

## BAB II

### LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Adapun beberapa jurnal penelitian sejenis yang telah ada sebelumnya yang menjadi bahan penyusunan tugas akhir ini adalah:

1. M.P. Nagarkar, G.J. Vikhe, K.R. Borole dan V.M. Nandedkar (2011) pada *International Journal of Automotive and Mechanical Engineering (IJAME)* tentang “*Active Control of Quarter-Car Suspension System Using Linear Quadratic Regulator*” yang di dalamnya menyatakan :  
Penelitian ini menganalisis sistem suspensi aktif dan pasif menggunakan kendali LQR. Sebuah model mobil seperempat linier digunakan untuk analisis dan simulasi. Penggunaan kendali LQR dibandingkan dengan sistem suspensi pasif. Penelitian ini diujikan dengan diberikan sebuah masukan *step, white noise, bump, triangular bump*, data kondisi jalan kota dan jalan raya. Penelitian ini juga membahas respon keluaran defleksi kendaraan, kecepatan dan percepatan.
2. Tinnavelli Ramamohan Rao dan Punjala Anusha (2013) pada *International Conference on Control, Automation, Robotics and Embedded Systems (CARE)* tentang “*Active Suspension System of 3 DOF Quarter Car Using Fuzzy Logic Control for Ride Comfort*” yang di dalamnya menyatakan :  
Penelitian ini menganalisis sistem suspensi aktif dan pasif menggunakan model 3 DOF dan kendali *fuzzy logic*. Di dalamnya juga menerangkan hasil kendali dengan *fuzzy logic* dengan *output* berupa defleksi bodi, defleksi ban, dan defleksi *seat*, lalu percepatan bodi, percepatan ban, dan percepatan *seat*. Gangguan disimulasikan dengan sinyal *step*.
3. Abdel-Nasser Sharkawy, Ahmed Saad Ali, Nouby M. Ghazaly dan Gamal abdel-Jaber (2015) pada jurnal *International Journal of Advances in Engineering & Technology (IJAET)* tentang “*PID Controller of Active Suspension System for Quarter Car Model*” yang di dalamnya menyatakan tentang : Penelitian ini dirancang menggunakan pengendali PID. Analisis diberikan terhadap kecepatan kendaraan, percepatan kendaraan, defleksi

suspensi dan *whell travel*. Gangguan disimulasikan menggunakan sinyal sinusoidal dan random.

4. Rudy S.Wahjudi (2016) pada Jurnal Ilmiah Teknik Elektro (JETri) tentang “Perancangan Kendali Suspensi Aktif “ yang di dalamnya menyatakan :  
Sistem yang dianalisis adalah model seperempat kendaraan pasif dan aktif. Sistem pasif dimodelkan tanpa pengendali, sedangkan sistem aktif menggunakan kendali PID. Kemudian hasil analisis diperlihatkan bagaimana *output* dari sistem suspensi aktif dan pasif terhadap berbagai kondisi permukaan jalan yang berupa fungsi periodik.
5. Lei Cheng (2018) pada Jurnal *International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS)* tentang “*Fuzzy controller design for vehicle active suspension based on genetic algorithm*” yang di dalamnya menyatakan :  
Penelitian ini dirancang dengan pendekatan metode logika fuzzy dan dioptimalkan berdasarkan algoritma genetika. Pengujian dilakukan pada percepatan, defleksi, dan defleksi ban.

Penelitian yang akan dilakukan adalah mengenai sistem suspensi aktif mobil penumpang dengan model seperempat. Penelitian dilakukan dengan menggunakan dua jenis pengendali yang berbeda, yaitu kendali logika fuzzy dan *Linear Quadratic Regulator (LQR)*. Dengan melakukan sinyal uji masukan berupa sinyal step atau konstan dan gangguan pada jalan yang disimulasikan dengan berbagai sinyal. Kemudian hasil dari penelitian ini, akan dilakukan pengamatan *output* dari defleksi kendaraan dengan dua jenis pengendali tersebut dan akan dianalisis respon yang terjadi dari kedua jenis pengendali tersebut.

## 2.2 Sistem Suspensi

Pada umumnya sistem suspensi terletak di antara roda dan badan kendaraan. Sistem suspensi merupakan pemisah antara roda dan frame kendaraan dan bodi. Komponen dalam sistem suspensi terdiri dari:

- a. Pegas (*spring*). Pegas berfungsi menopang berat kendaraan maupun bebannya termasuk beban kejut terhadap kontur permukaan jalan.

- b. Peredam kejut (*shock absorber*). Peredam kejut berfungsi menyerap atau menghilangkan beban kejut atau getaran yang terjadi dan mencegah kendaraan berosilasi secara vertikal.

Tujuan dan fungsi pemakaian sistem suspensi adalah :

1. Meningkatkan kenyamanan dan stabilitas dalam berkendara

Fungsi sistem suspensi yang pertama adalah untuk meningkatkan kenyamanan dan stabilitas dalam berkendara. Hal ini bisa dicapai karena sistem suspensi bekerja untuk menyerap oksilasi, getaran, dan kejutan dari permukaan jalan agar tidak diteruskan ke bodi kendaraan.

Ada empat macam Gerakan oksilasi yang terjadi pada kendaraan yaitu:

- *Pitching* : Gerakan oksilasi naik turun pada bagian depan dan belakang kendaraan terhadap titik tengah kendaraan dilihat dari samping.
- *Rolling* : Gerakan bodi kendaraan miring ke salah satu sisi kendaraan, kanan atau kiri.
- *Bouncing* : Gerakan naik turun kendaraan secara vertikal
- *Yawing* : Gerakan bodi kendaraan kearah kanan atau kiri terhadap titik tengah kendaraan dilihat dari atas kendaraan.

Dengan adanya sistem suspensi, maka pada saat kendaraan melaju di atas jalan yang tidak rata dan bergelombang, komponenn suspensi seperti pegas akan menyerap kejutan dari permukaan jalan dan getaran dari roda-roda agar tidak diteruskan ke bodi kendaraan. Sedangkan komponen *shock absorber* berfungsi untuk meredam getaran oksilasi naik turun yang terjadi pada pegas.

2. Menghubungkan bodi kendaraan dengan roda

Menghubungkan roda dengan bodi kendaraan dan memberikan peluang terhadap kendaraan untuk berosilasi vertikal terhadap gangguan jalan agar memberikan kenyamanan terhadap penumpang.

3. Meningkatkan kemampuan dan daya cengkram roda terhadap jalan

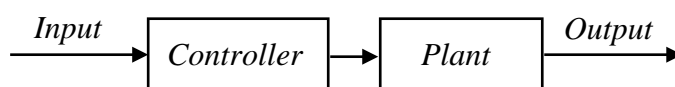
Dengan suspensi, bodi kendaraan dan komponen suspensi lainnya akan memberikan dorongan dan tekanan pada roda agar selalu kembali menempel pada jalan.

4. Menopang bodi kendaraan dan menjaga letak geometris antara bodi dan roda  
Sebagai sebuah sistem, suspensi dirancang sedemikian rupa agar dapat menopang bodi kendaraan namun tetap memberikan peluang kepada roda untuk dapat mengikuti kontur jalan. Meskipun roda bergerak bebas, sistem suspensi juga tetap menjaga letak geometris roda terhadap bodi kendaraan.
5. Memindahkan gaya gerak dan gaya pengereman yang terjadi di bodi kendaraan  
Pada saat mobil melakukan akselerasi dan pengereman, gaya dan gerakan yang terjadi disalurkan melalui suspensi menuju roda kendaraan. Oleh roda kendaraan, gaya tersebut berubah menjadi gesekan dan tekanan yang kuat dengan permukaan jalan sesuai dengan gaya yang diciptakan, (Novriza, 2007).

### 2.3 Sistem Kendali *Open Loop* dan *Closed Loop*

Sistem kendali dapat dikatakan sebagai hubungan antara komponen yang membentuk sebuah konfigurasi sistem yang lebih besar, yang akan menghasilkan tanggapan sistem yang diharapkan. Dalam sistem kendali, yang dikendalikan yaitu merupakan suatu sistem fisis, yang biasa disebut dengan kendalian (*plant*). Masukan dan keluaran merupakan variabel atau besaran fisis. Keluaran merupakan sesuatu yang dihasilkan oleh kendalian, sedangkan masukan adalah sesuatu yang mempengaruhi kendalian, sebagai kondisi awal sistem. Kedua dimensi masukan dan keluaran tidak harus sama. Pada sistem kendali dikenal sistem lup terbuka (*open loop system*) dan sistem lup tertutup (*closed loop system*). Pada sistem kendali lup terbuka, keluarannya tidak diumpanbalikkan ke masukan. Secara umum, karakteristik dari kendali lup terbuka adalah sebagai berikut:

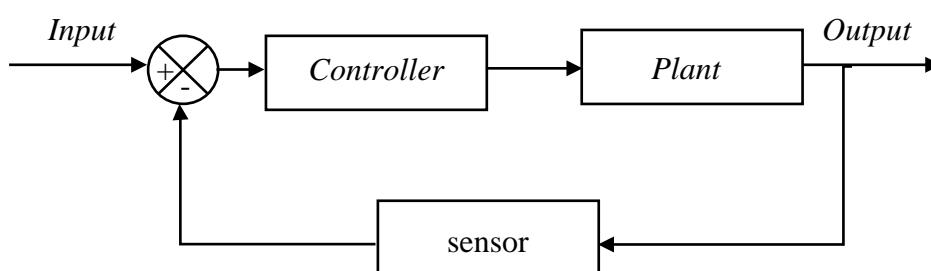
- a) Tindakan pengendaliannya tidak tergantung dari *output* sistem.
- b) Tidak memiliki kemampuan dalam kompensasi/koreksi jika ada gangguan
- c) Ketepatan hasil bergantung pada kalibrasi
- d) Sederhana dan murah



**Gambar 2.1** Diagram blok sistem kendali *open loop*

Sebaliknya, sistem kendali lup tertutup (*closed loop*) adalah sistem kendali yang sinyal keluarannya mempunyai pengaruh langsung pada aksi pengendalian. Sistem kendali lup tertutup merupakan sistem kendali berumpan balik. Sinyal kesalahan sistem, yang merupakan selisih antara sinyal masukan dan sinyal keluaran, diumpangkan ke *controller* untuk memperkecil kesalahan dan membuat agar keluaran sistem mendekati sinyal referensi masukan. Dengan kata lain, istilah “lup tertutup” berarti menggunakan aksi umpan balik untuk memperkecil kesalahan sistem. Secara umum, karakteristik dari kendali *closed loop* adalah:

- Tindakan pengendaliannya tergantung pada *output* sistem (melalui *feedback*).
- Dapat melakukan koreksi saat ada gangguan
- Terdapat kemungkinan terjadi “overkoreksi”, sehingga sistem justru menjadi tidak stabil
- Kompleks dan mahal, karena komponen lebih banyak



**Gambar 2.2** Diagram blok sistem kendali *closed loop*

#### 2.4 Kendali Konvensional dan Kendali Modern

Pada umumnya sistem kendali dibedakan menjadi kendali konvensional dan kendali modern. Kedua jenis pengendali ini berbeda. Sistem kendali konvensional bersifat *single input single output* (SISO), kemudian pemodelan matematikanya dengan menggunakan *transfer function*. Sistem kendali konvensional lebih mudah dalam tuning bobot pengendalinya karena masukan dan keluarannya hanya satu. Akan tetapi jika kendali ini diterapkan ke sistem yang kompleks, akan lebih sulit untuk dilakukan

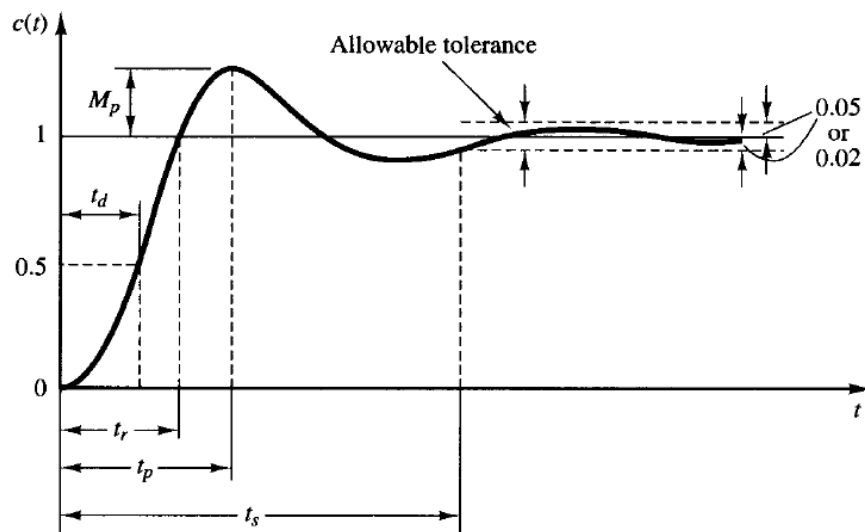
Kendali modern dapat diterapkan pada sistem *multi input multi output* (MIMO), yang kondisinya linier ataupun tak linier, dengan parameter sistem konstan atau berubah terhadap waktu. Penentuan model matematikanya biasanya menggunakan matriks. Tetapi kedua kendali ini dapat disimulasikan pada model persamaan dinamika sistem secara langsung.

## 2.5 Respon Dinamis Sistem

Dalam sistem kendali, terdapat pokok bahasan sebagai berikut:

- Waktu tunda (*delay time*),  $t_d$ : waktu yang dibutuhkan oleh respon untuk mencapai  $\frac{1}{2}$  harga akhir pada saat lonjakan pertama.
- Waktu naik (*rise time*),  $t_r$ : waktu yang dibutuhkan oleh respon untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0% menjadi 100% dari nilai akhir. Untuk sistem orde dua kurang teredam (*underdamped*) digunakan waktu naik 0%-100%, sedangkan untuk sistem redaman lebih (*overdamped*) digunakan waktu naik 10% - 90%.
- Waktu puncak (*peak time*),  $t_p$ : waktu yang diperlukan sistem untuk mencapai lonjakan/lewatan maksimum.
- *Maximum overshoot*,  $M_p$ : harga puncak maksimum dari kurva respon yang diukur dari satu. Jika harga keadaan tunak respon tidak sama dengan satu, maka dapat digunakan persen *maximum overshoot*.
- Waktu keadaan tunak (*settling time*),  $t_s$ : waktu yang dibutuhkan oleh respon untuk mencapai harga tertentu dan tetap dalam nilai akhir, biasanya (2% atau 5%).
- Galat keadaan tunak (*steady state error*),  $e_{tss}$ : nilai sinyal galat yang merupakan selisih dari nilai referensi dengan nilai sebenarnya pada waktu tak terhingga.

Perincian di atas bisa dilihat secara lebih jelas pada Gambar 2.3 (Ogata, 2010 :151) berikut:



**Gambar 2.3** Respon dinamis sistem.

Sumber: (Ogata, 2010 :151)

## 2.6 Pemodelan Sistem Menggunakan *State Space*

Dalam analisis *state space*, terdapat tiga jenis variabel yang terlibat dalam pemodelan sistem dinamis, yaitu : variabel *input*, variabel *output*, dan variabel status. Asumsikan sistem dengan *multi input multi output*, dan  $n$  integrator. Anggap juga terdapat  $r$  masukan  $u_1(t), u_2(t), \dots, u_r(t)$  dan  $m$  keluaran  $y_1(t), y_2(t), \dots, y_m(t)$ . Tetapkan  $n$  keluaran integrator sebagai variabel keadaan  $x_1(t), x_2(t), \dots, x_n(t)$ . Dari pernyataan-pernyataan di atas, sistem dapat dinyatakan dalam persamaan:

$$\begin{aligned}
 \dot{x}_1(t) &= f_1(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\
 \dot{x}_2(t) &= f_2(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 \dot{x}_n(t) &= f_n(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t)
 \end{aligned}
 \tag{2.1}$$

keluaran  $y_1(t), y_2(t), \dots, y_m(t)$  dari sistem dinyatakan oleh persamaan:

$$\begin{aligned}
\dot{y}_1(t) &= g_1(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\
\dot{y}_2(t) &= g_2(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\
&\quad \vdots \\
&\quad \vdots \\
&\quad \vdots \\
\dot{y}_m(t) &= g_m(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t)
\end{aligned} \tag{2.2}$$

Jika didefinisikan:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ \vdots \\ x_n(t) \end{bmatrix}, \quad f(x, u, t) = \begin{bmatrix} f_1(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\ f_2(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\ \vdots \\ \vdots \\ f_n(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \end{bmatrix},$$

$$y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ \vdots \\ y_m(t) \end{bmatrix}, \quad f(x, u, t) = \begin{bmatrix} g_1(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\ g_2(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \\ \vdots \\ \vdots \\ g_m(x_1, x_2, \dots, x_n; u_1, u_2, \dots, u_r; t) \end{bmatrix},$$

$$u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ \vdots \\ u_r(t) \end{bmatrix},$$

maka persamaan (2.1) dan persamaan (2.2) menjadi

$$\dot{x}(t) = f(x, u, t) \tag{2.3}$$

$$y(t) = g(x, u, t) \tag{2.4}$$

dimana persamaan (2.3) adalah persamaan keadaan dan persamaan (2.4) adalah persamaan keluarannya. Bila fungsi  $f$  atau  $g$  mengandung variabel yang eksplisit terhadap waktu  $t$ , maka sistem disebut sebagai sistem yang bervariasi terhadap waktu. (Ogata, 2010)

Bila persamaan (2.3) dan (2.4) dilinierkan terhadap keadaan operasi, maka akan didapatkan persamaan keadaan dan persamaan keluaran yang sudah dilinierkan sebagai berikut:



$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (2.5)$$

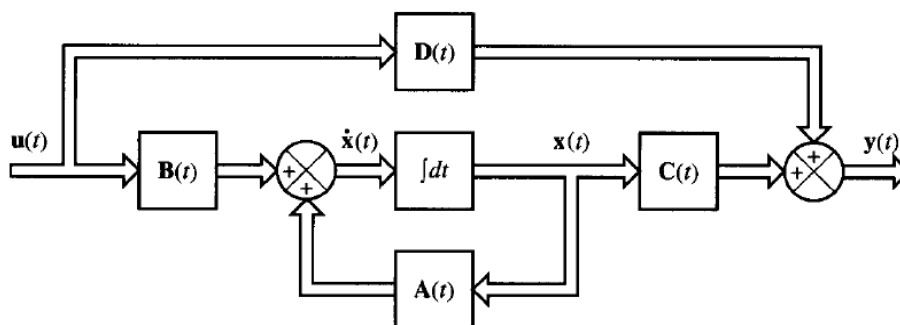
$$y(t) = C(t)x(t) + D(t)u(t) \quad (2.6)$$

dimana  $A(t)$  adalah matriks keadaan,  $B(t)$  adalah matriks masukan,  $C(t)$  adalah matriks keluaran, dan  $D(t)$  disebut matriks transmisi langsung. Bila fungsi  $f$  atau  $g$  pada persamaan (2.3) dan (2.4) tidak mengandung variabel yang eksplisit terhadap waktu  $t$ , maka sistem disebut sebagai sistem yang tidak bervariasi terhadap waktu dan persamaan (2.5) dan persamaan (2.6) dapat dinyatakan dalam bentuk yang lebih sederhana sebagai berikut:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.7)$$

$$y(t) = Cx(t) + Du(t) \quad (2.8)$$

Diagram blok yang merepresentasikan persamaan (2.5) dan persamaan (2.6) ditunjukkan oleh Gambar 2.5 (Ogata, 2010:31) berikut.



**Gambar 2.4** Diagram blok dari sistem linier waktu kontinu yang direpresentasikan dalam bentuk ruang keadaan.

Sumber: (Ogata, 2010:31)

## 2.7 Keterkendalian dan Keteramatan Sistem

Suatu sistem akan dapat dikatakan terkendali dan dapat diamati (teramati) jika memenuhi syarat sebagai berikut.

Matriks keterkendalian:

$$[ B \mid AB \mid \dots \mid A^{(n-1)}B ] \quad (2.9)$$

Agar sistem dapat dikendalikan, maka matriks keterkendalian harus memenuhi kriteria sebagai berikut.

$$\text{rank}[ B \mid AB \mid \dots \mid A^{(n-1)}B ] = n$$

atau

$$\det[ B \mid AB \mid \dots \mid A^{(n-1)}B ] \neq 0$$

Suatu sistem dikatakan dapat teramati jika sistem dari suatu kondisi awal  $x(t_0)$  dapat ditentukan oleh pengamatan  $y(t)$  selama periode waktu terhingga.

Matriks keteramatan:

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} \quad (2.10)$$

Agar sistem teramati, maka matriks keteramatan harus memenuhi kriteria sebagai berikut.

$$\text{rank}[ C^T \mid A^T C^T \mid \dots \mid A^{T(n-1)} C^T ] = n$$

atau

$$\det[ C^T \mid A^T C^T \mid \dots \mid A^{T(n-1)} C^T ] \neq 0$$

## 2.8 Kendali Logika Fuzzy

Logika fuzzy adalah suatu cara memetakan suatu ruang *input* ke dalam suatu ruang *output*. Pada tahun 1965, Profesor Lotfi Asker Zadeh, seorang guru besar *University of California* mempublikasikan karya ilmiah berjudul “*Fuzzy sets*”, dalam karya ilmiah tersebut Profesor Lotfi Asker Zadeh membuat terobosan baru yang memperluas konsep himpunan tegas (*crisp sets*), dalam arti bahwa himpunan tegas merupakan kejadian khusus dari himpunan *fuzzy* (*fuzzy sets*). Logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah biner (0 atau 1), hitam atau putih, ya atau tidak) sedangkan logika fuzzy memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk linguistik, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat”.

### 2.8.1 Bahasan Dalam Sistem Fuzzy

#### a. Variabel fuzzy

Variabel fuzzy merupakan variabel yang akan dibahas dalam suatu sistem fuzzy.

Contoh: umur, suhu, dan massa.

b. Himpunan fuzzy

Himpunan fuzzy merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam variabel fuzzy. Himpunan fuzzy memiliki dua atribut yaitu:

- Linguistik, yaitu penamaan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dengan menggunakan bahasa alami, seperti: sedikit, tetap, dan banyak.
- Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 5, 15, 25, 35.

c. Semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel fuzzy. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton.

d. Domain

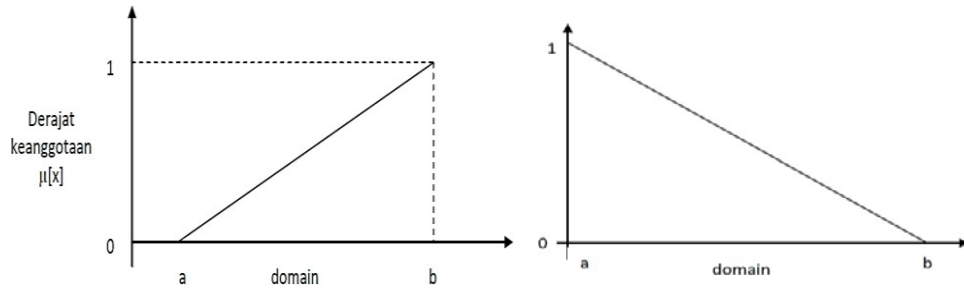
Domain himpunan fuzzy adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan fuzzy. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan.

### 2.8.2 Fungsi Keanggotaan

Fungsi keanggotaan 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. Contoh fungsi keanggotaan adalah :

a. Representasi linier

Pada representasi linear, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Terdapat 2 keadaan himpunan fuzzy yang linier yaitu linier naik dan linier turun. Linier naik menunjukkan semakin ke kanan nilai domain semakin besar derajat keanggotaannya, sedangkan linier turun merupakan kebalikannya seperti ditunjukkan pada Gambar 2.5



**Gambar 2.5** Representasi linier naik dan turun

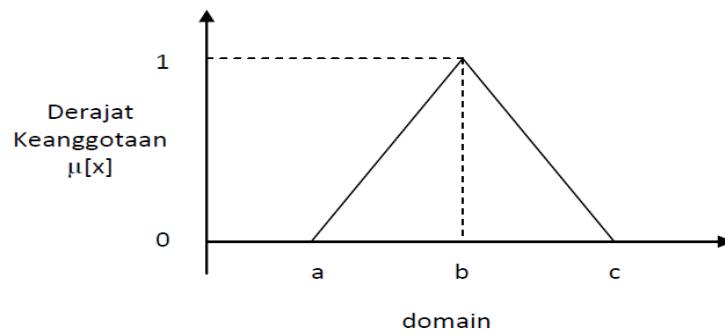
Sumber: (Kusumadewi dan Purnomo, 2004)

Nilai pada Gambar 2.5 adalah sebagai berikut :

$$\mu[x] = \begin{cases} 0 & ; x \leq a \\ \frac{x-a}{b-a} & ; a < x < b \\ 1 & ; x \geq b \end{cases} \qquad \mu[x] = \begin{cases} 0 & ; x \geq b \\ \frac{b-x}{b-a} & ; a < x < b \\ 1 & ; x \leq a \end{cases}$$

b. Representasi kurva segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara dua garis linear. Representasi jenis ini dapat ditunjukkan seperti Gambar 2.6 (Kusumadewi dan Purnomo, 2004).



**Gambar 2.6** Representasi kurva segitiga

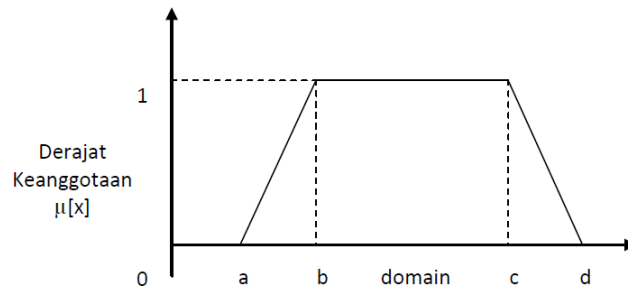
Sumber: (Kusumadewi dan Purnomo, 2004)

Nilai pada Gambar 2.6 adalah :

$$\mu[x] = \begin{cases} 0 & ; x \geq c \text{ atau } x \leq a \\ \frac{x-a}{b-a} & ; a < x < b \\ \frac{c-x}{c-b} & ; b < x < c \end{cases}$$

### c. Representasi Kurva Trapesium

Bentuk kurva trapesium ditunjukkan pada Gambar 2.7 berikut.



**Gambar 2.7** Representasi kurva trapesium

Sumber: (Kusumadewi dan Purnomo, 2004)

Nilai pada Gambar 2.7 adalah :

$$\mu[x] = \begin{cases} 0; & x \geq d \text{ atau } x \leq a \\ \frac{x-a}{b-a}; & a < x < b \\ \frac{d-x}{d-c}; & c < x < d \\ 1 & ; b \leq x \leq c \end{cases}$$

### 2.8.3 Operator Dasar Zadeh

Terdapat beberapa operasi yang digunakan untuk mengkombinasi dan memodifikasi himpunan fuzzy. Hasil dari operasi 2 himpunan sering dikenal dengan nama fire strength. Ada 3 operator dasar yang diciptakan oleh Zadeh, yaitu:

#### a. Operator AND

*Fire strength* hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil dari himpunan yang bersangkutan

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

#### b. Operator OR

*Fire strength* dari hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar dari himpunan yang bersangkutan.

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

c. Operator NOT

*Fire strength* dari hasil operasi dengan operator NOT diperoleh dengan mengurangi nilai keanggotaan elemen pada himpunan yang bersangkutan.

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

#### 2.8.4 Fungsi Implikasi

Bentuk umum dari aturan yang digunakan dalam fungsi implikasi adalah,

$$\text{if } x \text{ is } A \text{ then } y \text{ is } B$$

dengan  $x$  dan  $y$  adalah skalar, sedangkan  $A$  dan  $B$  adalah himpunan fuzzy. Proposisi yang mengikuti *if* disebut anteseden dan proposisi yang mengikuti *then* disebut konsekuen.

#### 2.8.5 Sistem Inferensi Fuzzy

Salah satu sistem inferensi fuzzy yaitu metode Mamdani, atau sering juga dikenal dengan nama metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Pada metode ini, untuk mendapatkan *output* diperlukan 4 langkah yaitu :

1. Fuzzifikasi

Variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan.

2. Aplikasi Fungsi Implikasi

Fungsi implikasi yang digunakan dalam metode Mamdani adalah min.

3. Komposisi Aturan (Agregasi)

Terdapat tiga metode yang digunakan dalam melakukan inferensi sistem fuzzy, yaitu: max, sum, dan probor.

4. Defuzzifikasi

*Input* dari defuzzifikasi adalah suatu himpunan yang diperoleh dari komposisi aturan-aturan fuzzy, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan fuzzy tersebut. Beberapa metode defuzzifikasi yaitu :

a. Metode *Centroid* (COA)

b. Metode *Mean of Maximum* (MOM)

c. Metode *Largest of Maximum* (LOM)

- d. Metode *Smallest of Maximum* (SOM)
- e. *Bisector*

### 2.8.6 Langkah Mendesain Dengan Logika Fuzzy

- a. Menentukan semua variabel yang terkait dalam proses yang akan dikendalikan. Proses ini meliputi penentuan variabel masukan dan variabel keluaran kemudian penentuan semesta numeris dan nilai linguistiknya yang direpresentasikan menggunakan fungsi keanggotaan tertentu. Hal yang perlu diperhatikan ketika menentukan variabel sistem kendali adalah,
  - *Error* ( $e$ ), yaitu selisih antara nilai sesungguhnya dengan nilai yang diinginkan.
  - *Delta Error* ( $\Delta e$ ) atau turunan dari *error*, yang menyatakan perubahan *error* itu sendiri.

Sistem kendali akan menghasilkan nilai keluaran yang menyatakan besar pengendalian yang harus dilakukan berdasarkan nilai  $e$  dan  $\Delta e$ .

- b. Menentukan fuzzifikasi yang sesuai dengan variabel-variabel yang terlibat.
- c. Membangun basis aturan fuzzy, berupa implikasi dari relasi variabel masukan dan keluaran sistem
- d. Menentukan defuzzifikasi yang sesuai untuk mengubah himpunan fuzzy menjadi bilangan tegas (*crisp*)

## 2.9 Konsep Sistem Kendali Optimal *Linear Quadratic Regulator* (LQR)

Sistem kendali optimal adalah sistem kendali dimana mempunyai unjuk kerja performansi terbaik. Kendali optimal memerlukan suatu kriteria optimasi yang dapat meminimumkan hasil pengukuran dengan perubahan perilaku sistem terhadap referensinya. Pengukuran tersebut dengan cara menentukan indeks performansi yang merupakan suatu tolak ukur terhadap besar kinerja sistem apakah sistem tersebut sudah sesuai dengan yang diinginkan.

Asumsikan sebuah sistem dengan persamaan (2.7) dimana  $x$  adalah vektor keadaan ( $n$  vektor),  $u$  adalah vektor kendali ( $r$  vektor),  $A$  adalah matriks konstan berorde  $n \times n$ , dan  $B$  adalah matriks konstan berorde  $n \times r$ . Dalam mendesain suatu sistem kendali, dalam memilih vektor kendali  $u(t)$  sehingga indeks

performansinya diminimalkan. Dapat dibuktikan bahwa sebuah indeks performansi kuadratis, dimana batas-batas integrasi nya adalah 0 dan  $\infty$  seperti pada persamaan:

$$J = \int_0^{\infty} \mathcal{L}(x, u) dt \quad (2.11)$$

dimana  $\mathcal{L}(x, u)$  adalah fungsi kuadratis atau Fungsi Hermitian dari  $x$  dan  $u$  akan menghasilkan hukum kendali linier yaitu:

$$u(t) = -Kx(t) \quad (2.12)$$

dimana  $K$  adalah matriks berorde  $r \times n$ , atau

$$\begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_r \end{bmatrix} = - \begin{bmatrix} k_{11} & k_{12} & \cdot & \cdot & \cdot & k_{1n} \\ k_{21} & k_{22} & \cdot & \cdot & \cdot & k_{2n} \\ \cdot & \cdot & & & & \cdot \\ \cdot & \cdot & & & & \cdot \\ \cdot & \cdot & & & & \cdot \\ k_{r1} & k_{r2} & \cdot & \cdot & \cdot & k_{rn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \quad (2.13)$$

Oleh karena itu, desain dari sistem kendali optimal dan sistem regulator optimal berdasarkan indeks performansi kuadratis seperti di atas tergantung dari penentuan nilai elemen-elemen dari matriks  $K$ .

Kelebihan dari penggunaan skema kendali kuadratis optimal adalah sistem yang didesain akan selalu stabil, kecuali pada kasus dimana sistem tidak terkendali. Dalam mendesain suatu sistem kendali berdasarkan minimisasi dari indeks performansi kuadratis, diperlukan penyelesaian pada persamaan Aljabar Riccati tereduksi. Program matlab mempunyai perintah `LQR` yang dapat memberikan solusi dari persamaan Aljabar Riccati waktu kontinyu dan menentukan nilai penguatan matriks umpan balik optimal.

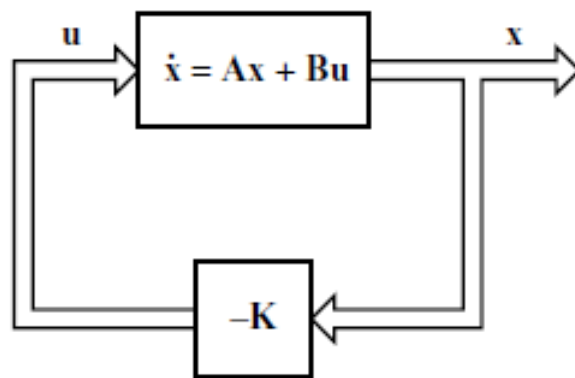
Berikut ini, diasumsikan persoalan untuk menentukan nilai vektor kendali optimal  $u(t)$  untuk sistem yang dinyatakan dalam persamaan (2.7) dan indeks performansinya diberikan oleh persamaan:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (2.14)$$

dimana  $Q$  adalah matriks definit positif (atau semidefinit positif),  $R$  adalah matriks definit positif, dan  $u$  tidak dibatasi. Tujuan utama dari sistem kendali optimal adalah untuk meminimalkan indeks performansinya sehingga sistem menjadi stabil. Dari banyak pendekatan yang berbeda untuk menemukan solusi dari permasalahan seperti ini, salah satu pendekatan yang ada adalah dengan menggunakan metode kedua Liapunov. (Ogata, 1997: 916)



Untuk menentukan matriks Q dan matriks R yaitu dengan cara *trial and error* dengan syarat matriks Q adalah matriks real simetris positif atau real semidefinit positif ( $Q > 0$ ) dan matriks R adalah matriks real simetris definit positif ( $R > 0$ ). Selanjutnya hukum kendali linier berdasarkan persamaan (2.12) adalah hukum kendali optimal. Oleh karena itu, jika elemen yang tidak diketahui dari matriks K ditentukan, untuk meminimalkan indeks kinerja, maka  $u(t) = -Kx(t)$  adalah optimal untuk setiap kondisi awal  $x(0)$ . Diagram blok yang menunjukkan konfigurasi optimal ditunjukkan pada gambar berikut.



**Gambar 2.8** Diagram kendali LQR

Sumber: (Ogata, 1997:920)

Secara singkat tahapan dalam mendesain menggunakan kendali *Linear Quadratic Regulator* (LQR) adalah :

1. Selesaikan Persamaan (2.17) yang merupakan persamaan matriks Riccati tereduksi untuk mendapatkan matriks P.

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (2.17)$$

2. Substitusikan matriks P ini ke dalam Persamaan (2.18). Matriks yang dihasilkan K adalah matriks optimal dan sinyal kendali umpan balik optimal adalah  $u(t)$ .

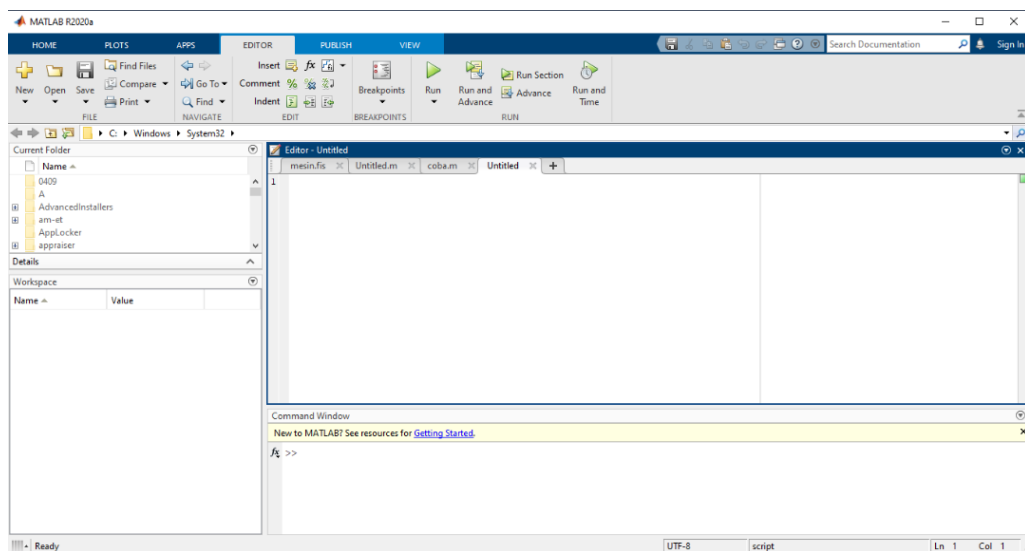
$$u(t) = -Kx(t) = -R^{-1}B^T P x(t) \quad (2.18)$$

## 2.10 Bahan Penelitian

Bahan Penelitian yang digunakan dalam penelitian ini yaitu sebuah sistem suspensi mobil model seperempat lengkap dengan model matematikanya. Model suspensi ini akan dikendalikan dengan dua jenis sistem kendali yang berbeda, yaitu kendali logika fuzzy (*fuzzy logic*) dan *Linear Quadratic Regulator* (LQR).

## 2.11 Alat/Software yang Digunakan

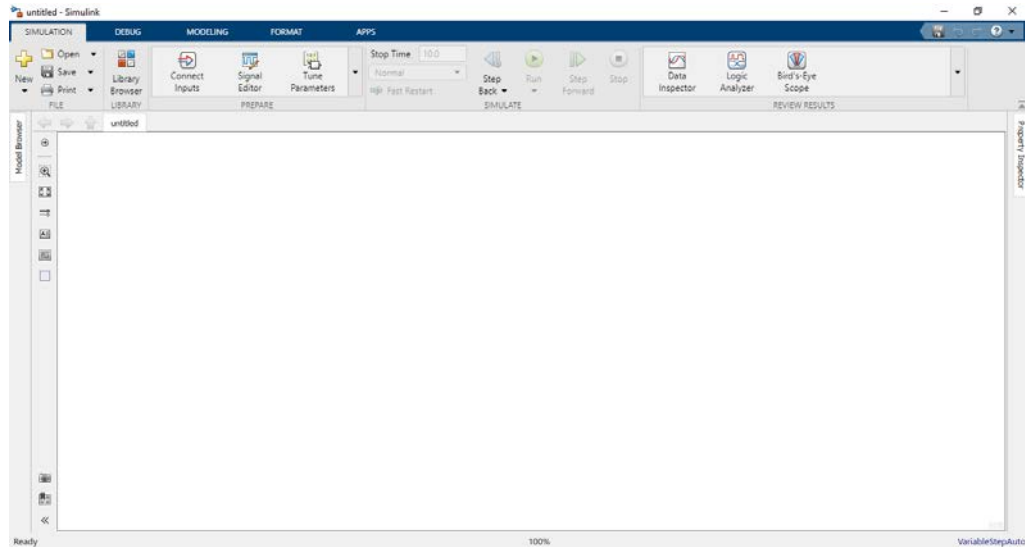
Alat yang digunakan dalam penelitian ini yaitu sebuah *software* matlab R2020a versi 9.8.0.1323502 dan simulink. Matlab merupakan sebuah program komputasi numerik yang menggunakan bahasa berbasis matriks yang digunakan untuk menganalisis data, membuat algoritma, serta membuat pemodelan dan aplikasi. Matlab adalah sebuah alat dengan (*high performance*) kinerja tinggi untuk komputasi masalah teknik. Beberapa penggunaan matlab dalam bidang matematika dan komputasi, pembentukan algoritma, akuisisi data, pemodelan simulasi dan pembuatan prototipe, analisa data eksplorasi dan visualisasi, dan rekayasa.



**Gambar 2.9** Tampilan lembar kerja matlab

Salah satu fitur yang ada dalam matlab yaitu simulink, dimana simulink ini merupakan *tools* yang digunakan untuk menggambarkan pemodelan dan simulasi dari suatu sistem yang dirancang. Dengan menggunakan simulink, maka dapat dengan mudah untuk melihat aliran proses sinyal dari sebuah sistem, karena digambarkan secara nyata dibandingkan dengan membuat program langsung pada

lembar kerja mfile. Beberapa fitur dalam simulink juga mempermudah dalam pemrosesan sistem. Model yang dirancang dalam simulink menggunakan *library* yang sudah tersedia, seperti *sinks*, *sources*, dan *string*.



**Gambar 2.10** Tampilan lembar kerja simulink