala busi untuk proses pembakarannya (*spark ignition engine*) dan dirancang menggunakan bahan bakar bensin [4]. Mesin bensin 4 langkah memiliki 4 langkah berurutan, yaitu, langkah hisap (*intake*), langkah kompresi (*compression*), langkah ekspansi (*expansion*), dan langkah buang (*exhaust*) [4].



Gambar 1. Siklus kerja mesin 4 langkah

Gambar 1 menunjukkan siklus kerja mesin 4 langkah. Model mesin yang digunakan adalah model yang dilakukan oleh Anna Stefanoupolou [8], Rohit M [9] dan Yang Bai [10]. Dinamika sistem dimulai dengan *engine* (mesin) menerima masukan berupa *throttle* dengan bukaan sudut tertentu. Bukaan sudut *throttle* melewati udara dengan massa aliran tertentu yang kemudian berada pada *intake manifold*. Setelah itu, udara masuk ke dalam silinder di mana proses pemampatan dan pembakaran terjadi. Dari proses pembakaran tersebut dihasilkan torsi mesin untuk menggerakkan *crankshaft* dengan kecepatan tertentu dan udara sisa pembakaran keluar dari *exhaust manifold*. Untuk menjelaskan hasil pemodelan pada penelitian ini dapat diuraikan melalui beberapa bagian.

#### 2.1.1. Breathing process dynamics

Pada proses ini, udara masuk ke *intake manifold* dengan diatur oleh *throttle*. Dinamika pada *manifold* dideskripsikan melalui persamaan diferensial orde satu

yang menghubungkan tingkat perubahan tekanan pada *manifold* ( $P_m$ ) dengan tingkat aliran yang masuk dan keluar dari *intake manifold* ( $\dot{m}_\theta$  dan  $\dot{m}_{cyl}$ ) seperti pada persamaan (1) [8][9].

$$\frac{d}{dt}P_m = 0,413(\dot{m}_\theta - \dot{m}_{cyl}) \quad (1)$$

Tingkat aliran udara yang masuk ke *intake manifold* ( $\dot{m}_\theta$ ) melalui *throttle* merupakan fungsi dari sudut bukaan *throttle* ( $\theta$ ) dan tekanan hulu ( $P_o$ ) diasumsikan sebagai tekanan atmosfer, sehingga  $P_o = 1$  bar serta tekanan pada hilir merupakan tekanan pada *manifold* ( $P_m$ ). Ketika tekanan pada *manifold* bernilai kurang dari setengah tekanan atmosfer, tingkat aliran udara yang masuk ke *manifold* berada dalam kondisi *sonic flow*, dimana nilai  $\dot{m}_\theta$  hanya bergantung pada fungsi dari sudut bukaan *throttle* ( $\theta$ ) saja, karena  $g(P_m)$  bernilai 1, seperti yang dijelaskan dalam persamaan (2)-(4) [8][9].

$$\dot{m}_\theta = f(\theta)g(P_m) \quad (2)$$

$$f(\theta) = 2,821 - 0,05231\theta + 0,10299\theta^2 - 0,00063\theta^3 \quad (3)$$

$$g(P_m) = \begin{cases} 1 & , \text{ if } P_m \leq \frac{P_o}{2} \\ 2\sqrt{P_m - P_m^2} & , \text{ if } P_m > \frac{P_o}{2} \end{cases} \quad (4)$$

Tingkat aliran udara yang dipompa pada mesin secara umum merupakan fungsi dari tekanan pada *manifold* ( $P_m$ ) dan *engine speed* ( $N$ ) yang didapatkan secara empiris [8][9].

$$\dot{m}_{cyl} = -0,366 + 0,008979NP_m - 0,0337NP_m^2 + 0,0001N^2P_m \quad (5)$$

#### 2.1.2. Exhaust process dynamic

Langkah terakhir pada mesin 4 tak adalah langkah *exhaust* (pembuangan) yang menghasilkan gas NOx dan sisa oksigen yang tidak terbakar atau sering disebut dengan istilah *Exhaust Gas Oxygen* (EGO). Persamaan dari EGO dapat dilihat pada persamaan (6) [11][12].

$$EGO = \frac{1 - \tanh(4(AFR - 14,67))}{2} \quad (6)$$

#### 2.1.3. Compression

Blok *compression* sebenarnya tidak termasuk dalam bentuk fisik *plant*. Blok tersebut hanya membantu perhitungan dalam menghasilkan perhitungan massa udara yang masuk pada tabung pembakaran tiap terjadinya proses pembakaran ( $\dot{m}_a$ , *g/intake event*). Integrator mengakumulasi aliran massa udara silinder di dalam blok *intake*. *Valve timing* blok mengeluarkan pulsa yang sesuai dengan posisi rotasi untuk mengatur waktu *intake and*

compression. Proses terbukanya katup pada silinder terjadi saat rotasi cam atau setiap crankshaft berotasi 180°. Setiap pembukaan katup, udara masuk kedalam silinder dan terjadi kompresi. Output dari trigger blok dengan udara yang terkompresi kemudian diumpan balik ke intake integrator untuk melakukan reset. Compression subsystem menggunakan blok unit delay untuk memasukkan 180° delay antara intake dan combustion di setiap air charge. Terjadi siklus komplit mesin 4 langkah (four stroke cycle) ketika proses hisap, dimana katup hisap terbuka sehingga blok intake simulink megumpulkan integerates massa aliran udara dari manifold. Setelah crank berotasi 180°, katup hisap tertutup dan unit delay pada compression subsystem berjalan. Setelah itu terjadi proses pembakaran/combustion sampai 180° selanjutnya dan katup terbuka lagi untuk membuang sisa pembakaran/exhaust [9], [11].

2.1.4. Torque Generation

Torsi yang dihasilkan oleh mesin bergantung pada pengapian silinder, formasi pencampuran antara udara dan bahan bakar, serta parameter – parameter fisik mesin yang spesifik. Persamaan (7) adalah persamaan torsi yang didapatkan melalui percobaan secara empiris [11][8][9].

$$T_b = -181,3 + 379,36m_a + 21,91 \frac{A}{F} - 0,85 A/F^2 + 0,26\sigma - 0,0028\sigma^2 + 0,027 N - 0,000107 N^2 + 0,00048 N\sigma + 2,55\sigma m_a - 0,05\sigma^2 m_a \quad (7)$$

dengan

- $m_a$  : massa dari udara yang diisikan (g / intake event)
- $A/F$  : air-to-fuel ratio
- $\sigma$  : sudut dari spark advance
- $N$  : engine speed (rad/sec)

2.1.5. Torque to Engine Speed

Untuk mendapatkan nilai kecepatan dengan satuan rad/s dan rpm, hasil torsi dibagi dengan engine rotational moment of inertia yang ditunjukkan dalam persamaan (8) [11], [13].

$$J\dot{N} = T_b \quad (8)$$

dengan

- $J$  = engine rotational moment of inertia dalam satuan kg.m<sup>2</sup>
- $\dot{N}$  = kecepatan mesin dalam satuan rad/s
- Untuk mendapatkan kecepatan mesin dalam satuan rpm, maka hasil kecepatan mesin dalam satuan rad/s dikalikan dengan 30/pi. Dalam Penelitian ini nilai moment of inertia yang digunakan bernilai 0,14. Nilai tersebut didapatkan dari hasil penelitian sebelumnya pada mobil Ford Company yang dilakukan oleh Anna dan Cook [8], [11].

2.1.6. Air Flow Estimation

Estimasi aliran udara intake didapatkan dari hasil pekalian engine speed (N), manifold pressure (P<sub>m</sub>) dan time-varying scale factor (C<sub>pump</sub>). C<sub>pump</sub> didapatkan dari lookup table dan dikalikan dengan speed dan pressure untuk mendapatkan nilai estimasi awal aliran udara pada intake. Rumus estimasi dapat dilihat pada persamaan (9) [11],[14].

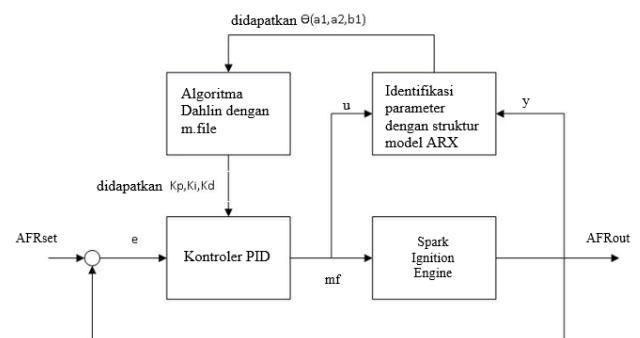
$$q = C_{pump}(N, P_m)NP_m \quad (9)$$

dengan

- $q$  = Estimasi aliran udara intake
- $N$  = Kecepatan putaran mesin (rad/sec)
- $P_m$  = Tekanan manifold

2.2. Perancangan Kontrol

Kontroler PID Adaptif yang dirancang oleh penulis merupakan kontroler yang dapat melakukan tuning parameter kontrol proporsional, integral, dan derivatif secara otomatis berdasarkan identifikasi parameter pada model mesin bensin 4 langkah. Pada perancangan kontroler PID Adaptif ini digunakan metode PID Dahlin. Gambar 2 merupakan diagram skematik kontroler PID adaptif.



Gambar 2. Blok diagram kontrol PID adaptif

2.2.1. Identifikasi Parameter dengan Recursive Least Square

Inti dari metode ini adalah bahwa kecocokan antara model dengan sistem yang akan diidentifikasi diperoleh dengan meminimumkan selisih kuadrat antara keluaran model dengan keluaran sistem yang diidentifikasi seperti pada persamaan (10).

$$J(\theta) = \sum_{i=0}^k e^2(i) = \sum_{i=0}^k [y(k) - \hat{y}(k)]^2 \quad (10)$$

Skema identifikasi dengan metode Least Square terdapat model identifikasi yang digunakan untuk mengestimasi parameter pada proses. Model identifikasi tersebut dinyatakan pada persamaan (11).

$$\hat{G}(Z) = \frac{\hat{B}(z^{-1})}{\hat{A}(z^{-1})} = \frac{\hat{b}_1 z^{-1} + \hat{b}_2 z^{-2} + \dots + \hat{b}_m z^{-m}}{1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2} + \dots + \hat{a}_n z^{-n}} \quad (11)$$

Error merupakan persamaan kesalahan prediksi, yaitu dihitung berdasarkan selisih antara keluaran proses dengan keluaran model identifikasi yang dirumuskan pada persamaan (12).

$$e(k) = y(k) - \hat{y}(k) \quad (12)$$

Keluaran ( $y(k)$ ) dihitung berdasarkan perkalian antara vektor regresi dan vektor estimasi seperti pada persamaan (13).

$$y(k) = \varphi^T(k) \theta(k-1) \quad (13)$$

dengan vektor regresi

$$\varphi^T(k-1) = [-y(k-1) \dots -y(k-n) \ u(k+m-n) \dots u(k-n)] \quad (14)$$

dan vektor estimasi

$$\theta^T(k) = [a_1 \dots a_n \ b_0 \dots b_m] \quad (15)$$

Dengan menggunakan metode *Recursive Least Square*, parameter  $\theta$  diestimasi dengan menggunakan persamaan (16).

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)\varepsilon(k) \quad (16)$$

$$\varepsilon(k) = y(k) - \varphi^T(k)\hat{\theta}(k-1) \quad (17)$$

$$K(k) = \frac{C(k-1)\varphi(k)}{1 + \varphi^T(k)C(k-1)\varphi(k)} \quad (18)$$

$$C(k) = C(k-1) - \frac{C(k-1)\varphi(k)\varphi^T(k)C(k-1)}{(1 + \varphi^T(k)C(k-1)\varphi(k))} \quad (19)$$

### 2.2.2. Tuning Parameter PID dengan Metode PID Dahlin

Pada perancangan sistem ini parameter PID dapat melakukan *tuning* secara otomatis berdasarkan vektor estimasi pada persamaan (21), (22), dan (23). Perubahan pada nilai vektor estimasi menyebabkan perubahan pada parameter PID [15]. Persamaan kontroler PID yang digunakan adalah seperti pada persamaan (20).

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (20)$$

Adapun untuk melakukan *tuning* tersebut digunakan persamaan

$$K_p = \frac{-(a_1 + a_2)Q}{b_1} \quad (21)$$

$$T_d = \frac{T_0 a_2 Q}{K_p b_1} \quad (22)$$

$$T_i = \frac{-T_0}{\frac{1}{a_1 + a_2} + 1 + \frac{T_d}{T_0}} \quad (23)$$

$T_0$  adalah waktu sampel,  $T_d$  adalah waktu turunan, dan  $T_i$  adalah waktu integral. Variabel  $Q$  pada persamaan (21) dan (22) didefinisikan

$$Q = 1 - e^{-\frac{T_0}{B}} \quad (24)$$

Parameter  $a_1$ ,  $a_2$ , dan  $b_1$  merupakan nilai dari vektor estimasi pada persamaan (15) hasil estimasi pada model mesin bensin 4 langkah.

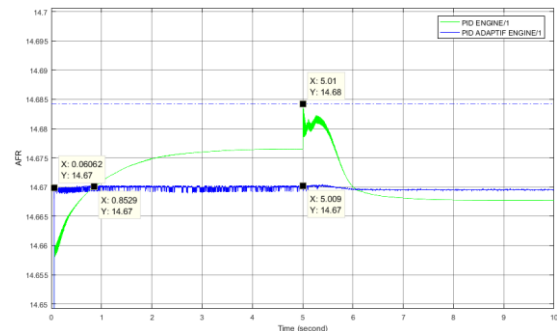
## 3. Hasil dan Analisa

### 3.1. Pengujian Sistem dengan Variasi Nilai Throttle

Pengujian dilakukan dengan memberikan variasi masukan berupa pedal gas (*throttle*) dengan nilai 15°, 30°, 50°, 70° dan 90°. Pengujian yang dilakukan bertujuan untuk mengetahui unjuk kerja kontroler dengan beberapa kriteria pengujian yaitu perbandingan AFR, perbandingan kecepatan mesin, konsumsi bahan bakar, serta perbandingan IAE dan ITAE. Kriteria hasil pengujian pada kontroler PID adaptif Dahlin dibandingkan dengan hasil pengujian pada kontroler PID *trial and error*.

#### 3.1.1. Perbandingan AFR

Perbandingan AFR dengan variasi *throttle* 15° ditunjukkan pada Gambar 3.



Gambar 3. Grafik perbandingan AFR PID dan PID adaptif Dahlin dengan nilai throttle 15°

Dari hasil simulasi didapatkan unjuk kerja sistem dengan kontroler PID AFR hampir mencapai angka 14,684 dengan *overshoot* 0,014 dan pada sistem dengan kontroler PID adaptif AFR nilai puncak mencapai 14,67 dengan *overshoot* sebesar 0. Sistem dengan kontroler PID AFR *setpoint* tercapai pada 0,8592s sedangkan pada sistem kontroler PID adaptif AFR *setpoint* tercapai pada 0,06062s. Setelah terjadi perubahan *throttle* dari 0° ke 15° pada detik ke-5, sistem dengan kontroler PID mengalami lonjakan namun kontroler PID adaptif dapat mempertahankan nilai AFR sehingga tidak terjadi

lonjakan. Pada akhir detik ke 10 kedua sistem PID maupun PID adaptif mencapai *setpoint* dengan nilai AFR akhir 14,67. Nilai AFR pada variasi *throttle* yang lain disajikan dalam Tabel 1.

Tabel 1. Perbandingan AFR

Throttle	AFR PID Adaptif	AFR PID
15°	14,67	14,67
30°	14,67	14,65
50°	14,69	14,94
70°	14,7	15,22
90°	14,71	15,37

Tabel 1 menunjukkan bahwa nilai AFR kontroler PID adaptif lebih mendekati *setpoint* 14,67 daripada PID pada variasi *throttle* 15°, 30°, 50°, 70° dan 90°. *Throttle* bernilai 50°, 70° dan 90° tidak dapat mencapai nilai 14,67 namun mendekati nilai 14,67. Berdasarkan perbedaan nilai AFR dan *setpoint* didapatkan nilai *error* yang dapat dihitung *Integral Absolute Error* (IAE) dan *Integral Time Area Error* (ITAE). IAE untuk kontroler PID Adaptif pada variasi *throttle* 15° memiliki nilai 0,3905, sedangkan untuk kontroler PID memiliki nilai IAE sebesar 0,4282. IAE pada variasi kecepatan lain disajikan dalam Tabel 2. ITAE untuk kontroler PID adaptif pada variasi *throttle* 15° memiliki nilai 0,02443, sedangkan untuk kontroler PID memiliki nilai ITAE sebesar 0,1828. ITAE pada variasi kecepatan lain disajikan dalam Tabel 3.

Tabel 2. Perbandingan Integral Absolute Error (IAE)

Throttle	IAE PID Adaptif	IAE AFR PID
15°	0,3905	0,4282
30°	0,394	0,4935
50°	0,4328	1,167
70°	0,4829	1,915
90°	0,5116	2,317

Berdasarkan Tabel 2 dapat dilihat bahwa semakin besar variasi *throttle* maka nilai IAE semakin besar juga. Begitu juga dengan nilai ITAE.

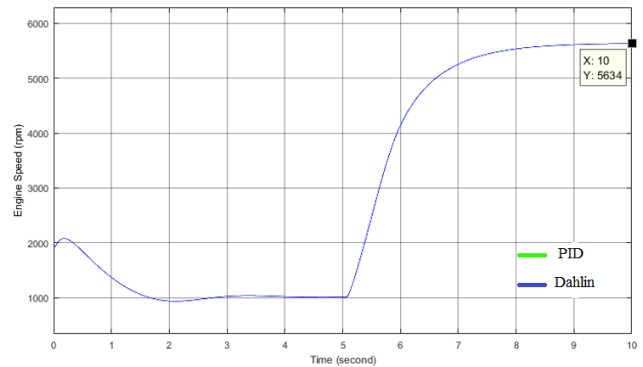
Pada Gambar 4 terlihat nilai kecepatan mesin antara sistem dengan kontroler PID dan sistem dengan kontroler PID adaptif bernilai sama sehingga tidak terlihat perbedaan pada grafik. Pada saat detik ke-5 terjadi perubahan nilai kecepatan dikarenakan terdapat perubahan *throttle* dari 0-15 hingga detik seterusnya. Nilai kecepatan mesin pada sistem PID maupun PID adaptif bernilai sama, yaitu 5634 rpm. Nilai kecepatan mesin pada variasi *throttle* yang lain disajikan dalam Tabel 4.

Tabel 3. Perbandingan Integral Time Area Error (ITAE)

Throttle	ITAE PID Adaptif	ITAE AFR PID
15°	0,02471	0,1828
30°	0,05349	0,7154
50°	0,3876	6,522
70°	0,8177	12,96
90°	1,066	16,43

### 3.1.2. Perbandingan Kecepatan Mesin

Pengujian kecepatan mesin dilakukan untuk melihat nilai kecepatan mesin terhadap waktu serta untuk mengetahui bagaimana pengaruh AFR pada performa kecepatan mesin. Gambar 4 merupakan grafik kecepatan mesin kontroler PID adaptif dan PID dengan nilai variasi *throttle* 15°.



Gambar 4. Grafik perbandingan kecepatan PID dan PID adaptif Dahlin dengan nilai *throttle* 15°

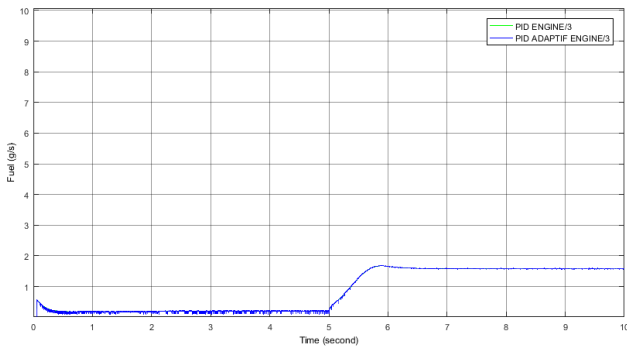
Tabel 4. Perbandingan kecepatan mesin

Throttle	Kecepatan mesin PID Adaptif (rpm)	Kecepatan mesin PID (rpm)
15°	5634	5634
30°	9413	9415
50°	12527	12501
70°	14121	14059
90°	14855	14767

Berdasarkan Tabel 4 dapat dilihat perbedaan nilai kecepatan mesin. Pada pengujian variasi *throttle* 50°, 70° dan 90° nilai kecepatan mesin kontroler PID adaptif lebih unggul dari kontroler PID. Hal tersebut disebabkan perbedaan nilai AFR pada sistem. Kontrol PID adaptif dapat mempertahankan performa kecepatan mesin dengan AFR mendekati *setpoint* sedangkan performa kecepatan mesin kontrol PID berada dibawah PID adaptif dikarenakan tidak dapat mempertahankan nilai AFR *setpoint*. Nilai AFR PID diatas *setpoint* dimana kondisi tersebut dinamakan campuran miskin sehingga semakin besar nilai AFR, maka semakin kecil kecepatan mesin. Semakin kecil nilai AFR, maka semakin besar nilai kecepatan mesin.

### 3.1.3. Perbandingan Konsumsi Bahan Bakar

Pengujian konsumsi bahan bakar dilakukan untuk melihat konsumsi bahan bakar terhadap waktu serta untuk mengetahui pengaruh AFR pada besarnya konsumsi bahan bakar. Gambar 5 merupakan grafik konsumsi bahan bakar kontroler PID adaptif dan PID dengan nilai variasi *throttle* 15°.



Gambar 5. Grafik perbandingan konsumsi bahan bakar PID dan PID adaptif Dahlin dengan nilai throttle 15°

Pada Gambar 5 terlihat nilai konsumsi bahan bakar antara sistem dengan kontroler PID dan sistem dengan kontroler PID adaptif bernilai relatif sama sehingga tidak terlihat perbedaan pada grafik. Pada saat detik ke-5 terlihat konsumsi bahan bakar berubah dikarenakan terdapat perubahan throttle dari 0-15. Hingga detik ke 10, nilai konsumsi bahan bakar pada sistem PID maupun PID adaptif bernilai hampir sama. Konsumsi bahan bakar pada kontroler PID adaptif selama 10 detik bernilai 8,4271 gram dan kontroler PID bernilai 8,4512 gram. Nilai konsumsi bahan bakar dengan variasi throttle yang lain disajikan dalam Tabel 5.

Tabel 5. Perbandingan konsumsi bahan bakar

Throttle	Konsumsi Bahan Bakar 10 detik PID Adaptif (gram)	Konsumsi Bahan Bakar 10 detik PID (gram)
15°	8,4271	8,4512
30°	23,3584	23,4147
50°	45,0311	44,4307
70°	59,3156	57,5610
90°	66,4559	63,8700

Berdasarkan Tabel 5 dapat dilihat perbedaan nilai konsumsi bahan bakar. Pada pengujian variasi throttle 15° dan 30°, konsumsi bahan bakar pada kontroler PID adaptif lebih sedikit dibandingkan kontroler PID. Pada saat variasi throttle 15° dan 30°, nilai AFR kontroler PID adaptif maupun PID bernilai hampir sama, dengan konsumsi bahan bakar PID adaptif lebih unggul. Pada pengujian variasi throttle 50°, 70° dan 90° nilai konsumsi bahan bakar kontroler PID adaptif lebih banyak dari kontroler PID. Hal tersebut disebabkan perbedaan nilai AFR pada sistem. Kontrol PID adaptif dapat mempertahankan AFR mendekati setpoint sedangkan AFR kontrol PID berada diatas setpoint dengan nilai jauh diatas 14,67. Kondisi tersebut dinamakan campuran miskin sehingga semakin besar nilai AFR, maka semakin kecil konsumsi bahan bakar. Semakin kecil nilai AFR, maka semakin besar konsumsi bahan bakar.

## 4. Kesimpulan

Pada simulasi variasi throttle 15° dan 30°, sistem dengan kontrol PID adaptif dapat mempertahankan AFR pada nilai 14,67. Untuk variasi throttle 50°, 70° dan 90° sistem PID adaptif dapat mempertahankan AFR mendekati 14,67 dengan masing-masing nilai 14,69, 14,7 dan 17,71. Nilai tersebut masih dikategorikan kedalam pembakaran yang sempurna, karena nilai AFR tidak lebih dan kurang 1% dari kondisi stoikiometri. Berdasarkan seluruh uji coba yang telah dilakukan, dapat disimpulkan bahwa kontroler PID adaptif algoritma Dahlin memiliki nilai IAE dan ITAE terkecil dan kinerja yang lebih baik dibandingkan dengan kontroler PID trial and error. PID adaptif memiliki nilai rata-rata IAE dan ITAE dengan nilai 0,4423 dan 0,4699 sedangkan PID memiliki nilai rata-rata IAE dan ITAE sebesar 1,2641 dan 7,3620.

## Referensi

- [1]. A. Ghaffari, A. H. Shamekhi, A. Saki, and E. Kamrani, "Adaptive Fuzzy Control for Air-Fuel Ratio of Automobile Spark Ignition Engine," vol. 2, no. 12, pp. 284–292, 2014.
- [2]. Buntarto, *Dasar-Dasar Sistem Kontrol Pada Kendaraan*. Yogyakarta: Pustaka Baru, 2015.
- [3]. R. Stone and J. Ball, *Automotive engineering fundamentals*. 2004.
- [4]. and S. C. J. Melorose, R. Perroy, *Automobile Mechanical and Electrical Systems*. 2015.
- [5]. M. A. Reactor, A. Triwiyatno, and B. Setiyono, "Desain Kontrol Air to Fuel Ratio (AFR) Pada Model Spark Ignition Engine dengan Metode Hybrid Fuzzy PI," *Transient*, vol. 2, no. 2, pp. 169–175, 2013.
- [6]. A. Triwiyatno, Y. A. Candra, and B. Setiyono, "DESAIN KONTROLER PID ADAPTIF BERBASIS ANFIS UNTUK PENGATURAN RASIO UDARA DAN BAHAN BAKAR (AFR) PADA MODEL MESIN BENSIN 4 LANGKAH," *Transient*, pp. 1–10, 2015.
- [7]. A. Yar and A. I. Bhatti, "Control of Air-to-Fuel ratio of spark ignited engine using super twisting algorithm," *Proc. - 2012 Int. Conf. Emerg. Technol. ICET 2012*, pp. 71–75, 2012.
- [8]. A. Stefanopoulou, "Modeling and Control of Advanced Technology Engines," 1996.
- [9]. M. Rohit and K. Rohit, "Modelling and Simulation of Spark Ignition Engines," vol. 3, no. 5, pp. 55–94, 2014.
- [10]. Y. Bai, "Studies on SI Engine Simulation and Air/Fuel Ratio Control Systems Design," no. September, 2013.
- [11]. Mathworks, *Using Simulink and Stateflow in Automotive Applications*, vol. xx. 1998.
- [12]. "Modeling a Fault-Tolerant Fuel Control System - MATLAB & Simulink Example." [Online]. Available: <https://www.mathworks.com/help/simulink/examples/modeling-a-fault-tolerant-fuel-control-system.html>. [Accessed: 17-Jan-2017].

- [13]. "Modeling Engine Timing Using Triggered Subsystems - MATLAB & Simulink Example." [Online]. Available: <https://www.mathworks.com/help/simulink/examples/modeling-engine-timing-using-triggered-subsystems.html>. [Accessed: 02-Mar-2017].
- [14]. J. Sun, "Engine Control," *Engine Control*, pp. 1-34, 1995.
- [15]. J. M. Vladimír Bobál, Joseph Böhm, Jaromír Fessler, *Digital self-tuning controllers: algorithms, implementation and applications*. 2005.