

BAB 3

PERANCANGAN SISTEM

3.1 Perancangan Pengendali PDC pada Sistem Truk-Trailer

Model linear fuzzy T-S untuk sistem truk dengan tiga trailer telah dimodelkan sebelumnya, yakni sesuai persamaan (2.44), yaitu

$$\dot{x} = \sum_{i=1}^2 h_i(x_1) \{A_i x + B_i u\} \quad (3.1)$$

dimana terdapat dua aturan (*rule*) yang berlaku. Maka pada pengendali fuzzy PDC juga terdapat dua aturan sehingga persamaan (2.8) dan (2.9) dapat dituliskan menjadi

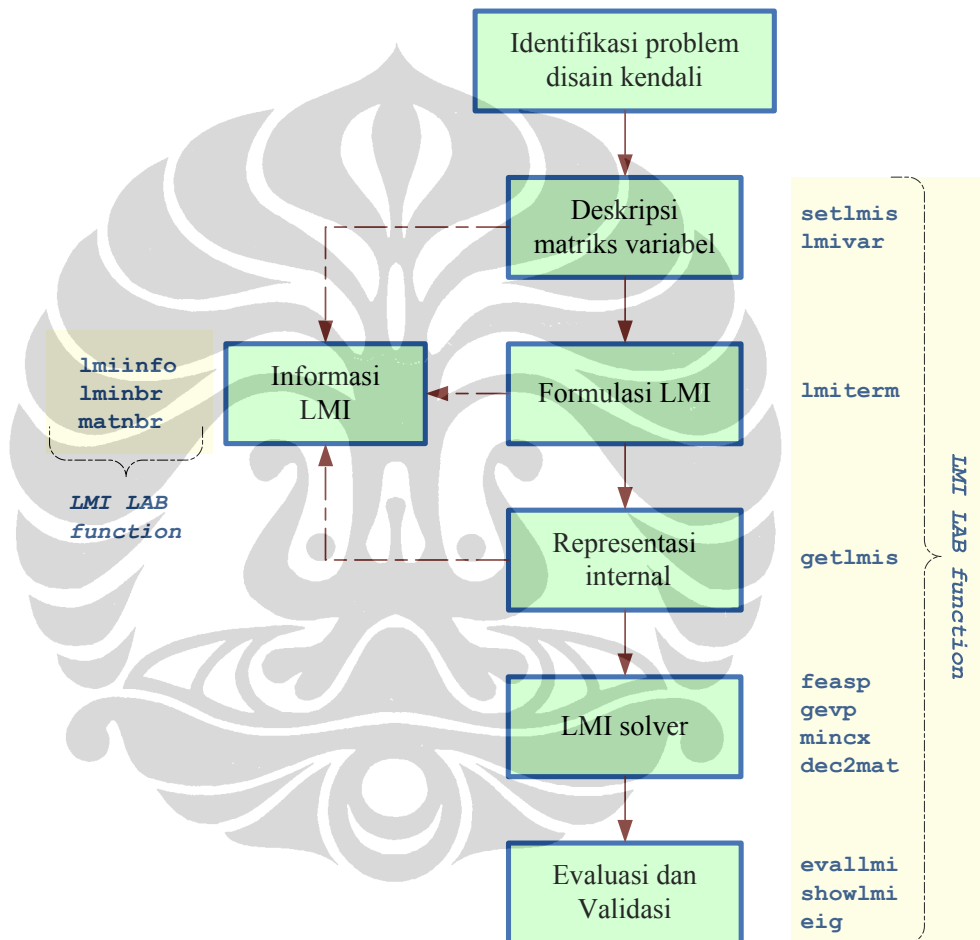
$$\dot{x} = \sum_{i=1}^2 \sum_{j=1}^2 h_i(x_1) h_j(x_1) (A_i - B_i F_j) x \quad (3.2)$$

$$\dot{x} = \sum_i^2 h_i(x_1) h_i(x_1) G_{ii} x + 2 \sum_i^2 \sum_{i < j} h_i(x_1) h_j(x_1) \left\{ \frac{G_{ij} + G_{ji}}{2} \right\} x \quad (3.3)$$

Disain pengendali PDC dapat dilakukan dengan beberapa metode. Jika menggunakan Teorema 2 atau kondisi (2.11) dan (2.12), maka terlebih dahulu ditentukan letak pole yang diinginkan atau *closed-loop eigenvalues* yang diinginkan untuk masing-masing model aturan pertama dan kedua. Matriks *feedback closed-loop* F_1 dan F_2 dicari dengan menggunakan metode penempatan kutub. Kemudian matriks positif definit P ditentukan dengan algoritma optimisasi LMI. Metode yang lain yaitu dengan desain pengendali dimana matriks *feedback gain* F_1 dan F_2 bersama-sama dengan matriks P ditentukan dengan algoritma optimisasi LMI. Untuk memudahkan perancangan, pada tesis ini digunakan LMI *Lab* dan untuk menyelesaikan solusi numerik dari kondisi LMI dan *graphical user interface* (GUI) 'lmiedit' untuk menetapkan kondisi maupun persyaratan sistem dalam formulasi LMI.

Langkah-langkah implementasi LMI *Lab* pada disain kendali fuzzy diperlihatkan pada gambar 3.1 berikut ini. Fungsi-fungsi LMI *Lab* digunakan pada proses-proses deskripsi matriks variabel, formulasi kondisi LMI, representasi

internal, penentuan solusi LMI, evaluasi dan validasi maupun untuk mendapatkan info mengenai sistem LMI yang dibuat. Fungsi-fungsi LMI *solver* digunakan tergantung pada metoda optimisasi, seperti 'feasp' digunakan untuk masalah *feasibility*, 'gevp' digunakan untuk masalah *generalized eigenvalue minimization problem* (GEVP), 'mincx' digunakan untuk masalah minimasi obyektif linear (*linear objective minimization*).



Gambar 3.1 Implementasi LMI Lab pada disain kendali fuzzy

3.2 Disain 1: Pengendali Stabil

Pada perancangan pengendali stabil, kondisi-kondisi yang disyaratkan dalam Teorema 2 perlu dinyatakan dalam P dan F_1 dan F_2 .

Sesuai defenisi sebelumnya,

$$G_{11} = A_1 - B_1 F_1 \quad \Rightarrow \quad G_{11}^T = A_1^T - F_1^T B_1^T$$

$$G_{22} = A_2 - B_2 F_2 \quad \Rightarrow \quad G_{22}^T = A_2^T - F_2^T B_2^T$$

$$G_{12} = A_1 - B_1 F_2 \quad \Rightarrow \quad G_{12}^T = A_1^T - F_2^T B_1^T$$

$$G_{21} = A_2 - B_2 F_1 \quad \Rightarrow \quad G_{21}^T = A_2^T - F_1^T B_2^T$$

dan dengan mendefenisikan matriks variabel baru $X = P^{-1}$, dan mengalikan ruas kiri dan kanan pada pertidaksamaan (2.11) dan (2.12) dengan X , maka dapat ditulis ulang sebagai berikut,

$$-XA_1^T - A_1X + XF_1^T B_1^T + B_1 F_1 X > 0,$$

$$-XA_2^T - A_2X + XF_2^T B_2^T + B_2 F_2 X > 0,$$

$$-XA_1^T - A_1X - XA_2^T - A_2X + XF_2^T B_1^T + B_1 F_2 X + XF_1^T B_2^T + B_2 F_1 X \geq 0,$$

Dengan mendefenisikan $M_1 = F_1 X$ dan $M_2 = F_2 X$ sehingga untuk $X > 0$, dapat diperoleh $F_1 = M_1 X^{-1}$ dan $F_2 = M_2 X^{-1}$. Substitusi ke dalam pertidaksamaan diatas menghasilkan maka syarat kondisi Teorema 2 dapat dinyatakan dalam formulasi LMI seperti berikut,

Temukan matriks X , M_1 dan M_2 yang memenuhi

$$X > 0 \tag{3.4}$$

$$-XA_1^T - A_1X + M_1^T B_1^T + B_1 M_1 > 0, \tag{3.5}$$

$$-XA_2^T - A_2X + M_2^T B_2^T + B_2 M_2 > 0, \tag{3.6}$$

$$-XA_1^T - A_1X - XA_2^T - A_2X + M_2^T B_1^T + B_1 M_2 + M_1^T B_2^T + B_2 M_1 \geq 0 \tag{3.7}$$

dimana

$$X = P^{-1}, \quad M_1 = F_1 X, \quad M_2 = F_2 X$$

Feedback gains F_1 dan F_2 dan matriks bersama P dapat diperoleh dari

$$P = X^{-1}, \quad F_1 = M_1 X^{-1}, \quad F_2 = M_2 X^{-1} \tag{3.8}$$

Sistem LMI di atas dapat dinyatakan dengan LMI editor dengan perincian sebagai berikut [3]:

- (1) Tentukan dimensi dan struktur setiap matriks variabel. Dalam hal ini matriks variabel $X \in R^{6 \times 6}$ simetris, matriks variabel $M_1, M_2 \in R^{2 \times 6}$.
- (2) Nyatakan setiap sistem LMI dalam bentuk pernyataan matriks simbolis (MATLAB *expressions*).

Gambar 3.2 memperlihatkan *graphical user interface* (GUI) LMI Editor untuk menyatakan sistem LMI (3.4) – (3.7) dalam ekspresi simbol matriks atau ekspresi MATLAB. Dengan mengetikkan 'lmiedit' pada *Command Window* MATLAB, akan tampil GUI LMI Editor dengan beberapa area pengeditan untuk mendeklarasikan nama matriks variabel (*variable name*), struktur (*type, structure*) untuk menyatakan tipe dan dimensi matriks variabel.

Setelah sistem LMI dinyatakan dideskripsikan secara lengkap pada LMI Editor, beberapa hal dapat dilakukan dengan menekan/meng-klik tombol yang bersesuaian, yaitu [3]:

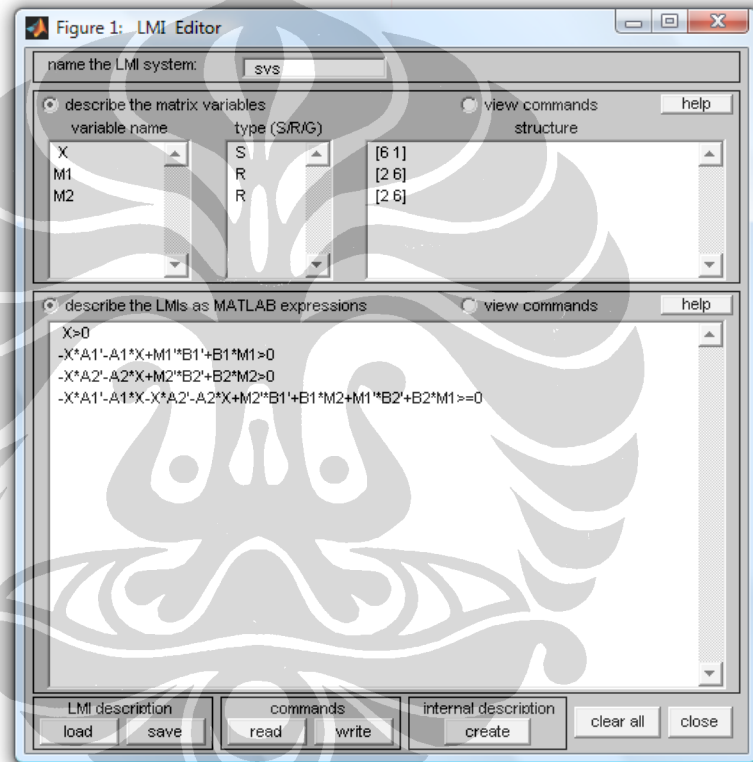
- Menampilkan deskripsi sistem LMI dalam fungsi LMI Lab, ('*view command buttons*') atau sebaliknya, sistem yang dinyatakan secara khusus dalam fungsi LMI Lab dapat ditampilkan dalam ekspresi simbol matriks dengan meng-klik tombol '*describe ...*'.
- Nyatakan setiap sistem LMI dalam bentuk pernyataan matriks simbolis (MATLAB *expressions*).
- Menyimpan ekspresi simbolik dari sistem LMI dalam format *string* ('*save button*').
- Membaca file LMI yang telah ada sebelumnya ('*read button*').
- Menghasilkan representasi internal dengan menekan tombol '*create*'. Hasilnya ditulis dalam MATLAB variabel sesuai dengan nama sistem LMI. Semua data LMI yang terkait akan disimpan dalam *workspace* MATLAB.

Representasi internal ini dapat diarahkan langsung pada LMI *solver* ataupun fungsi LMI Lab lainnya.

Dengan LMI *solver* untuk mencari solusi *feasible* sistem LMI (3.4) – (3.7), digunakan fungsi '*feasp*' dengan format sebagai berikut:

```
[tmin, xfeas]=feasp(sys)
Xf=dec2mat(sys, xfeas, X)
M1f=dec2mat(sys, xfeas, M1)
M2f=dec2mat(sys, xfeas, M2)
```

Matriks Xf , $M1f$, dan $M2f$ masing-masing bersesuaian dengan matriks variabel X , M_1 , dan M_2 , sehingga dapat diperoleh matriks definit positif P dan *feedback gains* F_1 dan F_2 menurut persamaan (3.8)



Gambar 3.2 GUI LMI editor

Penyelesaian sistem LMI (3.4) – (3.7) menghasilkan:

$$F_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.499 & 0 \\ 1.666 & -1.796 & 1.651 & -0.089 & 0 & 1.002 \end{bmatrix},$$

$$F_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.499 & 0 \\ 1.734 & -1.800 & 1.660 & -0.074 & 0 & 1.125 \end{bmatrix}, \text{ dan}$$

$$P = \begin{bmatrix} 1.351 & -0.195 & 0.148 & 0.183 & 0 & 0.669 \\ -0.195 & 1.252 & -0.611 & 0.112 & 0 & -0.034 \\ 0.148 & -0.611 & 2.015 & -0.867 & 0 & 0.087 \\ 0.183 & 0.112 & -0.867 & 1.549 & 0 & 0.144 \\ 0 & 0 & 0 & 0 & 0.875 & 0 \\ 0.669 & -0.034 & 0.087 & 0.144 & 0 & 1.214 \end{bmatrix} > 0$$

Untuk kondisi *relaxed stability* sesuai Teorema 3, formulasi LMI dapat dinyatakan seperti berikut,

Temukan matriks X , Y , M_1 dan M_2 yang memenuhi

$$X > 0 \quad (3.9)$$

$$Y \geq 0 \quad (3.10)$$

$$-XA_1^T - A_1X + M_1^T B_1^T + B_1 M_1 - (s-1)Y > 0, \quad (3.11)$$

$$-XA_2^T - A_2X + M_2^T B_2^T + B_2 M_2 - (s-1)Y > 0, \quad (3.12)$$

$$\begin{aligned} & 2Y - XA_1^T - A_1X - XA_2^T - A_2X \\ & + M_2^T B_1^T + B_1 M_2 + M_1^T B_2^T + B_2 M_1 \geq 0 \end{aligned} \quad (3.13)$$

dimana

$$X = P^{-1}, \quad M_1 = F_1 X, \quad M_2 = F_2 X, \quad Y = XQX$$

Feedback gains F_1 dan F_2 dan matriks bersama P dan Q dapat diperoleh dari

$$P = X^{-1}, \quad F_1 = M_1 X^{-1}, \quad F_2 = M_2 X^{-1}, \quad Q = PYP \quad (3.14)$$

3.3 Disain 2: Pengendali Stabil dengan *Constraint* pada Input

Dalam mendesain pengendali, terdapat batasan-batasan (*constraints*) yang perlu diperhatikan dalam hal input pengendali ataupun keluaran model. Sebagai contoh, dalam perancangan pengendali fuzzy untuk truk dengan 3 trailer, diterapkan *constraint* untuk input sudut kemudi, besarnya tidak melebihi 30° . Sedangkan *constraint* untuk keluaran selisih sudut antara truk dan trailer ataupun antar trailer besarnya tidak melebihi 90° .

Teorema 4. Asumsikan bahwa kondisi awal $x(0)$ diketahui. Batasan input kendali $\|u(t)\|_2 \leq \mu$ dipenuhi sepanjang waktu $t \geq 0$ jika syarat LMI

$$\begin{bmatrix} 1 & x(0)^T \\ x(0) & X \end{bmatrix} \geq 0, \quad (3.15)$$

$$\begin{bmatrix} X & M_i^T \\ M_i & \mu^2 I \end{bmatrix} \geq 0, \quad (3.16)$$

dapat terpenuhi, dimana $X = P^{-1}$ dan $M_i = F_i X$.

Untuk desain pengendali dengan menambahkan batasan atau *constraint* pada input pengendali, yakni $\|u(t)\| \leq \mu$, dalam hal ini $\mu = 30^\circ$ maka kondisi LMI (3.9) – (3.13) dimodifikasi dengan penambahan kondisi (3.15) dan (3.16), yaitu:

Temukan matriks X , Y , M_1 dan M_2 yang memenuhi

$$X > 0 \quad (3.17)$$

$$Y \geq 0 \quad (3.18)$$

$$-XA_1^T - A_1X + M_1^T B_1^T + B_1 M_1 - (s-1)Y > 0, \quad (3.19)$$

$$-XA_2^T - A_2X + M_2^T B_2^T + B_2 M_2 - (s-1)Y > 0, \quad (3.20)$$

$$\begin{aligned} & 2Y - XA_1^T - A_1X - XA_2^T - A_2X \\ & + M_2^T B_1^T + B_1 M_2 + M_1^T B_2^T + B_2 M_1 \geq 0 \end{aligned} \quad (3.21)$$

$$\begin{bmatrix} 1 & x(0)^T \\ x(0) & X \end{bmatrix} \geq 0, \quad (3.22)$$

$$\begin{bmatrix} X & M_1^T \\ M_1 & \mu^2 I \end{bmatrix} \geq 0, \quad (3.23)$$

$$\begin{bmatrix} X & M_2^T \\ M_2 & \mu^2 I \end{bmatrix} \geq 0 \quad (3.24)$$

Penerapan algoritma optimisasi LMI dengan cara yang sama pada sub bab 3.3, diperoleh :

$$F_1 = \begin{bmatrix} -0.0004 & -0.000 & -0.000 & -0.000 & 0.0137 & -0.0001 \\ 0.0741 & -0.000 & 0.000 & 0.000 & -0.0001 & 0.0120 \end{bmatrix},$$

$$F_2 = \begin{bmatrix} -0.0001 & -0.000 & -0.000 & -0.000 & 0.0183 & -0.000 \\ 0.0898 & 0.000 & -0.000 & 0.0001 & -0.0003 & 0.0138 \end{bmatrix},$$

$$P = \begin{bmatrix} 0.0407 & 0.0002 & 0.0001 & 0.0001 & -0.0001 & 0.0045 \\ 0.0002 & 0.0062 & -0.0014 & 0.0001 & -0.0000 & 0.0000 \\ 0.0001 & -0.0014 & 0.0056 & -0.0018 & -0.0000 & 0.0000 \\ 0.0001 & 0.0001 & -0.0018 & 0.0046 & -0.0000 & 0.0000 \\ -0.0001 & -0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 \\ 0.0045 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0020 \end{bmatrix} > 0$$

3.4 Disain 3: Pengendali Stabil dengan *Constraint* pada Input dan Output

Teorema 5. Asumsikan bahwa kondisi awal $x(0)$ diketahui. Batasan output $\|y(t)\|_2 \leq \lambda$ dipenuhi sepanjang waktu $t \geq 0$ jika syarat LMI

$$\begin{bmatrix} 1 & x(0)^T \\ x(0) & X \end{bmatrix} \geq 0, \quad (3.25)$$

$$\begin{bmatrix} X & XC_i^T \\ C_i X & \lambda^2 I \end{bmatrix} \geq 0, \quad (3.26)$$

dapat terpenuhi, dimana $X = P^{-1}$ dan $M_i = F_i X$.

Untuk desain pengendali dengan menambahkan batasan atau *constraint* pada output pengendali, yakni $\|u(t)\|_2 \leq \lambda$, dalam hal ini $\lambda = 90^\circ$ dan dipilih x_2 , x_3 , dan x_4 sebagai output maka matriks keluaran C_i adalah

$$C_1 = C_2 = C = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Kondisi LMI (3.17) – (3.24) ditambahkan kondisi (3.26) yaitu:

$$\begin{bmatrix} X & XC^T \\ CX & \lambda^2 I \end{bmatrix} \geq 0, \quad (3.27)$$

Dengan penerapan algoritma optimisasi LMI cara yang sama, diperoleh:

$$F_1 = 10^{-5} \times \begin{bmatrix} -0.0098 & -0.0882 & -0.0990 & -0.1110 & 0.4940 & -0.0029 \\ 0.0023 & 0.0178 & 0.0109 & 0.0122 & -0.0001 & 0.0012 \end{bmatrix},$$

$$F_2 = \begin{bmatrix} 0.0007 & 0.0015 & 0.0014 & 0.0012 & 0.0022 & 0.0000 \\ 0.0316 & 0.0264 & 0.0232 & 0.0193 & -0.0000 & 0.0023 \end{bmatrix},$$

$$P = \begin{bmatrix} 0.0056 & 0.0071 & 0.0070 & 0.0069 & -0.0000 & 0.0002 \\ 0.0071 & 0.5109 & 0.0122 & 0.0088 & -0.0000 & 0.0006 \\ 0.0070 & 0.0122 & 0.5044 & 0.0063 & -0.0000 & 0.0006 \\ 0.0069 & 0.008 & 0.0063 & 0.4992 & -0.0000 & 0.0006 \\ -0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0001 & -0.0000 \\ 0.0002 & 0.0006 & 0.0006 & 0.0006 & -0.0000 & 0.0001 \end{bmatrix} > 0$$

$$Q = 10^3 \times \begin{bmatrix} 0.0018 & 0.0378 & 0.0368 & 0.0356 & -0.0000 & 0.0001 \\ 0.0378 & 2.6092 & 0.1244 & 0.0896 & -0.0000 & 0.0031 \\ 0.0368 & 0.1244 & 2.5429 & 0.0651 & -0.0000 & 0.0031 \\ 0.0356 & 0.0896 & 0.0651 & 2.4889 & -0.0000 & 0.0030 \\ -0.0000 & -0.0000 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\ 0.0001 & 0.0031 & 0.0031 & 0.0030 & -0.0000 & 0.0000 \end{bmatrix} \geq 0$$

3.5 Disain 4: Pengenal Stabil dengan *Initial State Independent*

Salah satu kelemahan dari desain pengendali dengan batasan input dan output adalah ketergantungan pada nilai awal sistem [2]. Ini berarti bahwa *feedback gains* F_i harus ditentukan kembali jika nilai awal variabel keadaan mengalami perubahan. Agar tidak perlu selalu menghitung F_i setiap kali terjadi perubahan nilai awal variabel keadaan, maka LMI dengan *constarint* pada input ataupun output dapat dimodifikasi, dimana nilai awal $x(0)$ tidak perlu diketahui tetapi batas atas Φ dari $\|x(0)\|$ diketahui, yaitu $\|x(0)\| \leq \Phi$. Batas Φ dapat diset cukup besar agar dapat mencakup lebih banyak nilai awal variabel keadaan bahkan jika nilai $x(0)$ tidak diketahui.

Modifikasi dari LMI *constraint* dapat diselesaikan sebagai berikut:

Teorema 6. Asumsikan bahwa kondisi awal $\|x(0)\| \leq \Phi$ dimana $x(0)$ tidak diketahui tetapi batas atas Φ diketahui. Maka

$$x^T(0) X^{-1} x(0) \leq 1, \quad (3.28)$$

jika

$$\Phi^2 I \leq X, \quad (3.29)$$

Dimana $X = P^{-1}$

Dapat dilihat bahwa pertidaksamaan (3.28) ekuivalen dengan (3.15) dan (3.25). kondisi (3.29) dapat digunakan dalam modifikasi LMI sebagai ganti dari (3.15) ataupun (3.25).

Disain pengendali dengan kondisi nilai awal variabel keadaan yang independen dibuat dengan menambahkan Teorema 6 dalam perancangan kondisi LMI, dalam hal ini batas atas dari nilai awal ditentukan, yaitu $\Phi = 10$. dan dengan algoritma LMI yang sama, diperoleh *feedback gain* F_1 dan F_2 serta matriks positif definit P sebagai berikut :

$$F_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.0323 & 0 \\ 0.1452 & 0.0006 & 0.0005 & 0.0004 & 0 & 0.0367 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.0444 & 0 \\ 0.1652 & 0.0002 & 0.0004 & 0.0004 & 0 & 0.0419 \end{bmatrix}$$

$$P = \begin{bmatrix} 0.1102 & 0.0007 & 0.0007 & 0.0007 & 0 & 0.0221 \\ 0.0007 & 0.0120 & 0.0002 & -0.0000 & 0 & 0.0002 \\ 0.0007 & 0.0002 & 0.0115 & -0.0001 & 0 & 0.0002 \\ 0.0007 & -0.0000 & -0.0001 & 0.0112 & 0 & 0.0002 \\ 0 & 0 & 0 & 0 & 0.0120 & 0 \\ 0.0221 & 0.0002 & 0.0002 & 0.0002 & 0 & 0.0160 \end{bmatrix} > 0$$

3.6 Validasi Hasil Optimisasi LMI

Kesemua matriks P yang diperoleh diatas adalah matriks simetris definit positif. Dimensi dan deskripsi berbagai matriks yang digunakan dalam disain kendali ini dapat dilihat pada tabel 3.1. Matriks P adalah matriks definit positif jika dan hanya jika semua *eigenvalue* matriks P adalah positif, hal ini benar jika dan hanya jika semua *leading principal minors* (determinan dari *leading principal submatrices*) dari matriks P adalah positif.

Matriks Q adalah matriks semidefinit positif jika dan hanya jika semua *eigenvalue* matriks P adalah non-negatif, hal ini benar jika dan hanya jika semua *leading principal minors* (determinan dari *leading principal submatrices*) dari matriks Q adalah non-negatif.

Tabel 3.1 Dimensi matriks

Matriks	Dimensi	Deskripsi
A_1, A_2	6×6	Matriks <i>state</i> aturan 1 dan 2
B_1, B_2	6×2	Matriks input aturan 1 dan 2
F_1, F_2	2×6	Matriks <i>feedback gains</i>
P	6×6	Matriks simetris definit positif
X	6×6	$X = P^{-1}$, simetris definit positif
Q	6×6	Matriks simetris semidefinit positif
M_1, M_2	2×6	$M_i = F_i X$

Jika P adalah matriks $n \times n$, maka *leading principal submatrix* orde- m (P_m) dari P adalah matriks yang dibentuk dengan menghapus $n - m$ baris dan kolom terakhir dari P . Matriks $P \in R^{6 \times 6}$ memiliki enam *leading principal submatrix*, dan $P_6 = P$.

Matriks S adalah negatif jika $-S$ adalah positif. Mengalikan sebuah matriks dengan -1 sama dengan mengalikan determinan matriks tersebut dengan $(-1)^n$. Dengan demikian matriks S adalah matriks negatif definite jika principal minor matriks S bergantian tanda negatif atau positif, dengan tanda negatif jika jumlah

baris (kolom) submatriks adalah ganjil, sebaliknya positif jika jumlah baris (kolom) submatriks adalah genap [12,13].

Sebagai contoh validasi untuk design terakhir, akan dibuktikan bahwa

$$\{1\} P > 0,$$

$$\{2\} (A_1 - B_1 F_1)^T P + P(A_1 - B_1 F_1) < 0$$

$$\{3\} (A_2 - B_2 F_2)^T P + P(A_2 - B_2 F_2) < 0$$

$$\{4\} \left(\frac{G_{ij} + G_{ji}}{2} \right)^T P + P \left(\frac{G_{ij} + G_{ji}}{2} \right) \leq 0$$

dimana,

$$G_{12} = A_1 - B_1 F_2 \Rightarrow G_{12}^T = A_1^T - F_2^T B_1^T$$

$$G_{21} = A_2 - B_2 F_1 \Rightarrow G_{21}^T = A_2^T - F_1^T B_2^T$$

Tabel 3.2 Nilai eigen matriks

{1}	{2}	{3}	{4}
0.1152	-0.0328	-0.0328	-0.0328
0.0121	-0.0005	-0.0008	-0.0007
0.0111	-0.0084	-0.0072	-0.0078
0.0111	-0.0058	-0.0055	-0.0057
0.0114	-0.0192	-0.0192	-0.0192
0.0120	-0.0008	-3.72×10^{-5}	-0.0005

Tabel 3.3 Nilai determinan *leading principal submatrix*

{1}	{2}	{3}	{4}
0.1102	-7.9685×10^{-3}	-6.6590×10^{-3}	-7.3138×10^{-3}
0.0013	1.5691×10^{-4}	1.3078×10^{-4}	1.4389×10^{-4}
1.52×10^{-5}	-2.3868×10^{-6}	-1.9893×10^{-6}	-2.1880×10^{-6}
1.70×10^{-6}	3.0169×10^{-8}	2.5097×10^{-8}	2.7633×10^{-8}
2.04×10^{-9}	-2.3434×10^{-11}	-9.3384×10^{-13}	-1.5105×10^{-11}
2.37×10^{-11}	1.2810×10^{-14}	7.1128×10^{-16}	1.0077×10^{-14}

Tabel 3.2 menunjukkan nilai eigen matriks dari masing-masing syarat kondisi kestabilan, dan Tabel 3.3 menunjukkan determinan dari *leading principal submatrices* yang bersesuaian untuk masing-masing syarat kondisi. Kolom {1} pada kedua tabel membuktikan bahwa matriks bersama P adalah definit positif. Hal yang sama ditunjukkan pada kolom {2} dan {3} dari masing-masing tabel bahwa matriks yang bersesuaian untuk syarat kondisi kestabilan adalah definit negatif. Syarat kestabilan {4} mencukupkan bahwa matriks yang bersesuaian minimal definit seminegatif, sedangkan hasil yang ditunjukkan pada kolom {4} dari masing-masing tabel menunjukkan bahwa matriks yang bersangkutan adalah definit negatif. Hasil ini menunjukkan bahwa model fuzzy dan pengendali fuzzy PDC yang dirancang memenuhi syarat kestabilan yang ditentukan.

