

## DESAIN KONTROL AERATOR PADA INSTALASI PENGOLAHAN AIR LIMBAH SUWUNG DENGAN FUZZY LOGIC

I Made Mataram

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Udayana  
Kampus Bukit Jimbaran, Bali, 80361  
Email: mataram@ee.unud.ac.id

### Abstrak

Limbah merupakan buangan yang dihasilkan dari suatu proses produksi baik industri maupun domestik (rumah tangga) dan harus dikelola agar tidak menimbulkan pencemaran dan penurunan kualitas lingkungan. Instalasi Pengolahan Air Limbah (IPAL) merupakan suatu tempat pengolahan limbah yang bertempat di daerah Suwung. Pengolahan limbah cair dilakukan dengan menggunakan sistem kolam aerasi dan kolam sedimentasi.

Pada proses aerasi yaitu proses reduksi BOD (*Biological Oxygen Demand*) dan COD (*Chemical Oxygen Demand*) secara aerob digunakan aerator sebagai penghasil oksigen yaitu dengan cara menempatkan aerator di dalam kolam aerasi sehingga menghasilkan oksigen berupa buih udara yang tercampur dengan air. Untuk IPAL Suwung pengoperasian aerator masih dengan cara manual yaitu dioperasikan pada jam tertentu sehingga *input* jumlah oksigen terkadang tidak sesuai dengan karakteristik *input* limbah yang diolah, maka diperlukan suatu sistem kontrol pengoperasian aerator yang dapat menghasilkan oksigen guna mereduksi COD secara tepat sesuai baku mutu limbah

Dalam penelitian ini dilakukan perencanaan desain kontrol pengoperasian aerator dengan *fuzzy logic*. Desain pengontrolan dengan menggunakan logika *fuzzy* pada pengoperasian aerator sudah dapat dibuat dan dapat bekerja sesuai dengan karakteristik *input/output* limbah, ini terlihat dari lama operasi aerator yang bekerja sudah sesuai dengan *input* limbah. Penggunaan energi listrik dengan pengontrolan *fuzzy* pada pengoperasian aerator lebih rendah dibandingkan dengan penggunaan energi listrik pengoperasian secara *manual*, ini terlihat dari penggunaan energi pengoperasian aerator manual dan *fuzzy* pada bulan Oktober 2010 yang memiliki selisih sebesar 6.693 kWh, bulan November 2010 memiliki selisih sebesar 11.275,9 kWh, dan pada bulan Desember 2010 memiliki selisih sebesar 12.429,7 kWh.

Kata kunci : Sistem Kontrol Aerator, IPAL, *Fuzzy Logic*

### 1. PENDAHULUAN

Limbah merupakan buangan yang dihasilkan dari suatu proses produksi baik industri maupun domestik (rumah tangga). Dimana masyarakat bermukim, disanalah berbagai jenis limbah akan dihasilkan. Limbah terdiri dari berbagai macam jenis yaitu limbah padat, gas dan cair. Agar limbah yg merupakan sisa buangan dapat dimanfaatkan kembali maka diperlukan suatu proses yang dapat merubah limbah menjadi suatu hal yang dapat dimanfaatkan. Di pulau Bali yang setiap hari masyarakatnya melakukan berbagai aktivitas yang menghasilkan berbagai macam limbah, maka diperlukan suatu sistem pengolahan limbah agar limbah tersebut tidak mencemari lingkungan.

Instalasi Pengolahan Air Limbah (IPAL) merupakan suatu tempat instalasi pengolahan limbah yang bertempat di daerah Suwung. Pengolahan limbah cair dilakukan dengan menggunakan sistem kolam aerasi dan kolam sedimentasi. Sistem aerasi digunakan dengan maksud untuk mengurangi kebutuhan luas lahan dan meningkatkan proses pengolahan menjadi lebih cepat sekaligus meniadakan bau yang mungkin timbul akibat proses oksidasi yang tidak sempurna.

Pada proses aerasi yaitu proses reduksi BOD (*Biological Oxygen Demand*) dan COD (*Chemical Oxygen Demand*) secara aerob digunakan aerator sebagai penghasil oksigen yaitu dengan cara menempatkan aerator di dalam kolam aerasi sehingga menghasilkan oksigen berupa buih udara yang tercampur dengan air. Untuk IPAL Suwung pengoperasian aerator masih dengan cara manual yaitu dioperasikan pada jam tertentu sehingga *input* jumlah oksigen terkadang tidak sesuai dengan karakteristik *input* limbah yang diolah, maka diperlukan suatu sistem kontrol pengoperasian aerator yang dapat menghasilkan oksigen guna mereduksi COD secara tepat sesuai baku mutu limbah.

Dengan semakin pesatnya perkembangan teknologi semakin banyak peralatan yang memanfaatkan kecerdasan buatan sebagai pengaturan kendalinya. Salah satu kecerdasan buatan adalah *fuzzy logic*. Kelebihan dari *logika fuzzy* adalah kemampuan dalam proses penalaran secara bahasa. Sehingga dalam perancangannya tidak memerlukan persamaan matematik dari objek yang akan dikendalikan. *Fuzzy logic* memiliki langkah sederhana untuk mendapatkan suatu nilai output yaitu fuzzyfikasi, *rules evaluation*, dan defuzzyfikasi. Fuzzyfikasi mengubah *crisp input* menjadi *fuzzy*

input dalam bentuk variabel fuzzy. Rules evaluation bertujuan mengolah fuzzy input berdasarkan if-then rules untuk menentukan keputusan. Bagian ketiga, defuzzifikasi yang bertujuan mengubah fuzzy output yang merupakan hasil rules evaluation menjadi crisp output sebagai suatu nilai akhir. Dengan mengaplikasikan fuzzy logic pada pengaturan aerator di IPAL Suwung diharapkan dapat mengatur pengoperasian aerator agar sesuai dengan input/output limbah yaitu mereduksi COD secara tepat.

Untuk dapat mengaplikasikan fuzzy logic pada pengaturan aerator dalam penelitian ini akan dilakukan suatu pemodelan sistem yang akan digunakan yaitu dengan cara mensimulasikan suatu sistem pengaturan aerator. Sehingga didapat pengetahuan tentang pengaturan aerator yang baik guna mereduksi COD, dapat mengurangi kesalahan ketika akan membuat suatu sistem nyata.

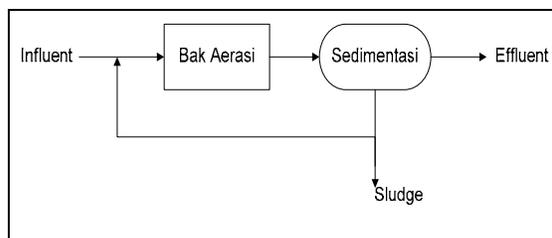
## 2. TINJAUAN PUSTAKA

### 2.1 Instalasi Pengolahan Air Limbah Suwung

Instalasi pengolahan air limbah (IPAL) Suwung merupakan usaha pengolahan limbah cair yang dibangun menggunakan sistem aerated lagoon dengan sedimentation pond guna mengolah air limbah rumah tangga pada daerah sekitar Denpasar, Sanur, dan Kuta.

### 2.2 Sistem Instalasi Pengolahan Air Limbah Suwung

Pengolahan air limbah pada IPAL Suwung menggunakan sistem aerated lagoon dan sedimentation pond. Gambar 2.1 merupakan diagram sistem tersebut.



Gambar 1. Diagram sistem Aerated Lagoon dan Sedimentation Pond

### 2.3 BOD dan COD

Biological Oxygen Demand (BOD) adalah banyaknya oksigen yang dibutuhkan oleh mikroorganisme pada waktu melakukan proses dekomposisi bahan organik yang ada diperairan. Nilai BOD tidak menunjukkan jumlah bahan organik yang sebenarnya, tetapi hanya mengukur secara relatif jumlah oksigen yang dibutuhkan untuk mengoksidasi bahan - bahan buangan tersebut. Jika konsumsi oksigen tinggi yang ditunjukkan dengan semakin kecilnya sisa oksigen terlarut, maka berarti

kandungan bahan - bahan buangan yang membutuhkan oksigen tinggi.

Sedangkan COD atau Chemical Oxygen Demand adalah jumlah oksigen yang diperlukan untuk mengurai seluruh bahan organik yang terkandung dalam air (Boyd, 1990). Hal ini karena bahan organik yang ada sengaja diurai secara kimia dengan menggunakan oksidator kuat kalium bikromat pada kondisi asam dan panas dengan katalisator perak sulfat (Boyd, 1990; Metcalf & Eddy, 1991), sehingga segala macam bahan organik, baik yang mudah urai maupun yang kompleks dan sulit urai, akan teroksidasi. Dengan demikian, selisih nilai antara COD dan BOD memberikan gambaran besarnya bahan organik yang sulit urai yang ada di perairan. Bisa saja nilai BOD sama dengan COD, tetapi BOD tidak bisa lebih besar dari COD. Jadi COD menggambarkan jumlah total bahan organik yang ada.

### 2.4 Logika Fuzzy

Pada tahun 1965, Lotfi A. Zadeh memodifikasi teori himpunan, dimana setiap anggotanya memiliki derajat keanggotaan yang bernilai kontinyu antara 0 sampai 1. Himpunan ini disebut dengan Himpunan Kabur (Fuzzy Set). Ada beberapa alasan digunakannya logika fuzzy, antara lain :

- Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran fuzzy sangat sederhana dan mudah dimengerti.
- Logika fuzzy sangat fleksibel.
- Logika fuzzy memiliki toleransi terhadap data-data yang tidak tepat.
- Logika fuzzy mampu memodelkan fungsi-fungsi nonlinier yang sangat kompleks.
- Logika fuzzy dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
- Logika fuzzy dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
- Logika fuzzy didasarkan pada bahasa alami.

### 2.5 Himpunan Fuzzy

Pada himpunan tegas (crisp), nilai keanggotaan suatu item  $x$  dalam suatu himpunan  $A$ , yang sering ditulis dengan  $\mu_A[x]$ , memiliki 2 kemungkinan, yaitu :

- Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
- Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Himpunan fuzzy didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan real pada interval  $[0,1]$ . Nilai keanggotaannya menunjukkan bahwa suatu item dalam semesta pembicaraan tidak hanya berada pada 0 atau 1, namun juga nilai yang terletak diantaranya.

Dengan kata lain, nilai kebenaran suatu item tidak hanya bernilai benar atau salah. Nilai 0 menunjukkan salah, nilai 1 menunjukkan benar, dan masih ada nilai-nilai yang terletak antara benar dan salah (Kusumadewi, 2004). Himpunan *fuzzy* memiliki 2 atribut, yaitu :

- a. Linguistik, yaitu penamaan suatu grup yang mewakili satu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: NORMAL, HANGAT, PANAS.
- b. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari satu variabel seperti: 24, 32, 40, dsb.

## 2.6 Domain Himpunan Fuzzy

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan riil yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif.

Contoh domain himpunan *fuzzy*:

- DINGIN = [0, 20]
- SEJUK = [15, 25]
- NORMAL = [20, 30]
- HANGAT = [25, 35]
- PANAS = [30, 40]

## 2.7 Variabel Fuzzy

Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*, seperti: temperatur, persediaan, permintaan, dsb.

## 2.8 Semesta Pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan riil yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh:

- Semesta pembicaraan untuk variabel umur:  $[0 + \infty]$
- Semesta pembicaraan untuk variabel temperatur:  $[0 \ 40]$

## 2.9 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan (fuzzifikasi) adalah dengan melalui pendekatan fungsi.

## 2.10 Aturan Fuzzy (Fuzzy Rule)

Aturan *fuzzy* memegang peranan yang terpenting dalam sistem *fuzzy*, bisa dibilang aturan *fuzzy* adalah otak dari sistem *fuzzy*. Dengan aturan *fuzzy* inilah segala pengetahuan dan pengalaman manusia dapat direpresentasikan melalui sebuah bahasa alami yang disebut aturan *IF-THEN* seperti berikut ini (Wang, 1997).

IF x is A THEN y is B

Dengan x dan y adalah sebuah variabel, serta A dan B adalah himpunan fuzzy. Pernyataan yang mengikuti IF disebut sebagai *antecedent*, sedangkan pernyataan yang mengikuti THEN disebut sebagai *consequent*. Proposisi ini dapat diperluas dengan menggunakan operator *fuzzy*, misalnya AND atau OR ( $\circ$ ), seperti berikut ini:

IF  $(x_1 \text{ is } A_1) \circ (x_2 \text{ is } A_2) \circ (x_3 \text{ is } A_3) \circ \dots \circ (x_N \text{ is } A_N)$  THEN y is B

## 2.11 Fuzzy Inference System (FIS)

*Fuzzy Inference System (FIS)* adalah proses pemetaan variabel *input* ke variabel *output* dengan menggunakan logika *fuzzy*. Struktur dari *FIS* terdiri dari variabel *input*, aturan *fuzzy*, variabel *output* dan *defuzzification* (penegasan). Dalam metode *FIS* kita menggunakan pengetahuan dan pengalaman berdasarkan kumpulan fakta untuk mengambil atau menduga suatu kesimpulan atau keputusan yang beralasan.

Proses *fuzzy inference system* terdiri dari beberapa langkah-langkah seperti terlihat pada gambar berikut :



Gambar 2. Blok diagram *Fuzzy Inference System*

### 2.12 Fuzzifikasi (*Fuzzification*)

Fuzzifikasi adalah membuat fungsi keanggotaan masing-masing variabel *fuzzy* serta membangkitkan nilai keanggotaan *input* tegas, dengan cara memetakan *input* tegas yang ada pada domain himpunan *fuzzy* melalui fungsi keanggotaannya menuju ke nilai keanggotaannya. Sehingga didapatkan nilai keanggotaan antara interval 0-1.

### 2.13 Operator *Fuzzy*

Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasi dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering dikenal dengan nama *fire strength* atau  $\alpha$ -predikat. Ada 3 operator dasar yang diciptakan oleh Zadeh, yaitu (Kusumadewi, 2004):

#### a. Operator AND

Operator ini berhubungan dengan operasi interaksi pada himpunan.  $\alpha$ -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$$

#### b. Operator OR

Operator ini berhubungan dengan operasi *union* pada himpunan.  $\alpha$ -predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

#### c. Operator NOT

Operator ini berhubungan dengan operasi komplemen pada himpunan.  $\alpha$ -predikat sebagai hasil operasi dengan operator NOT diperoleh dengan mengurangi nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1.

$$\mu_{A^c} = 1 - \mu_A[x]$$

### 2.14 Fungsi Implikasi (*Implication*)

Fungsi implikasi didefinisikan sebagai pemotongan *consequent fuzzy* berdasarkan *antecedent* yang telah dikombinasi dengan operator *fuzzy*. *Output* dari fungsi implikasi ini adalah sebuah himpunan *fuzzy*. Fungsi implikasi ini diterapkan pada masing-masing aturan *fuzzy*.

### 2.15 Komposisi Aturan (*Aggregation*)

Komposisi aturan adalah menggabungkan semua output (himpunan *fuzzy*) hasil implikasi masing-masing aturan secara paralel, kemudian dikombinasikan menjadi satu himpunan *fuzzy*.

Salah satu metode yang dapat digunakan dalam komposisi aturan adalah metode *Max* (*maximum*).

Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah *fuzzy*, dan mengaplikasikannya ke *output* dengan menggunakan operator OR (*union*). Jika semua proposisi telah dievaluasi, maka *output* akan berisi suatu himpunan *fuzzy* yang merefleksikan kontribusi dari tiap-tiap proposisi (Kusumadewi, 2004).

### 2.16 Defuzzifikasi

Proses defuzzifikasi bertujuan untuk mengubah *output* komposisi aturan menjadi sebuah nilai tegas (*crisp*). *Input* dari proses defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Jika diberikan suatu himpunan *fuzzy* dengan *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output*.

## 3. METODE

### 3.1 Tempat dan Waktu Penelitian

Penelitian tugas akhir ini dilakukan di Instalasi Pengolahan Air limbah (IPAL) Suwung, Denpasar Bali, dengan waktu penelitian yang dimulai pada bulan Oktober 2010.

### 3.2 Data

#### 3.2.1 Sumber data

Data yang digunakan dalam penelitian ini didapatkan dari Instalasi Pengolahan Air Limbah (IPAL) Suwung.

#### 3.2.2 Jenis data

Data-data yang dipakai dalam penyusunan tugas akhir ini berupa data sekunder yang diperoleh dari Instalasi Pengolahan Air limbah (IPAL) Suwung antara lain data harian COD, volume, jumlah aerator, standar baku mutu limbah, spesifikasi aerator, denah bangunan lengkap, diagram segaris sistem tenaga listrik lengkap dengan penjelasannya, dan data-data lain yang nantinya dianggap perlu.

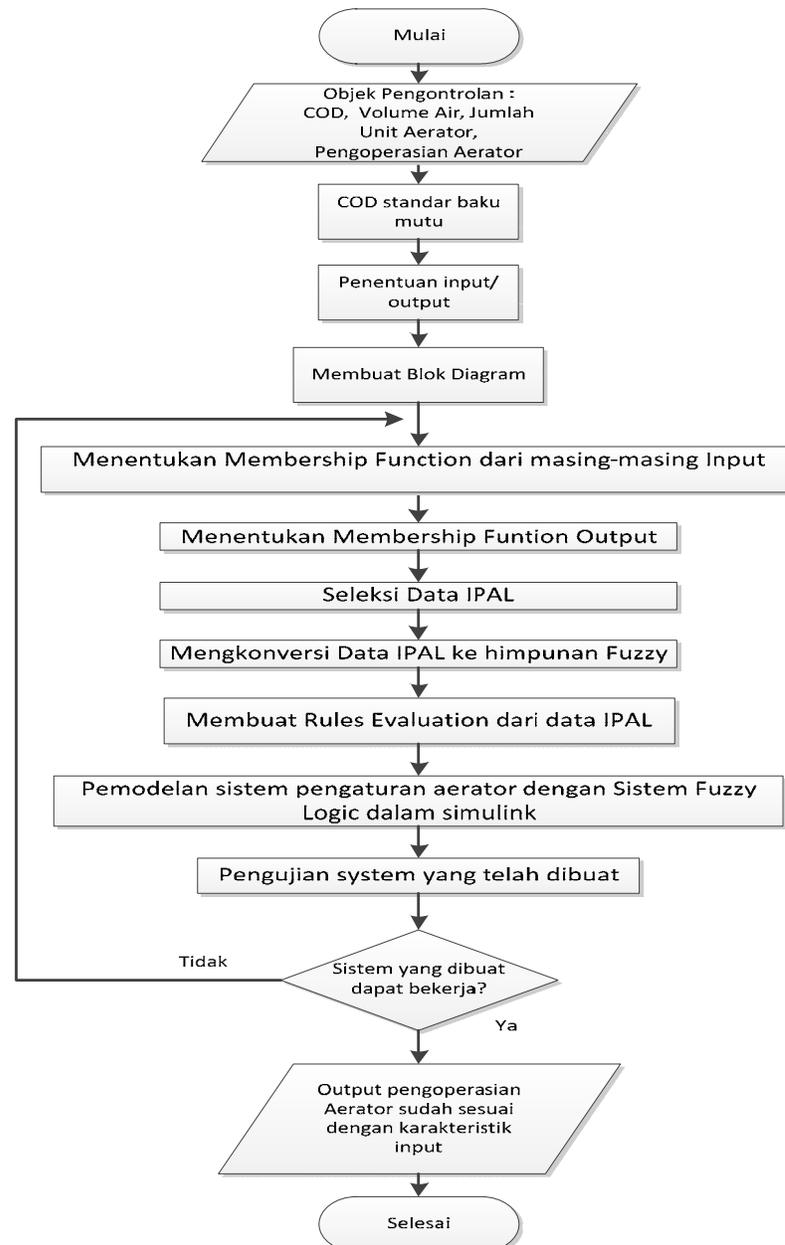
### 3.3 Analisis Data

Metode yang digunakan dalam analisis hasil penelitian ini adalah dengan langkah-langkah sebagai berikut :

1. Menentukan Objek pengontrolan.
2. Menentukan *input/output* pengontrolan.
3. Membuat blok diagram
4. Menentukan *Membership Function* dari masing-masing input.
5. Menentukan *Membership Function* dari output yang diinginkan
6. Menentukan *Rule Evaluation* dari semua input.
7. Pemodelan sistem pengaturan aerator dengan Sistem *Fuzzy Logic* dalam *simulink*

### 3.4 Alur Analisis

Analisis dilakukan sesuai yang tergambar dalam bagan alir pada gambar 3 :



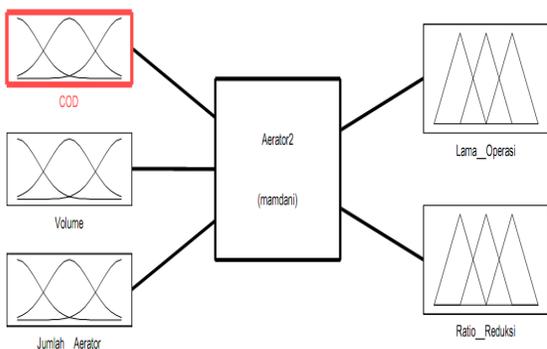
Gambar 3. Alur analisis

## 4. HASIL DAN PEMBAHASAN

### 4.1 Pemodelan Fuzzy

Untuk dapat memodelkan kebutuhan oksigen yang diperlukan pada kolam aerasi maka data lapangan yang diperlukan adalah data COD, jumlah aerator yang beroperasi dan Volume kolam sebagai parameter *input* sehingga dapat dihitung *output* berapa lama aerator bekerja agar menghasilkan oksigen sesuai dengan kebutuhan guna mereduksi

COD agar sesuai dengan nilai ambang batasnya yaitu maksimum 100 mg/l. Selain lama operasi, *ratio* reduksi juga di hitung pada sisi *output* yang nantinya akan digunakan untuk mencari besar nilai COD hasil fuzzifikasi. *Ratio* reduksi merupakan perbandingan antara COD input dengan besar reduksi COD (selisih COD). Sehingga dengan parameter tersebut dapat disusun fungsi keanggotaan dari *input/output* serta *rule evaluation* dari sistem *fuzzy*.



Gambar 4. Pemodelan Fuzzy Pengoperasian Aerator

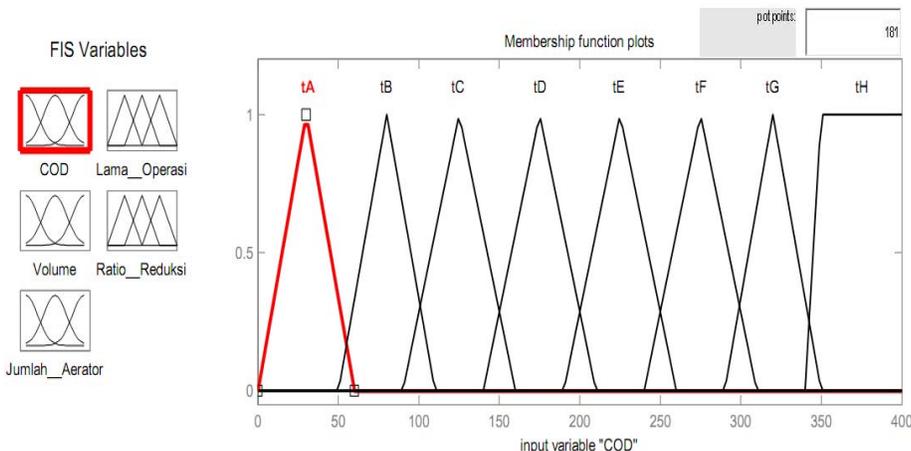
Parameter BOD tidak digunakan sebagai input karena dengan parameter COD sudah mewakili kebutuhan oksigen yang dibutuhkan, hal ini didasari pada pembuktian pakar bahwa

$$COD = 1,64 \times BOD + 11,36 \text{ (Ademoroti, 1986)}$$

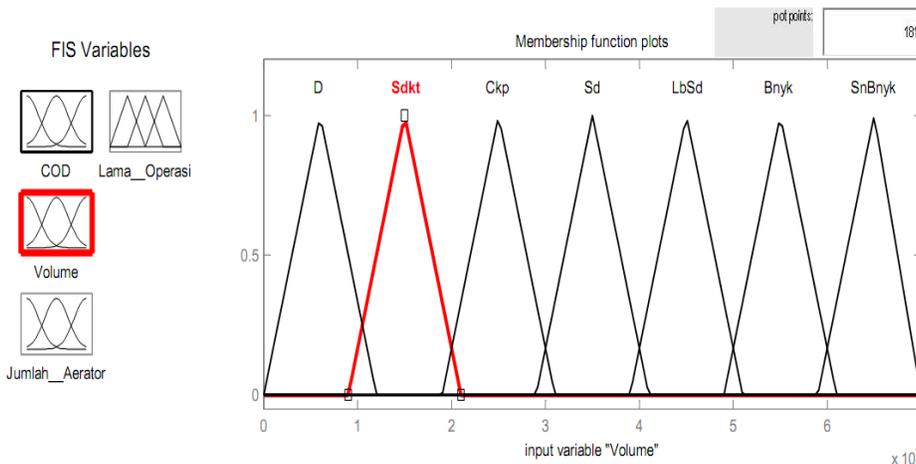
4.1.1 Penentuan Fungsi Keanggotaan

Untuk fungsi keanggotaan dilakukan dengan menentukan nilai maksimum dan minimum dari data yang ada, kemudian membagi dan mengelompokkan menjadi beberapa kelompok yang memiliki parameter tertentu. Sisi input yang berupa COD, Volume dan Jumlah Aerator serta sisi output berupa Lama Operasi dari aerator dan Ratio Reduksi akan dikelompokkan sesuai dengan parameternya masing-masing.

Untuk input COD yang digunakan adalah fungsi keanggotaan berupa representasi segitiga karena lebih mudah diterima logika manusia. Input COD dibagi menjadi 8 kuantisasi sesuai data lapangan dari nilai terkecil dan terbesar yang dibagi menjadi range - range nilai tertentu yaitu seperti gambar 5.



Gambar 5. Fungsi keanggotaan Input COD



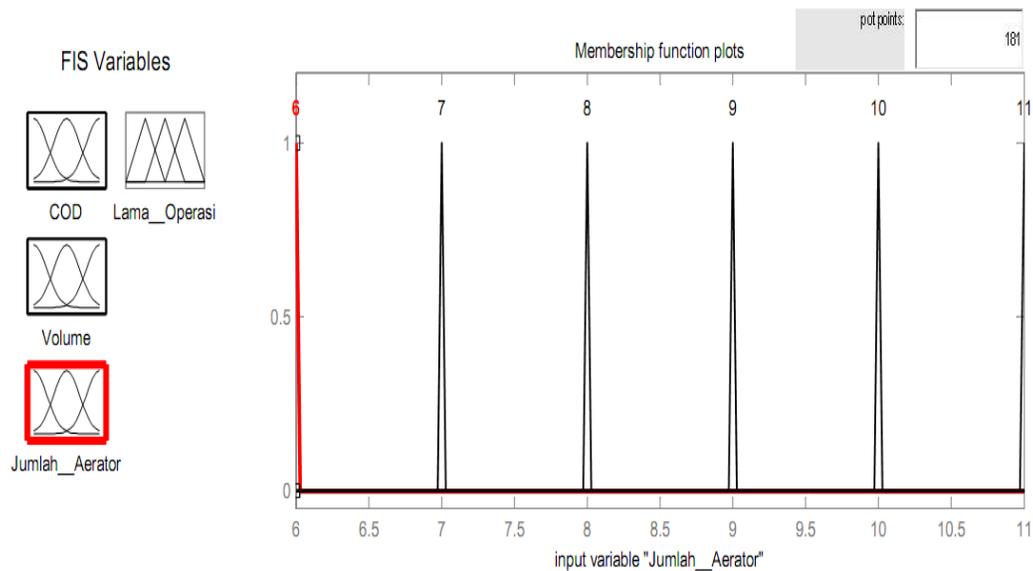
Gambar 6. Fungsi keanggotaan Input volume

Untuk *input* volume ( $m^3$ ) yang digunakan adalah fungsi keanggotaan berupa representasi segitiga dengan jumlah kuantisasi sebanyak 7 kuantisasi sesuai data terkecil dan terbesar yang dibagi menjadi range-range nilai tertentu yaitu seperti gambar 6.

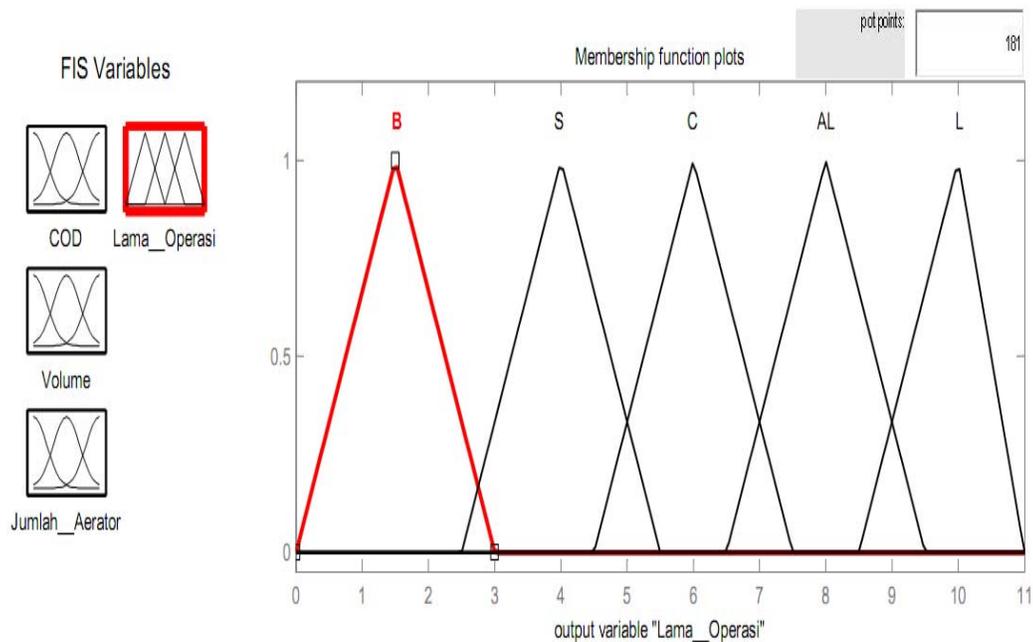
Untuk *input* jumlah unit aerator yang beroperasi (unit) adalah fungsi keanggotaan berupa representasi segitiga yang dibagi menjadi 6 kuantisasi yaitu seperti gambar 7.

Pada sisi *output* yaitu berupa waktu kerja (jam) aerator yang digunakan adalah fungsi keanggotaan berupa representasi segitiga dengan kuantisasi sebanyak 6 kuantisasi yaitu seperti gambar 8.

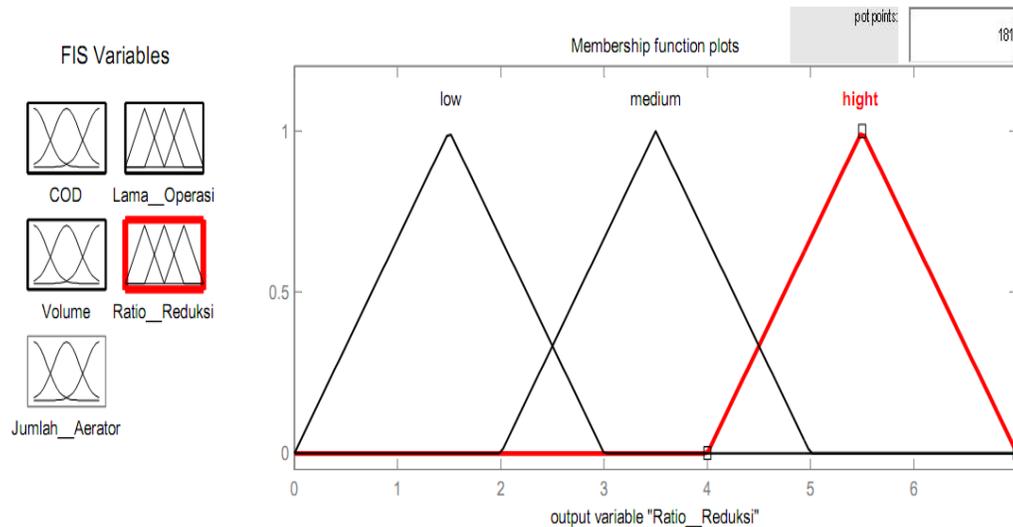
Sisi *output ratio* reduksi yang digunakan adalah fungsi keanggotaan berupa representasi segitiga dengan kuantisasi sebanyak 3 kuantisasi yaitu seperti gambar 9.



Gambar 7. Fungsi keanggotaan *Input* Jumlah Aerator



Gambar 8. Fungsi keanggotaan *Output* lama operasi



Gambar 9. Fungsi keanggotaan Output Ratio reduksi

#### 4.1.2 Penentuan Rule Evaluation

Setelah proses fuzzifikasi input dan output, tahap selanjutnya adalah membuat *rule evaluation* yang menunjukkan hubungan antara ketiga parameter *input* dan parameter *output* yang dinamakan *interfacing*. *Rule evaluation* diperoleh dari perhitungan serta penalaran terhadap data IPAL dari bulan Oktober-Desember 2010 berupa Volume harian, COD<sub>in</sub>, COD<sub>out</sub>, jumlah aerator yang beroperasi seperti terlihat pada lampiran A-C, kemudian diurutkan berdasarkan jumlah aerator yang digunakan.

Pada proses *interfacing* ini hubungan antara ketiga variabel input dan variabel output dapat diatur yaitu dengan menghubungkan ketiga variabel *input* dan variabel *output* menjadi IF-THEN seperti berikut ini :

1. If (COD is tB) and (Volume is D) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is AL)(Ratio\_Reduksi is medium)(1)
2. If (COD is tC) and (Volume is D) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
3. If (COD is tD) and (Volume is D) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is medium)(1)
4. If (COD is tA) and (Volume is Sdkt) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is S)(Ratio\_Reduksi is medium)(1)
5. If (COD is tB) and (Volume is Sdkt) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is C)(Ratio\_Reduksi is low)(1)
6. If (COD is tC) and (Volume is Sdkt) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is AL)(Ratio\_Reduksi is low)(1)
7. If (COD is tD) and (Volume is Sdkt) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)

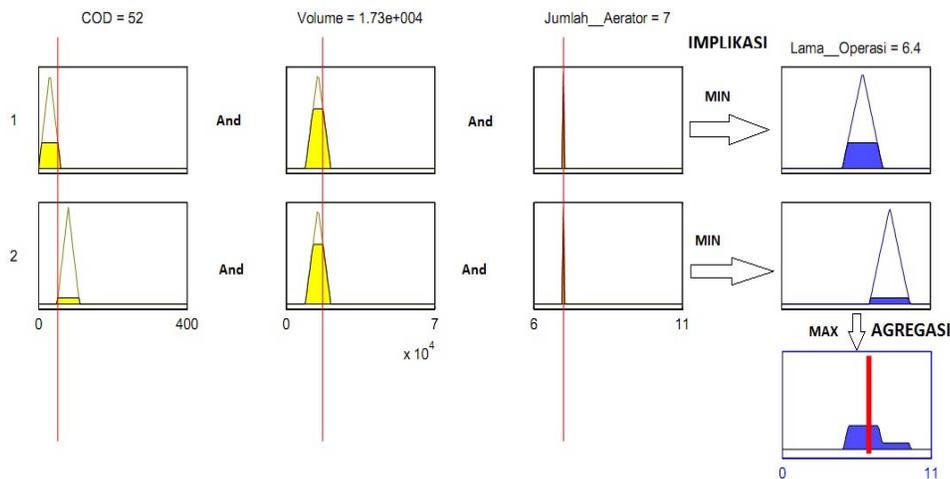
8. If (COD is tB) and (Volume is Ckp) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is C)(Ratio\_Reduksi is medium)(1)
9. If (COD is tE) and (Volume is Ckp) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
10. If (COD is tA) and (Volume is Sd) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is C)(Ratio\_Reduksi is high)(1)
11. If (COD is tB) and (Volume is Sd) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is C)(Ratio\_Reduksi is low)(1)
12. If(COD is tE) and (Volume is Sd) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
13. If (COD is tC) and (Volume is Bnyk) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is L)(Ratio\_Reduksi is high)(1)
14. (COD is tB) and (Volume is SnBnyk) and (Jumlah\_Aerator is 6) then (Lama\_Operasi is C)(Ratio\_Reduksi is low)(1)
15. If(COD is tB) and (Volume is D) and (Jumlah\_Aerator is 7 ) then (Lama\_Operasi is AL)(Ratio\_Reduksi is medium)(1)
16. If(COD is tC) and(Volume is D) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
17. If(COD is tE) and (Volume is D) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
18. If(COD is tA) and (Volume is Sdkt) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is C)(Ratio\_Reduksi is medium)(1)
19. If (COD is tB) and (Volume is Sdkt) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is AL)(Ratio\_Reduksi is medium)(1)

- 20. If(COD is tC) and( Volume is Sdkt) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
- 21. If (COD is tF) and (Volume is Sdkt) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is low) (1)
- 22. If (COD is tC) and (Volume is Ckp) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is medium) (1)
- 23. If (COD is tD) and (Volume is Ckp) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is L)(Ratio\_Reduksi is low) (1)
- 24. If (COD is tC) and (Volume is LbSd) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is C)(Ratio\_Reduksi is low) (1)
- 25. If (COD is tA) and (Volume is Bnyk) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is S)(Ratio\_Reduksi is low) (1)
- 26. If(COD is tB) and (Volume is SnBnyk) and (Jumlah\_Aerator is 7) then (Lama\_Operasi is S)(Ratio\_Reduksi is low) (1)
- 27. If (COD is tB) and (Volume is D) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is C)(Ratio\_Reduksi is medium) (1)
- 28. If(COD is tC) and (Volume is D) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is L)(Ratio\_Reduksi is low)(1)
- 29. If (COD is tH) and (Volume is D) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is L)(Ratio\_Reduksi is low) (1)
- 30. If(COD is tA) and (Volume is Sdkt) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is C)(Ratio\_Reduksi is high)(1)
- 31. If (COD is tB) and (Volume is Sdkt) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is AL)(Ratio\_Reduksi is high) (1)
- 32. If(COD is tC) and (Volume is Sdkt) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is L)(Ratio\_Reduksi is medium)(1)

- 33. If(COD is tD) and (Volume is Sdkt)and(Jumlah\_Aerator is 8)then(Lama\_Operasi is L)(Ratio\_Reduksi is low) (1)
- 34. If(COD is tB) and (Volume is Ckp) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is C)(Ratio\_Reduksi is medium) (1)
- 35. If (COD is tE) and (Volume is Ckp) and (Jumlah\_Aerator is 8) then (Lama\_Operasi is L)(Ratio\_Reduksi is low) (1)
- 36. If(COD is tB)and(Volume is Sd)and(Jumlah\_Aerator is 8)then (Lama\_Operasi is C)(Ratio\_Reduksi is medium) (1)
- 37. If (COD is tC) and (Volume is D) and (Jumlah\_Aerator is 9) then (Lama\_Operasi is AL)(Ratio\_Reduksi is low) (1)
- 38. If (COD is tC) and (Volume is Sdkt) and (Jumlah\_Aerator is 9) then (Lama\_Operasi is L)(Ratio\_Reduksi is medium) (1)

**4.1.3 Fuzzy Inference**

Fuzzy inference merupakan proses dalam menalar nilai masukan guna penentuan nilai keluaran sebagai bentuk pengambil keputusan yang terdiri dari proses implikasi dan agregasi. Model penalaran yang digunakan adalah penalaran implikasi min - agregasi max. Dalam penalaran max-min proses pertama yang dilakukan adalah melakukan operasi implikasi min sinyal keluaran lapisan fuzzyfikasi. Setelah keluaran IF-THEN rule ditentukan pada tahap implikasi maka tahap selanjutnya adalah melakukan proses agregasi, yaitu proses mengkombinasikan semua keluaran menjadi sebuah fuzzy set tunggal. Untuk mencari nilai keluaran yang selanjutnya akan difuzzifikasikan sebagai bentuk keluaran pengontrol operasional max-min tersebut.



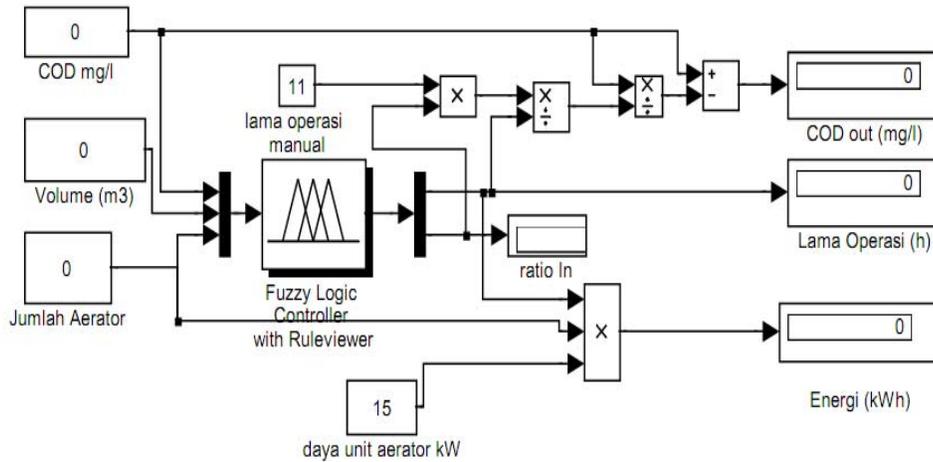
Gambar 10. Proses implikasi – agregasi

**4.1.4 Defuzzifikasi**

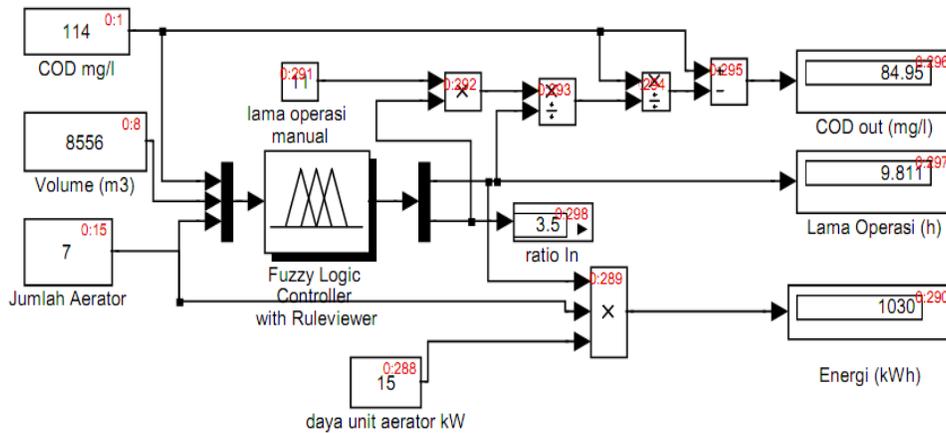
Jika masukan dari fuzzifikasi adalah sebuah bilangan tunggal, yaitu harga variabel masukan dan keluarannya adalah derajat keanggotaan dalam suatu *fuzzy set*, maka masukan defuzzifikasi adalah sebuah *fuzzy set* yaitu hasil dari proses agregasi dan keluarannya adalah sebuah bilangan tunggal untuk diisikan ke sebuah variabel keluaran FIS. Metode yang digunakan adalah metode CoA (*Centre of Area*). Keluaran dari proses implikasi dan agregasi akan menjadi *input* pada Defuzzifikasi sehingga menghasilkan output lama operasi.

**4.2 Simulasi Pengoperasian Aerator *Fuzzy Logic* dengan Simulink**

Simulasi dilakukan dengan memasukkan *nilai* variabel *input* yaitu berupa COD, Volume serta jumlah Unit Aerator kemudian jalankan (*running*) simulasi tersebut sehingga diperoleh *output* berupa Lama Operasi aerator, COD out serta Energi Listrik yang digunakan oleh aeraor.



**Gambar 11. Blok Simulink Pengontrolan Lama Operasi Aerator**



**Gambar 12. Hasil *running* 1 simulasi**

Pada simulasi yang dilakukan yaitu dengan *input* 114 mg/l COD, 8556 m<sup>3</sup> volume serta 7 unit aerator diperoleh lama operasi yaitu selama 9.811 jam dan memerlukan energi sebesar 1030 kWh.

**4.3 Perhitungan Energi**

Aerator yang digunakan pada IPAL Suwung menggunakan daya sebesar 15kW untuk setiap unitnya dan dioperasikan manual selama 11 jam setiap harinya. Jumlah unit yang dioperasikan tergantung dari unit yang layak operasi dan yang

mengalami gangguan. Energi yang digunakan dapat dihitung dengan formula :

$$E \text{ (kWh)} = 15 \text{ (kW)} \times 11 \text{ (h)} \times n \text{ (unit)}$$

Sehingga jumlah energi yang digunakan tiap bulannya tergantung dari unit yang beroperasi.

Pada Bulan Oktober 2010 jumlah aerator yang mengalami gangguan adalah rata-rata berjumlah 4 unit sehingga, hanya 7 unit yang dapat dioperasikan. Dengan membandingkan antara penggunaan energi pengoperasian aerator manual dengan pengoperasian aerator *fuzzy logic* diperoleh penggunaan energi menggunakan *fuzzy logic* lebih rendah dibandingkan dengan menggunakan pengoperasian *manual* yaitu sebesar 6.963 kWh.

Pada Bulan November 2010 jumlah aerator yang mengalami gangguan adalah rata-rata berjumlah 4 unit sehingga dari 11 unit aerator yang tersedia, hanya 7 unit yang dapat dioperasikan. Dengan membandingkan antara penggunaan energi pengoperasian aerator manual dengan pengoperasian aerator *fuzzy logic* diperoleh penggunaan energi menggunakan *fuzzy logic* lebih rendah dibandingkan dengan menggunakan pengoperasian *manual* yaitu sebesar 11.275,9 kWh.

Pada Bulan Desember 2010 jumlah aerator yang mengalami gangguan adalah rata-rata berjumlah 4 unit sehingga dari 11 unit aerator yang tersedia, hanya 7 unit yang dapat dioperasikan. Dengan membandingkan antara penggunaan energi pengoperasian aerator manual dengan pengoperasian aerator *fuzzy logic* diperoleh penggunaan energi menggunakan *fuzzy logic* lebih rendah dibandingkan dengan menggunakan pengoperasian *manual* yaitu sebesar 12.429,7 kWh.

## 5. SIMPULAN

Dari pembahasan di atas, maka dapat ditarik beberapa simpulan sebagai berikut :

1. Desain pengontrolan dengan menggunakan logika *fuzzy* pada pengoperasian aerator sudah dapat dibuat dan dapat bekerja sesuai dengan karakteristik *input/output* limbah, ini terlihat dari lama operasi aerator yang bekerja sudah sesuai dengan *input* limbah.
2. Penggunaan energi listrik dengan pengontrolan *fuzzy* pada pengoperasian aerator lebih rendah dibandingkan dengan penggunaan energi listrik pengoperasian secara *manual*, ini terlihat dari penggunaan energi pengoperasian aerator manual dan *fuzzy* pada bulan Oktober 2010 yang memiliki selisih sebesar 6.693 kWh, bulan November 2010 memiliki selisih sebesar 11.275,9 kWh, dan pada bulan Desember 2010 memiliki selisih sebesar 12.429,7 kWh.

## 6. DAFTAR PUSTAKA

- [1] Abdullah, S. 2007. *Operasi dan Pemeliharaan IPAL*. ----
- [2] Ellis, G. 2004. *Control System Design Guide A Practical Guide*. California : Elsevier Academic Press.
- [3] Gray, N. F. 2004. *Biology of Wastewater Treatment*. London : Imperial College Press.
- [4] Hunt, B. R. 2001. *A Guide Matlab for Beginners and Experienced Users*. New York : Cambridge University Press.
- [5] IPAL Suwung, 2010. *Presentasi Denpasar Sewerage Development Project*. ----
- [6] Kemmer, F. N. 1988. *The NALCO Water Handbook*. USA : McGraw-Hill
- [7] Keputusan Menteri Negara Lingkungan Hidup Nomor 112 Tahun 2003 tentang Baku Mutu Air Limbah Domestik.
- [8] Kusumadewi, S. 2004. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta : Graha Ilmu
- [9] Naba, A. 2009. *Belajar Cepat Fuzzy Logic Menggunakan Matlab*. Yogyakarta : Penerbit ANDI.
- [10] Ogata, K. 1997. *Modern Control Engineering*. USA : Viacom Company
- [11] Robandi, I. 2009. *Modern Power System Control Desain, Analisis, & Solusi Kontrol Tenaga Listrik*. Yogyakarta : Penerbit ANDI.
- [12] Sivanandam, S. N. 2007. *Introduction to Fuzzy Logic using Matlab*. Berlin : Springer.
- [13] Sutrisno, I. 2009. *Pemrograman Komputer Dengan Software Matlab*. Surabaya : ITS Press.
- [14] Tchobanoglous, G. 2003. *Wastewater Engineering Treatment and Reuse*. Taiwan : McGraw Hill Companies, Inc.
- [15] Wahid. 2001. *Diktat Kendali*. Teknik Kimia Universitas Indonesia.